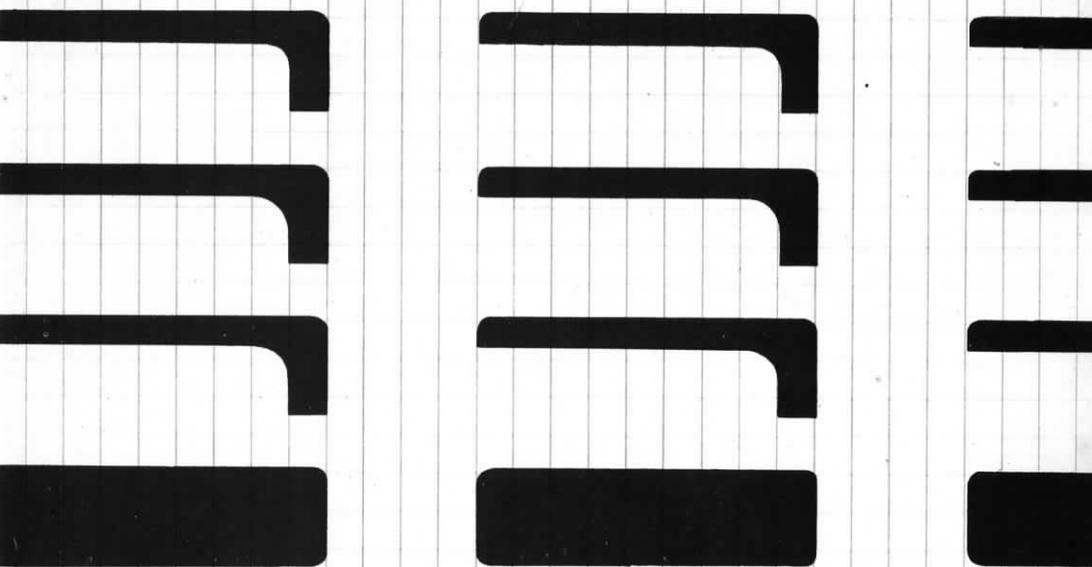


MICRO REVUE

LA REVUE DE L'INFORMATIQUE PORTABLE



© 1986 Bimestriel

Edité par PPC-T

N° 10 JANVIER - FEVRIER 1986

50 F.

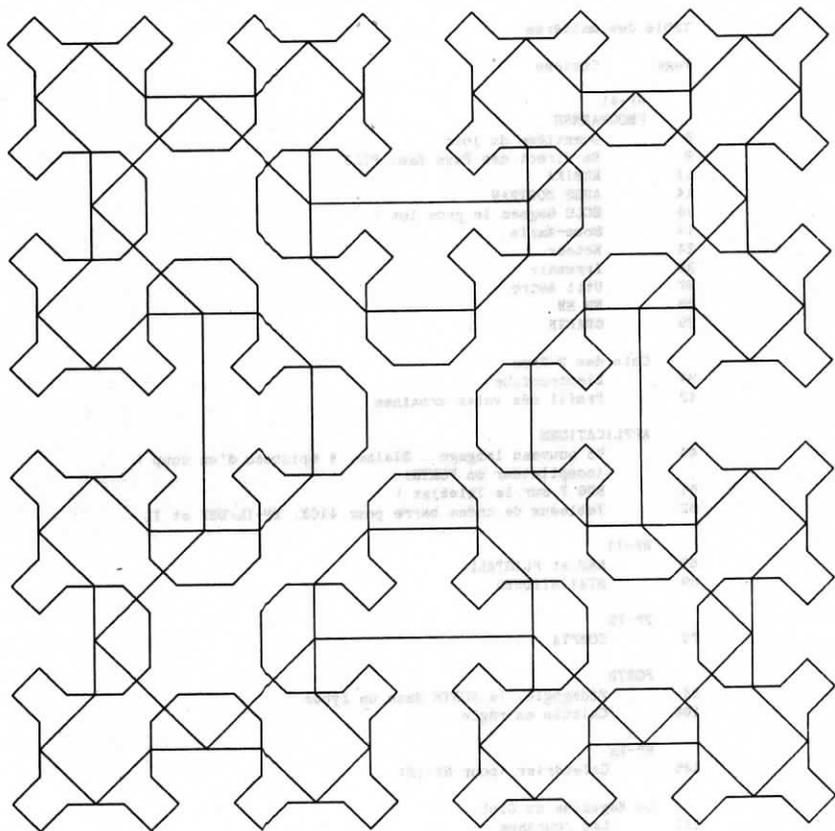


TABLE DES MATIERES

Table des matières

Page	Contenu
	HP-41
	PROGRAMMES
8	Quantième du jour
9	En direct des Pays Bas: PILE
11	RUBIKA
14	ARES NOUVEAU
18	ECLO Gagnez le gros lot !
19	Sous-Marin
24	Meteor
26	Arrondir
27	Util Astro
28	NH NH
29	GRAPHE
	Coin des U-Pros
34	Electronique
42	Profil des voies urbaines
	APPLICATIONS
44	Un nouveau langage : Blaise, 4 épisodes d'un coup ! (compilateur en FORTH)
61	BUG ? sur la Thinkjet !
62	Tableaux de codes barre pour 41CX, HP-IL/DEV et IL
	HP-71
67	KAS et PLISTALL
69	STATISTIQUES
	HP-75
78	COMPTA
	FORTH
88	Pédagogie, le FORTH dans un lycée
100	Calculs en règle
	HP-1x
105	Calendrier (pour HP-12)
	Le Magazine du Club
111	Les Journaux
117	Petites annonces
119	Réactions
121	Sur le marché
124	Nouvelles brèves
126	Argus de l'occasion HP
127	Une énigme : un bravo à qui trouve la solution !
128	Organisation de PPC-T
130	Avez-vous réglé votre cotisation ?

TABLE DES MATIERES



EDITORIAL

Chers amis,

Voici un numéro de MICRO-REVUE qui est bien en retard ! En fait, bien que le présent éditorial soit écrit le 4 Février, il n'est pas sûr que vous aurez reçu votre revue avant la fin du Mois. Rappelons que tout le travail fait au club est bénévole et que des problèmes personnels peuvent intervenir pour contrarier nos efforts.

Les "jeunes" du bureau, étudiants, se sont mobilisés et ont réalisé la majeure partie de l'édition du présent numéro. Sur leur lancée, ils ont également préparé une grande partie du prochain numéro (le 11) qui devrait donc être moins en retard.

Il y a eu pas mal d'améliorations dans l'administration du club. Toutes les rentrées d'argent font maintenant l'objet d'une facture et sont reprises en comptabilité, ce qui règle le problème des "avoirs" aux adhérents, puisque nous avons un suivi précis des comptes de chacun.

Nous vous proposons, entre autre, dans le présent numéro un argus des prix d'occasion remis à jours (et qui le sera régulièrement) sur la base du dernier tarif HP.

Nous vous proposons également un serveur minitel où vous pourrez laisser des messages à l'usage des autres membres du club et où vous trouverez des nouvelles de MICRO-REVUE. Ce serveur n'est pas à Toulouse, nous le partageons avec d'autres. Ne l'utilisez pas pour communiquer avec le bureau, notre budget ne nous permet pas des communications trop fréquentes.

Voilà. Je vous laisse à votre lecture.

Jean-Daniel Dodin

ADRESSE :

MICRO-REVUE, PPC-T, 77 rue du Cagire 31100 Toulouse France.

EDITION :

Edité par le club de Programmation Personnelle sur Calculatrice de Toulouse (PPC-T).

REDACTION-ADMINISTRATION :

Rédacteur en chef, directeur de la publication : Jean-Daniel Dodin ; secrétaires de rédaction : Gilles Barret et Jean-Yves Pasquier ; administration : Jean-François Sibille.

VENTE ET DISTRIBUTION :

Vente en librairie et par correspondance. Diffusion par l'éditeur. De plus, MICRO-REVUE est disponible à un tarif très avantageux pour les autres clubs et les associations qui le désirent, ainsi que pour les professionnels et les libraires qui souhaitent le revendre.

PARUTION :

Bimestriel, paraît en début de bimestre civil (Janvier, Mars, Mai, Juillet, Septembre, Novembre), 6 numéros par an.

TARIFS (franco de port) :

Prix de vente au numéro : 50F (60F par avion).

Abonnement individuel : 250F (300F par avion). Les envois à destination de l'Afrique n'arrivent pas toujours à destination, nous ne pouvons en aucun cas en assumer la responsabilité, dans la mesure du possible nous donner une adresse en Europe.

SOUSSION D'ARTICLES :

Tous les membres du club sont invités à fournir des articles pour MICRO-REVUE. L'original à nous faire parvenir doit être tapé à la machine avec un ruban bien noir, recto seul, simple interligne, avec une largeur maximale de 15 cm. Un envoi sur disquette IBM simple face, sur cassette HP82161A est recommandé. Les articles manuscrits sont acceptés s'ils sont calligraphiés sur papier blanc quadrillé 5x5, sans sauter de ligne, recto seul, au stylo noir. Nous faire parvenir cartes ou cassettes (qui vous seront remplacés) pour les programmes, ainsi qu'un listing. Cet original sera utilisé tel que pour la publication, soignez-le ! Ceux qui prévoient à l'avance d'écrire des articles sont invités à nous demander un "Kit Auteur" qui leur donnera les modèles et toutes les instructions nécessaires.

Les articles qui nous sont fournis sont certifiés originaux par leurs auteurs. Ni la revue, ni le club ni les auteurs ne sont responsables de l'usage qui sera fait des programmes qui sont uniquement fournis à titre documentaire, les utilisateurs sont seuls responsables de l'adaptation des programmes à leur propre usage. Ceci est spécialement vrai des programmes professionnels. Le fait, pour les auteurs, de nous envoyer un article ou un programme vaut autorisation de publication dans MICRO-REVUE ou dans toute autre publication du club, sans qu'aucune rémunération soit due à l'auteur de ce fait.

(c) 1985 REPRODUCTION INTERDITE

Les droits de reproduction des programmes et articles de MICROREVUE restent la propriété des auteurs et de la revue. En dérogation aux mentions ci-dessus, la reproduction des programmes est autorisée pour l'usage personnel des lecteurs, à l'exclusion de toute reproduction commerciale.

Les auteurs délèguent à la revue le droit d'autoriser la reproduction des articles et programmes dans d'autres revues à caractère non commercial.

PROGRAMMES

HP-41

Quantité de matière
Il est très facile de trouver dans les ouvrages de physique des données de référence de la masse molaire des gaz, des liquides et des solides. Les données de la masse molaire des gaz sont données dans les ouvrages de physique des gaz, des liquides et des solides. Les données de la masse molaire des liquides et des solides sont données dans les ouvrages de physique des liquides et des solides.

La masse molaire d'un gaz est donnée dans les ouvrages de physique des gaz, des liquides et des solides. La masse molaire d'un liquide et d'un solide est donnée dans les ouvrages de physique des liquides et des solides.

La masse molaire d'un gaz est donnée dans les ouvrages de physique des gaz, des liquides et des solides. La masse molaire d'un liquide et d'un solide est donnée dans les ouvrages de physique des liquides et des solides.

La masse molaire d'un gaz est donnée dans les ouvrages de physique des gaz, des liquides et des solides. La masse molaire d'un liquide et d'un solide est donnée dans les ouvrages de physique des liquides et des solides.

PROGRAMMES

Quantième du jour.

Il est très fréquent d'utiliser le numéro du jour dans l'année et tous les calendriers n'indiquent pas ces quantième. Le petit PRGM "NJ" (N° du Jour) nous indique le quantième du jour où l'on est ou de celui que l'on indique (de l'année en cours).

Utilisation : XEQ "NJ"

à l'affichage de 'J', taper simplement R/S pour obtenir le quantième du jour où l'on est;
pour obtenir le quantième d'un jour de l'année en cours, introduire la date dans le format DMY (jj,mmaaaa) et taper 'R/S'.

La forme DMY doit dans les 2 cas être adoptée et FIX Ø et CF 29 ne sont simplement que conseillés.

Ce PRGM présente l'avantage de ne pas avoir à être modifié chaque année, le 31 DECembre à minuit (?).....

Ø1 LBL "NJ" 1,Ø1 DATE E-2 / FRC E2 / + "J" CF 22
PROMPT FC?C 22 DATE DDAYS E + END

Le 'END' se trouve au Pas 18 et ce PRGM utilise 37 Octets, mais si vous l'appellez 'Q', vous gagnerez 1 Octet et il est peut-être possible de l'optimiser (?).

Bonnes PRGMmations, AS (T178)

EN DIRECT DES PAYS BAS :

PILE

La Pile

par Wouter Vloothuis

Comme tous les utilisateurs savent sans doute, la HP-41 dispose d'une pile opérationnelle (stack), se compose de registres X, Y, Z, T et le registre extra L.

Pourtant c'est dommage, que, en programma s on employe cela peu Bien de fois on prend directement les autres registres qui d'ailleurs sont présent en abondance (les registres 00/99 dependent de SIZE). Le disadvantage de la pile c'est-a-dire, est qu'on doit registrer où quel nombre se trouve. Contrairement les opérations dans la pile se va plus vite, que dans les autres registres, parceque le microprocesseur as ces registres directement en avance et les autres registres sont plus loin de lui et ce "marcher" coute du temps.

La programme "Plus haut - Plus bas" est écrit tout à fait à la pile. L'ordinateur génère un nombre qu'on doit trouver, où après chaque "jouet" est indiqué "plus haut" ou "plus bas".

Le programme s'explique simplement. Après il est appelé avec XEQ HL, quelque texte se passe et l'affichage clignotte avec "plus haut 0". Maintenant vous tapez un nombre et le programme continue, il ne faut pas R/S.

C'est un programme simple pour tourner à la pile. Il n'est pas necessaire de retenir beaucoup, ou de calculer beaucoup. Il n'est possible toujours de faire tourner un programme sans d'autres registres, mais ca vaut la peine d'examiner que ce va avec moins de registres par usage de la pile.

Remarques ou abrégé écrit à l'auteur

Wouter Vloothuis (10273)

ou à Piet van Unen, qui a traduit cet article, qu par en
Hollande dans PROMPT HP-CG de mai 1984.

P.A.G. VAN UNEN
PLATAANSTRAAT 18
2803 SW GOUDA
01820-31077

Plus haut - Plus bas par Wouter Vloothuis

- ligne 1-8 : le nomme est dans l'affichage, beau par FIX 0, CF 29.
- ligne 9-19 : un nombre arbitraire est generer et mis dans le registre X (4 et 10^x cote moins d'octets que 1E4 ou 1000)
- ligne 20-23 : soigner les textes
- ligne 24-25 : le compteur du nombre de tour est mis en nul (le "." est lu plus vite que le "0"),
- ligne 26-34 : on voit ici ou le nombre entré est plus grand ou plus moins et deparndent de cela*est rempli déjà avec PLUS HAUT ou PLUS BAS, après cela on saut à LBL 02.
- ligne 35-41 : le numérateur est retiré et affiché sur écran avec text, après cela on saut à LBL 00.
- ligne 42-52 : C'est la partie la plus intéressante Le texte* est affiché sur l'écran et le programme va á une boucle d'attend, en ligne 45 on regarde ou il y a d'entré numérique (flag 22). Non ? Va à LBL 02 et attendre de nouveau. Alors la pile est préparé dans ligne 47 et 48 à d'entré et dans ligne 49 le compteur est incrementé. Dans ligne 50 est un NOP, car la fonction ISG n'a pas un caractère conditionnel, mais est seulement compteur. Ligne 51 saut à LBL 01 et le nombre entré* nouveau est comparu avec le nombre generé.

03 PLUS HAUT PLUS BAS

20 " nombre entre "

27 "plus bas"

31 "plus haut"

36 "bien en"

38 "r fois"

- 01*LBL "HL"
- 02*LBL 00
- 03 *HOGER LAGER*
- 04 AVIEW
- 05 PSE
- 06 CF 29
- 07 CF 22
- 08 FIX 0
- 09 TAN
- 10 ABS
- 11 9821
- 12 *
- 13 ,211327
- 14 +
- 15 FRC
- 16 4
- 17 101X
- 18 *
- 19 INT
- 20 * GETAL TUSSEN *
- 21 "+0-10000"
- 22 AVIEW
- 23 PSE
- 24 ,
- 25 STO Z
- 26*LBL 01
- 27 *LAGER
- 28 ARCL X
- 29 X)?
- 30 GTO 02
- 31 *HOGER
- 32 ARCL X
- 33 X)?
- 34 GTO 02
- 35 BEEP
- 36 *GOED IN *
- 37 ARCL Z
- 38 "- KEER"
- 39 AVIEW
- 40 STOP
- 41 GTO 00
- 42*LBL 02
- 43 AVIEW
- 44 PSE
- 45 FC?C 22
- 46 GTO 02
- 47 X)?
- 48 RDN
- 49 ISC Z
- 50 CLA
- 51 GTO 01
- 52 END

RUBIKA

BOITE DES INSTRUCTIONS

RUBIKA se joue seul sur un damier de 16 cases qui contiennent les lettres A à P dans le désordre. Le but est bien sûr, de les remettre dans l'ordre alphabétique.

Le damier est divisé logiquement en 9 cases principales, contenant elles-mêmes 4 cases élémentaires:



Jouer une des 9 cases principales c'est effectuer une rotation d'un quart de tour dans le sens des aiguilles d'une montre, des contenus de 4 cases élémentaires qui la composent. Par exemple, si l'on a :

H	D	L	B
C	K	E	I
M	G	J	O
A	N	F	P

jouer la case

5 donne

H	D	L	B
C	G	K	I
M	J	E	O
A	N	F	P

Il fallait donc un programme qui :

- affiche l'état du damier,
- place aléatoirement les lettres
- pour une case principale logique donnée, effectue la rotation physique de ses 4 composantes élémentaires
- réaffiche le damier après rotation
- détermine si le joueur a gagné

RUBIKA

ROLE DES PROCEDURES

- FILL remplit le damier avec des lettres de code ASCII croissants
- SWAP mélange le tableau
- RAN renvoie un nombre aléatoire entre 1 et 16
- AFF imprime l'état du tableau
- READ lit le numéro de case joué et prépare la rotation
- ROT effectue la rotation
- TEST teste si le tableau est dans l'ordre alphabétique

MODE D'EMPLOI

Lancer : XEQ ▾ RUBIKA

Le programme demande "Reprendre?". Répondre N pour commencer une nouvelle partie, O pour reprendre une partie enregistrée sur carte.

Attendre que le tableau soit prêt

Le damier est imprimé

A la demande CASE ? répondre en donnant le numéro de la case logique que l'on veut permuter et ce, jusqu'à ce que le tableau soit remis en ordre (ce qui est loin d'être facile !)

Pour s'arrêter et enregistrer la partie sur carte, répondre Ø

Pierre BARTHELEMY
12, square Hopkinson
13004 MARSEILLE

T-659

01*LBL "RUBIKA"	53*LBL "TEST"	140*LBL "READ"
02 CF 21	54 1.016	141 CLST
03 "0"		142 X<Y
04 ASTO Y	55*LBL 03	143 CLX
05 "REPRENDRE ?"	56 ENTER†	144 9
06 RDN	57 INT	145 "CASE ?"
07 STOP	58 64	146 TONE 1
08 ROFF	59 +	147 PROMPT
09 ASTO X	60 RCL IND Y	148 STO 00
10 X=Y?	61 SF 04	149 X<Y?
11 GTO 00	62 X=Y?	150 SF 00
12 TONE 8	63 CF 04	151 1
13 1.016	64 FS? 04	152 X<Y?
14 RDTAX	65 RTN	153 SF 00
15 GTO 05	66 RCL Z	154 FS? 00
	67 ISG X	155 GTO 06
16*LBL 00	68 GTO 03	156 2
17 "NBR ?"	69 RTN	157 +
18 PROMPT	70*LBL "ROT"	158 X<Y
19 STO 00	71 RCL IND X	159 X<Y?
20 XEQ "FILL"	72 STO 00	160 SF 01
21 XEQ "SWAP"	73 RDN	161 FS? 01
	74 ENTER†	162 GTO 07
22*LBL 05	75 ENTER†	163 RCL Z
23 SF 21	76 4	164 3
24 CLD	77 +	165 -
25 XEQ "AFF"	78 RCL IND X	166 X<Y
26 XEQ "READ"	79 STO IND Z	167 X<Y?
27 XEQ "ROT"	80 RCL Z	168 SF 02
28 XEQ "TEST"	81 5	169 FS? 02
29 FS? 04	82 +	170 GTO 07
30 GTO 05	83 RCL IND X	171 SF 03
31 BEEP	84 STO IND T	172 GTO 07
32 "GAGNE"	85 RCL T	173*LBL 06
33 AVIEN	86 3	174 CF 21
34 RTH	87 -	175 TONE 8
	88 RCL IND X	176 X<Y
35*LBL "AFF"	89 STO IND T	177 X=0?
36 ADV	90 RCL 00	178 GTO 08
37 SF 12	91 STO IND Z	179 ASHF
38 1.016	92 RTH	180 STOP
		181*LBL 08
39*LBL 04	93*LBL "RAN"	182 "ERREUR"
40 RCL IND X	94 RCL 00	183 AVIEN
41 ACCNR	95 9821	184 PSE
42 RCL Y	96 *	185 SF 21
43 4	97 ,211327	186 GTO "READ"
44 MOD	98 +	187*LBL 07
45 INT	99 FRC	188 CLST
46 X=0?	100 STO 00	189 RCL 00
47 PRBUF	101 16	190 FS? 01
48 R†	102 *	191 0
49 ISG X	103 1	192 +
50 GTO 04	104 +	193 FS? 02
51 CF 12	105 INT	194 1
52 RTH	106 RTN	195 +
		196 FS? 03
		197 2
		198 +
		199 .END.

BELAIRE PATRICK

65, rue Paul Eluard

(.T567.)

A AVION, le 23/12/85

62210 AVION

MEILLEURS VOEUX AU PPC-T

CHER JEAN-DANIEL,

Ci-joint un programme paru dans JEUX&STRATEGIE n° II (OCT.-NOV. 81).
Nommé "ARES NOUVEAUX" il était prévu pour HP 34C mais malheureusement truffé
d'erreurs et incomplet. Il était écrit aussi pour TI 58, donc: pas de problèmes
pour la traduction. J'ai écourté un peu la règle du jeu pour ne pas monopoliser
trop de pages dans notre revue. Sinon elle est inchangée: vous êtes héritier
d'une population, d'une superficie de terres cultivées et d'une réserve de grain
-s. Chaque année vous aurez à répartir la moisson pour nourrir la population,
prenant en considération son taux de croissance et garder suffisamment de
semences pour l'année suivante. Vous devrez également utiliser une partie de
la récolte pour acheter des terrains à ensemençer pour alimenter les nouveaux
venus. Car votre but reste, en définitive, l'accroissement de la population.

AMICALEMENT,

Be...

P.S.: Peux-tu m'envoyer un exemplaire de LOOPING et me dire s'il est possible
que HEWLETT-PACKARD me l'envoie régulièrement ? -MERCII-

ARES NOUVEAU

Vous n'êtes pas responsable de la situation initiale.

Vous héritez d'une population, d'une superficie de terres cultivées et d'une réserve de grains. Chaque année, vous aurez à répartir le moisson pour nourrir la population, prenant en considération son taux de croissance et garder suffisamment de semences pour l'année suivante. Vous devrez également utiliser une partie de la récolte pour acheter des terrains à ensemercer pour alimenter les nouveaux venus. Car votre but reste, en définitive, l'accroissement de la population.

MARCHE A SUIVRE :

1. Entrez un nombre entre 0 et 1, (après avoir tapé AER"ARES", la 4I affiche: "ALEA ? 0/X/I").
2. "ANNEE : 0" est affiché, pressez R/S.
3. La population de votre ville est affichée: "POP. : xxx,n", où xxx représente le nombre d'habitants et n le nombre de 0 derrière la virgule représentent chacun 4% du taux d'accroissement de la population.
4. Pressez R/S, le nombre de quintaux de blé à votre disposition au début de l'année.
5. A nouveau R/S, la surface de terre arable disponible.
6. En pressant de nouveau R/S, vous retournez au pas 2.

REPARTITION DU BLE :

- a. Chaque personne a besoin d'un quintal de blé par an comme nourriture.

- b. Un quintal peut ensemercer 1 are.
- c. 1 are peut produire 10 quintaux l'année suivante.
- d. Il faut 20 quintaux pour acheter un are de terre.
- e. Il faut 6 personnes pour faire la récolte d'un are.
- f. Une personne "sacrifiée" peut être échangée contre 6 quintaux.

Ces six règles forment la base de votre logique pour déterminer la répartition des récoltes.

- 7. Votre stratégie décidée, pressez B.
- 8. La calculatrice vous demande combien de quintaux vous désirez acheter avec votre sacrifice, elle affiche : "Q. SACRIFICE?".
- 9. Pressez R/S, elle affiche : "Q. NOURRITURE ?", entrez le nombre de quintaux nécessaires pour nourrir votre population.
- 10. Pressez R/S, elle vous demande combien de quintaux voulez-vous comme semence. (affichage de : "Q. SEM. ?").
- 11. Pressez R/S, " Q. TERRAINS ?" : combien de quintaux voulez-vous utiliser pour acheter des terrains ?
- 12. Pressez R/S et vous retournez au pas 2.

Si vous constatez que vous avez fait une erreur de répartition, avant de presser R/S au pas 12, pressez A pour revoir la situation ou B pour refaire une répartition.

Notez que l'année ne change qu'après accomplissement avec succès du pas 12. Continuez de cette façon pendant 10 années. Quand l'année 9 aura été terminée, votre population finale sera affichée.

Chaque jeu est programmé pour atteindre une population de 999. Vous pouvez donc juger votre habileté en comparant votre score à 999. La semence initiale que vous choisirez est un guide approximatif de la difficulté du jeu : plus la semence est grande, plus la difficulté sera forte. Ceci parce que le taux de croissance initial (pendant le premier tour) est déterminé par cette semence.

Bon amusement,

-T567-

001 LBL "ARES"
002 SF 27
003 CLRG
004 "ALEA ? 0/X/I
005 PROMPT
006 ABS
007 FRC
008 STO 04
009 STO 03
010 I0
011 STO 09
012 I
013 STO 05
014 LBL 00
015 XEQ 07
016 DSE 09
017 GTO 00
018 3
019 I07X
020 RCL 05
021 /
022 STO 05
023 LASTX
024 , I
025 Y7X
026 STO 01
027 RCL 05
028 *
029 30
030 RCL 01
031 2I
032 *
033 -
034 /
035 INT
036 ST0 06
037 STO 02
038 RCL 03
039 STO 04
040 LBL 01
041 RCL 06
042 INT
043 RCL 02
044 X/Y?
045 GTO 02
046 RDN
047 STO 02
048 LBL 02
049 RCL 05
050 6
051 /
052 INT
053 RCL 02
054 X/Y?
055 GTO 03
056 RDN
057 STO 02
058 LBL 03
059 RCL 02
060 I0
061 *
062 STO 00
063 LBL A
064 RCL 06

065 FRC
066 I0
067 *
068 FIX 0
069 CLA
070 " ANNEE : "
071 ARCL X
072 PROMPT
073 RCL 04
074 FRC
075 I07X
076 I
077 +
078 2
079 /
080 STO 09
081 FIX IND 09
082 RCL 05
083 INT
084 CLA
085 "POP. : "
086 ARCL X
087 PROMPT
088 FIX 0
089 CLA
090 ARCL 00
091 " QUINTAUX "
092 PROMPT
093 RCL 06
094 INT
095 CLA
096 ARCL X
097 " ARES "
098 PROMPT
099 GTO A
100 LBL B
101 3
102 STO 09
103 FIX 4
104 "Q. SACRIFICE?"
105 PROMPT
106 ABS
107 INT
108 ST+ 00
109 5
110 +
111 6
112 /
113 INT
114 RCL 05
115 X/Y?
116 GTO 03
117 RDN
118 ST- 05
119 LBL 04
120 RCL 09
121 3
122 X=Y?
123 "Q. NOURRIT. ?"
124 RDN
125 2
126 X=Y?
127 "Q. SEM. ?"
128 RDN

129 I
130 X=Y?
131 "Q. TERRAINS ?"
132 PROMPT
133 ABS
134 INT
135 STO IND 09
136 DSE 09
137 GTO 04
138 RCL 01
139 RCL 02
140 RCL 03
141 +
142 +
143 RCL 00
144 X#Y?
145 GTO A
146 XEQ 07
147 FLX 4
148 RCL 03
149 RCL 05
150 INT
151 X/=Y?
152 GTO 05
153 -
154 ST+ 05
155 STOP
156 LBL 05
157 RCL 01
158 20
159 /
160 INT
161 ST+ 06
162 , I
163 ST+ 06
164 RCL 06
165 FRC
166 X#0?
167 GTO 01
168 LBL 06
169 RCL 05
170 INT
171 "POP. FIN : "
172 ARCL X
173 PROMPT
174 LBL 07
175 RCL 04
176 FRC
177 I07X
178 STO 04
179 50
180 /
181 I
182 +
183 ST* 05
184 RTN
185 END

Champigny le, 30 Juillet 85

PPCT

N. 25

DEBAQUE PHILIPPE
86, avenue Jules Valles
94500 CHAMPIGNY S/MARNE

MICRO-REVUE PPC-T
Mr. J.D. DODIN
77, rue du Cagire
31100 TOULOUSE

PROGRAMME ECART LOTO " ECL0 "

Chers amis HPistes,

Ce petit programme permettra aux joueurs assidus du "LOTO NATIONAL" d'être à jour et de peut-être trouver les six bons numéros (je leur souhaite) en connaissant régulièrement les écarts des 49 numéros après chaque tirages. Il suffit de se procurer pour se servir du programme la première fois, une liste à jour des écarts des numéros sortis, (Journaux spécialisés, à votre point de validation habituel). Quand vous aurez cette liste il suffira de l'introduire dans la mémoire de votre HP chérie. Lors des tirages suivants, il vous suffira d'appuyer sur A en mode USER et de passer dans le lecteur de cartes les deux cartes de données contenant les écarts précédents.

Ce programme fonctionne avec 1'XFUNCTIONS, le LECTEUR DE CARTE, le PRINTER et un SIZE 065 mis automatiquement par le programme (Prenez garde d'avoir assez de registres!). J'espère que ce programme vous facilitera pour faire le choix, combien difficile pour cocher les 6 bons numéros. Si par bonheur vous gagnez, n'oubliez pas de brûler un cerierge pour votre humble serviteur !!

DESCRIPTION DU PROGRAMME

- 1 à 8 Demande si vous actualisez les chiffres des précédents tirages, si c'est le cas pressez A en mode USER.
- 9 à 14 Initialisation
- 15 à 28 Zone de programme à utiliser seulement pour la première fois, lors de l'introduction des écarts des numéros, puis transcription sur les deux cartes de données.
- 29 à 35 Remise en taille standard du programme (partie facultative ou à modifier selon votre taille habituelle de mémoire), mais ne pas oublier d'annuler le XEQ 17 au pas 27 si vous ne souhaitez pas utiliser se sous programme.
- 36 à 91 Introduction de la date du tirage, puis passer les cartes de données contenant les précédents tirages et introduction des numéros à cette date.
- 92 à 108 Incrémentatation d'une unité de tous les registres puis mise à zéro des registres contenant les anciens écarts des numéros sortis.
- 109 à 127 Impression des écarts (ne pas oublier d'éliminer le XEQ 17, si vous n'avez pas introduit se sous programme).

Bonne programmation
et bonne chance aux futurs millionnaires

Ph. DEBAQUE

```

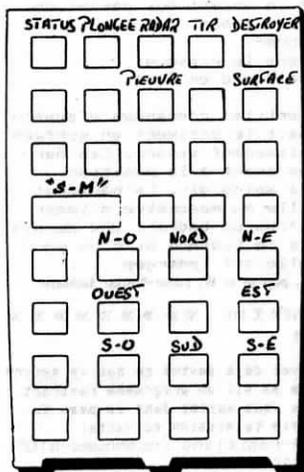
81*LBL "ECL0"
82 *ACTU. N.=LBL A"
83 AVIEW
84 ADV
85 ADV
86 ADV
87 ADV
88 STOP
89 65
18 PSIZE
11 CF 29
12 FIX 0
13 1.04901
14 STO 00
15*LBL 50
16 CLA
17 *ECART NO."
18 ARCL 00
19 "+ ?"
20 PROMPT
21 STO IND 00
22 ISG 00
23 GTO 50
24 1.049
25 WDTAX
26 BEEP
27 XEQ 17
28 OFF
29*LBL 17
30 26
31 PSIZE
32 SF 29
33 FIX 2
34 CLRG
35 PTN
36*LBL A
37 SF 12
38 AN
39 *DATE TIRAGE?"
40 PROMPT
41 PPA
42 CLA
43 AOFF
44 CF 12
45 ADV
46 65
47 PSIZE
48 1.04901
49 STO 00
50 FIX 0
51 CF 29
52 1.049
53 RDTAX
54 *(X)(X)(X)(X)
55 PRA
56 * TIRAGE"
57 PRA
58 *(X)(X)(X)(X)
59 PRA
60 AVIEW
61 CLA
62 *121314"
63 PROMPT
64 SF 12
65 PRX
66 STO 51
67 RDN
68 PRX
69 STO 52
70 RDN
71 PRX
72 STO 53
73 RDN
74 PRX
75 STO 54
76 *51617"
77 PROMPT
78 CLA
79 PRX
80 STO 55
81 RDN
82 PRX
83 STO 56
84 RDN
85 PRX
86 STO 57
87 *(X)(X)(X)(X)
88 PRA
89 ADV
90 ADV
91 CLA
92*LBL 01
93 1
94 ST+ IND 00
95 ISG 00
96 GTO 01
97 0
98 STO IND 51
99 STO IND 52
100 STO IND 53
101 STO IND 54
102 STO IND 55
103 STO IND 56
104 STO IND 57
105 1.04901
106 STO 00
107 ADV
108 ADV
109*LBL 03
110 RCL 00
111 10
112 XY?"
113 *
114 ARCL 00
115 "+ ECART "
116 ARCL IND 00
117 PRA
118 CLA
119 ISG 00
120 GTO 03
121 ADV
122 ADV
123 1.04901
124 WDTAX
125 BEEP
126 XEQ 17
127 OFF
128 .END.

```


Evidemment, comme tout jeu de rôle vous possédez un certain nombre de points de vie (FORCE) qu'il vous faudra maintenir positifs. De plus votre sous-marin ayant une autonomie limitée au contenu de ses réservoirs de gazole, il est bon de jeter un oeil sur la jauge (synthétisée par les flags 0 à 4) de temps en temps. Il est à noter qu'en cas de destruction d'un objectif vous gagnez des points de vie, une torpille et un plein de gazole !

*** Action des touches en mode USER ***

Se référer à l'overlay ci-dessous



[A]: Affichage de la position du sous-marin (suivant les axes X, Y et Z), de votre force et du nombre de torpilles restantes.

[B]: Plongée. Vous plongez de 10m dans la direction en cours. Immersion maximale 50 mètres.

[C]: Radar. Indique la direction et la distance du sous-marin à la cible, et si le tir est possible ou pas.

[D]: Tir d'une torpille sur l'objectif uniquement (plongée ou surface).

[E]: Engage le combat avec un destroyer.

[H]: Engage le combat contre la pieuvre.

[J]: Remontée. Immersion diminuée de 10m à chaque fois. Si vous êtes en immersion périscopique, affichage de "SURFACE". Si vous êtes en surface affichage d'"IMPOSSIBLE".

Les touches numériques déplacent le sous-marin dans la direction qu'elles symbolisent, la touche 5 étant prise comme centre. Exemple: touche [7]: déplacement vers le nord-ouest ou plongée vers le nord-ouest.

REMARQUE: Si vous êtes en plongée Z>0, la direction ne change pas, mais la vitesse est divisée par 5. Les déplacements en surface sont donc

très rapides et l'on peut ainsi repérer assez facilement l'endroit de l'objectif, mais pour bien s'approcher de celui-ci (pour avoir des bonnes chances de la détruire), il faudra certainement plonger. Ce ne sera, cependant, pas nécessaire à chaque fois. A vous de juger.

*** Actions & réactions ***

En surface vous pouvez rencontrer des mines, des avions, des destroyers. En plongée vous pouvez rencontrer des pieuvres géantes, du corail et des épaves piégées.

A l'affichage des messages, il faudra appuyer sur la bonne touche. Vous avez juste le temps d'une pause (0,8s.).

Destroyer: Plongée ou attaque, à tout moment vous pouvez cesser le combat en plongeant. Touches [B] ou [E] à l'affichage "DESTROYER".

Avions: Plongez! Réagissez vite, car ils vous ont déjà mitraillé et enlevé 5 points de vie.



Mines: Appuyez sur une touche de déplacement ([1] à [4] et [6] à [9]) au choix, mais n'y réfléchissez pas trop longtemps.

Corail: Surface !

Pieuvre: Engager le combat pour la faire fuir (touche [H]). Attention elle peut revenir à l'assaut. Vous ne pouvez faire surface car elle vous entraîne dans les abîmes par son propre poids (attention à l'implosion).

Panne sèche: Vous avez mis trop de temps pour essayer de détruire cet objectif, tant pis pour vous !

Il n'y a que l'épave piégée contre laquelle vous ne pouvez rien. Comme la pieuvre, elle déplacera votre sous-marin de 0 à 4 cases dans le sens nord-est. Attention à la côte qui est symbolisée par les bords du damier de jeu. Tout contact avec elle vous coûtera 30 points de vie et le message "#ENSABLE#" si vous demandez votre position ([A]).

Gains de points

Une cible: 25 points, 1 torpille et un plein.

Un destroyer: 35 points.

Pertes de points

Avions: 5 puis 10 si vous ne réagissez pas à CHAQUE attaque.

Destroyer: 20 points par attaque.

Mines: 15 points à chaque fois qu'elle explose.

Pieuvres: 2 points.

Corail: 25 points si vous ne faites pas surface à temps.

Epave piégée: 8 points.

Côtes touchées: 30 points.

Si les cibles ripostent (lorsque vous les ratez) tout dépend de l'avarie qu'elles vous causeront.



*** Mémoires ***

R00: Nombre pseudo-aléatoire

R01 & R02: X et Y de la 5^{ème} cible

R03 & R04: X et Y 4^{ème} cible

R05 & R06: X et Y 3^{ème} cible

R07 & R08: X et Y 2^{ème} cible

R09 & R10: X et Y 1^{ère} cible

R11: X du sous-marin

R12: Y du sous-marin

R13: Z du sous-marin

R14: Distance cible - sous-marin

R15: Commande manœuvre

R16: Nombre de torpilles

R17: Points de vie

R18: Pointeur du No de cible

R19: Niveau de gazole

*** Matériel ***

HP 41 munie de 2086 octets et
SIZE 020

Un module X-FUNCTIONS

Un module TIME

Le module PPC-ROM

Si vous n'avez pas ce module,
remplacez les XROM "T1" par 13
TONE 89 (9F 59).

*** Flags ***

F00: Niveau de gazole <101.

F01: Niveau de gazole <201.

F02: Niveau de gazole <301.

F03: Niveau de gazole <401.

F04: Niveau de gazole maximum

F05: Mine

F06: Corail

F07: UP= Sud; DOWN= Nord

F08: UP= Ouest; DOWN= Est

F09: Combat pieuvre

F10: Attente d'un ordre pendant un
combat

F11: Combat destroyer

F12: ~

F13: ~

F14: ~

F14: Combat avions

F16: ~

F17: Sous-marin en immersion

F18: ~

F19: Attente d'une action (LBL "Z")

REIGNER CyrilPPC-T282***

4PRP *S-M*
 4 10:10 10.04%
 4%
 01NLBL *S-M*
 4 * 0111 0 * RCL I STO 0%
 4SIZE? 20 X?Y? PSIZES
 4 * *S0US-MARIN* AVIEM %
 4STO 11 STO 12 9 %
 4STO 15 STO 18 E1 %
 4STO 14 STO 16 7 * %
 4STO 17 TIME * FRC %
 4STO 00 41 STO 19 %
 4S01 E-4 STO 13 XE0 63%
 4%
 4 31NLBL 10%
 4XE0 99 TONE ^ E2 4%
 4INT STO IND 14 DSE 14%
 4GTO 10 XE0 *A* %
 4%
 4 41NLBL *C*%
 4FS? 10 GTO *V* XE0 80%
 4XE0 *X* "DIST." %
 4ARCL 14 FS? 07 *M S*%
 4FC?C 07 *M M* FS? 08%
 4 *M0* FC?C 08 *ME* %
 4AVIEM PSE %
 4%
 4 50NLBL *Z*%
 4RCL 17 X<0? GTO *P* %
 4SF 19 *ACTION ?* %
 4TONE 1 PROMPT %
 4%
 4 60NLBL *A*%
 4FS? 10 GTO *V* XE0 80%
 4FIX 0 *IX: * RCL 11 %
 4SF 25 ADATE FC? 25 %
 4SF 12 SF 25 *MILY: %
 4RCL 12 ADATE *M* %
 4FS?C 25 FS?C 12 %
 4 * *M* ENGABLE* AVIEM %
 4PSE * * * * * IZ: * RCL 13%
 4ADATE *M* <<< AVIEM %
 4PSE *FRC= * ARCL 17%
 4AVIEM PSE %
 4 *TORPILLES: * ARCL 16%
 4AVIEM PSE FIX 1 %
 4FS?C 23 RTN GTO *Z* %
 4%
 4%
 4105NLBL 05%
 4STO 15 E1 DSE 19 %
 4GTO 00 *PLUS DE GAZOLE*%
 4AVIEM RCL b CF 00 %
 4TONE 0 SF 00 TONE 1 %
 4DSE Y STO b %
 4 *PANNE SECHE* GTO 77 %
 4%
 4%
 4121NLBL 00%
 4RCL 19 X<Y / FRC %
 4I=0? CF IND L CF 05 %
 4CF 10 %
 4%
 4130NLBL 80%
 4FS? 05 GTO 51 FS? 11%
 4GTO 53 FS? 15 GTO 54%
 4FS? 60 GTO 55 FS? 09%
 4GTO 57 RTN %
 4%
 4142NLBL 08%
 4%
 4430NLBL *B*%
 48 XE0 05 5 FS? 17 %
 4S1GN ST+ X ST+ 12 %
 4XE0 60 RCL 12 E2 %
 4X<Y? GTO *Z* GTO *K*%
 4%
 4157NLBL 02%
 4%
 458NLBL *2*%
 42 XE0 05 5 FS? 17 %
 4S1GN ST+ X ST- 12 %
 4XE0 60 RCL 12 X<0? %
 4GTO *K* GTO *Z* %
 4%
 4171NLBL 06%
 4%
 4172NLBL *6*%
 46 XE0 05 5 FS? 17 %
 4S1GN ST+ X ST+ 11 %
 4XE0 60 RCL 11 E2 %
 4X<Y? GTO *Z* GTO *K*%
 4%
 4186NLBL 04%
 4%
 4187NLBL *4*%
 44 XE0 05 5 FS? 17 %
 4S1GN ST+ X ST- 11 %
 4XE0 60 RCL 11 X<0? %
 4GTO *K* GTO *Z* %
 4%
 4200NLBL 09%
 4%
 4201NLBL *9*%
 49 XE0 05 5 FS? 17 %
 4S1GN ST+ 11 ST+ 12 %
 4XE0 60 E2 RCL 11 %
 4X<Y? GTO *K* CLX %
 4RCL 12 X<Y? GTO *K* %
 4GTO *Z* %
 4%
 4219NLBL 07%
 4%
 4220NLBL *7*%
 47 XE0 05 5 FS? 17 %
 4S1GN ST+ 12 ST- 11 %
 4XE0 60 E2 RCL 12 %
 4X<Y? GTO *K* RCL 11 %
 4X<0? GTO *K* GTO *Z*%
 4%
 4237NLBL 03%
 4%
 4238NLBL *3*%
 43 XE0 05 5 FS? 17 %
 4S1GN ST- 12 ST+ 11 %
 4XE0 60 E2 RCL 11 %
 4X<Y? GTO *K* RCL 12 %
 4X<0? GTO *K* GTO *Z*%
 4%
 4255NLBL 01%
 4%
 4256NLBL *1*%
 4 E I E0 05 5 FS? 17 %
 4S1GN ST- 11 ST- 12 %
 4XE0 60 RCL 11 X<0? %
 4GTO *K* RCL 12 X<0? %
 4GTO *K* GTO *Z* %
 4%
 4272NLBL 60%
 4XE0 99 11 * 5 + %
 4SF IND X FS? 17 GTO 11%
 4FC?C 14 FS?C 10 %
 4GTO *M* FS? 11 GTO *L*%
 4FC? 15 FS?C 13 GTO *M*%
 4FC? 17 GTO 00 %
 4%
 4291NLBL 11%
 4FS? 06 GTO *0* FS?C 08%
 4GTO *U* FS? 09 GTO *S*%
 4%
 4298NLBL 00%
 4CF IND X RTN %
 4%
 4301NLBL *B*%
 4CF 11 CF 15 XE0 80 %
 4 *PLONGEE* %
 4%
 4306NLBL 93%
 4AVIEM SF 17 ISB 13 %
 4GTO IND 15 %
 4 *TU IMPLOSES* GTO 77 %
 4%
 4%
 4313NLBL *J*%
 4CF 06 XE0 80 RCL 13 %
 4INT X<0? GTO 00 %
 4 *IMPOSSIBLE* AVIEM %
 4GTO *Z* %
 4%
 4323NLBL 00%
 4 *SURFACE* E1 X<Y? %
 4 *MONTEE* AVIEM X<Y? %
 4CF 17 CF 06 ST- 13 %
 4GTO IND 15 %
 4%
 4334NLBL 99%
 4RCL 00 R-D FRC STO 00%
 4RTN %
 4%
 4340NLBL *D*%
 4CF 10 XE0 80 DSE 16 %
 4FS? 30 GTO 65 7 %
 4RCL 14 *TIR TORPILLE*%
 4X<Y? *TROP LOIN* AVIEM%
 4TONE 1 X<Y? GTO *Z* %
 4 18 + XE0 99 X<Y? %
 4GTO 00 *TIR RATE* %
 4AVIEM TONE 8 TONE 0 %
 4GTO *V* %
 4%
 4365NLBL 00%
 4 *TOUCHEE* AVIEM TONE 1%
 425 ST+ 17 E ST+ 16%
 4 * * * * * BOUN * * * * * AVIEM %
 4TONE 0 TONE 3 TONE 7%
 4TONE 3 TONE 7 TONE 2%
 4TONE 7 TONE 2 TONE 3%
 4CLX STO IND 18 2 %
 4ST- 18 RCL 18 X<0? %
 4GTO *Y* FIX 0 X<Y / %
 4 *ND * < * ARCL X %
 4 * * * * * CIBLE * E X<Y? %
 4 * * * * * AVIEM TONE J 41%
 4RCL 19 %
 4%
 4405NLBL 28%
 4 *VENTER* "GAZOLE" %
 4ADATE *M* E1 / %
 4FRC X=0? SF IND L %
 4AVIEM TONE IND L SIGNS
 4 X<Y? GTO 28 STO 19%
 4 *S0UTES: 0K* AVIEM %
 4TONE X FIX 1 %
 4 * * * * * 60 * * * * * AVIEM %
 4BEEP XE0 63 GTO *Z* %
 4%
 4%
 4431NLBL *0*%
 4 *R1POSTE* AVIEM %
 4 *XRM *T1* XE0 99 ,1 %
 4RCL 14 + X<Y? GTO 12%
 4 *M RATEE* AVIEM TONE 0%
 4TONE 0 GTO 00 %
 4%
 4446NLBL 12%
 4XE0 99 ,4 X<Y? SF 05%
 4 *PETIT* FS? 05 *GROSS*%
 4 *ME AVARIE* AVIEM %
 4TONE 1 TONE 1 E1 %
 4FS?C 05 ST+ X ST- 17%
 4RCL 17 X<0? GTO *P* %
 4%
 4%
 4465NLBL 00%
 4SF 10 *ALORS ?* AVIEM%
 4TONE 9 PSE GTO *V* %
 4%
 4%
 4472NLBL 65%
 4 *PLUS DE TORPILL *ME*%
 4AVIEM TONE 0 %
 4 * * * * * PERDU * * * %
 4%
 4478NLBL 77%
 4AVIEM TONE 7 SF 11 %
 4OFF GTO 75 %
 4%
 4484NLBL *M*%
 4SF 05 *UNE NINE* AVIEM%
 4PSE %
 4%
 4489NLBL 51%
 415 ST- 17 %
 4 *ELLE EXPLOSE* AVIEM %
 4%

4588BLBL "M" \$
 4FS? 10 GTO "V" CF 09 \$
 4XED 80 FS? 19 GTO "Z" \$
 4"MECHARGE ELECTRA" \$
 4"LIQUE" AVIEM TONE \$ \$
 4Z ST- 17 XED 99 ,6 \$ \$
 4X(=Y? GTO 00 \$ \$
 4"ELLE S'EN VA" AVIEM \$ \$
 4BEEP GTO "Z" \$ \$
 4649BLBL 00 \$ \$
 4XED 63 "ELLE REVIENT" \$ \$
 4AVIEM PSE GTO 57 \$ \$
 \$ \$
 4 \$ \$
 4615BLBL 63 \$ \$
 4XED 99 5 + INT \$ \$
 4ST+ 11 ST+ 12 CF 09 \$ \$
 4 E2 RCL 11 X)? \$ \$
 4GTO "K" CLX RCL 12 \$ \$
 4X(=Y? RTN \$ \$
 4631BLBL "K" \$ \$
 4"COTE TOUCHEE" AVIEM \$ \$
 4TONE 5 "VOIE D'EAU" \$ \$
 4AVIEM TONE 7 30 \$ \$
 4ST- 17 GTO "Z" \$ \$
 4641BLBL "E" \$ \$
 4FS? 10 GTO "V" CF 11 \$ \$
 4XED 80 FS? 19 GTO "Z" \$ \$
 4") FEU <<< AVIEM \$ \$
 4XROM "TI" XED 99 ,6 \$ \$
 4" A COTE" X)? AVIEM \$ \$
 4X() GTO 53 \$ \$
 4"IL EXPLOSE" AVIEM \$ \$
 4BEEP TONE 5 TONE B 35 \$ \$
 4ST+ 17 GTO "Z" \$ \$
 \$ \$
 4668BLBL "X" \$ \$
 4RCL 13 11 X)? GTO 14 \$ \$
 4STEN RCL 18 + \$ \$
 4RCL IND L RCL 11 - \$ \$
 4X() SF 08 X'2 \$ \$
 4RCL IND V RCL 12 - \$ \$
 4X() SF 07 X'2 + \$ \$
 4SRT STO 14 7 X)? \$ \$
 4GTO 00 "TIR POSSIBLE" \$ \$
 4AVIEM TONE 9 TONE B \$ \$
 4RTN \$ \$
 \$ \$
 4678BLBL 00 \$ \$
 4"RIEN EN VUE" AVIEM \$ \$
 4TONE 3 TONE 3 RTN \$ \$
 \$ \$
 \$ \$
 4703BLBL 14 \$ \$
 4"PERISCOPE TROP" \$ \$
 4"CCOURT" AVIEM GTO "Z" \$ \$
 \$ \$
 \$ \$
 4708BLBL "Y" \$ \$
 4"FANTASTIQUE" AVIEM \$ \$
 4BEEP "TU PASSES" AVIEM \$ \$
 4TONE 7 TONE 8 TONE 5 \$ \$
 4TONE 7 * AMIRAL * \$ \$
 4AVIEM TONE H TONE H 1 \$ \$
 4TONE H TONE 8 TONE 8 \$ \$
 4TONE 7 SORT SORT \$ \$
 4TONE 7 SORT SORT \$ \$
 4TONE 3 \$ \$

2086 OCTETS
 298 REGISTRES

USER KEYS:
 11 "A"
 12 "B"
 13 "C"
 14 "D"
 15 "E"
 -21 X(<
 -22 X(<F
 23 "H"
 -23 XROM 57,53
 -24 XROM 57,54
 25 "J"
 -25 XROM 57,55
 -31 XROM 28,43
 -32 DATE
 -33 BBOBEEP
 41 "S-H"
 -41 SIZE?<
 42 ENDIR
 -42 DSE
 43 XROM 00,27
 -44 SW
 51 PACK
 -51 XROM 01,60
 52 "7"
 -52 XROM 01,53
 53 "8"
 -53 XROM 01,54
 54 "9"
 -54 XROM 01,55
 61 DEL
 -61 XROM 05,60
 62 "4"
 -62 XROM 05,53
 -63 XROM 05,54
 64 "6"
 -64 XROM 05,55
 71 XTOA
 -71 ATOX
 72 "1"
 -72 XROM 61,25
 73 "2"
 74 "3"
 81 SAVEP
 -81 BEEP
 -82 XROM 61,09
 -83 XROM 61,45
 -84 PRONPT

METEOR

LAVILLE Laurent (T.634)
45 Rue Riquet
75019 PARIS

Cher Jean-Daniel,

Voici une version pour HP-41 (C+Time ou CX) du programme de Michel SUSINI parut dans l'ordinateur de poche en juin 83.

Le jeu se déroule en deux dimensions. Votre vaisseau spatial se trouve au milieu d'aérolithes et votre seule chance d'en réchapper consiste à passer au-dessus de ces obstacles dont l'altitude varie entre 0 et 9.

Cependant votre vaisseau ne peut naviguer qu'entre l'altitude 0 et 7.

Essayer de monter plus haut, ou de descendre plus bas entraînerait la désintégration de votre aéronef. Alors comment faire pour passer au-dessus d'un 7, 8 ou 9, car il faut que votre altitude soit strictement supérieure à celle de l'obstacle. Pour cela vous disposez d'un canon qui si il fait mouche permet de diminuer l'altitude de l'obstacle d'une unité.

Une météorite ne peut être touchée que par un tir venant d'une altitude inférieure. Chaque météorite touchée rapporte des points. Un bonus est donné au joueur à chaque fois que celui-ci amène une météorite à l'altitude 0. L'espace de jeu diminue lorsque vous atteignez un score supérieur à 3000 points et diminue encore tous les 2000 autres points.

Les commandes du vaisseau sont les suivantes:

E_ pour monter
J_ pour descendre
I_ pour tirer ,ceux sont les touches des 2 rangées supérieures que l'on utilise en mode USER.

L'affichage est alors le suivant durant la partie:

Y:Z A
ou: Y= obstacle suivant

Z= obstacle en vue

A= altitude de votre vaisseau

l'obstacle Z se déplace de la gauche vers la droite et il faut être remonter au-dessus de lui avant qu'il ne soit sur vous.

En espérant que mes explications auront été claires je vous laisse au plaisir de vous divertir entre deux affaires plus sérieuses.

Bonne programmation à tous

01	<u>LEL "NEZBOR"</u>	52	<u>LEL 02</u>	103	RCL 01
02	"I B"	53	DSE 06	104	-
03	RCL M	54	FS? 30	105	RCL 03
04	STO d	55	GTO 07	106	+
05	CLRG	56	XEQ 04	107	RCL 06
06	TIME	57	7	108	*
07	INT	58	RCL 02	109	ST+ 04
08	LASTX	59	E	110	<u>LEL 03</u>
09	*	60	TONE IND 03	111	XEQ 04
10	PRC	61	SE 04	112	GTO 02
11	STO 09	62	PSE	113	<u>LEL 04</u>
12	<u>LEL 01</u>	63	CF 04	114	CLA
13	RCL 10	64	"RATE"	115	ARCL 00
14	RCL 04	65	AVIEW	116	"I:"
15	X>Y?	66	TONE 3	117	RCL 03
16	STO 10	67	GTO 03	118	RCL 08
17	X<=Y?	68	<u>LEL E</u>	119	-
18	X<>Y	69	CF 04	120	<u>LEL 05</u>
19	"HIGH SCORE"	70	" UP↑"	121	"I "
20	AVIEW	71	AVIEW	122	DSE X
21	PGE	72	+	123	GTO 05
22	"TO MEMBER: "	73	X>Y?	124	ARCL 01
23	ARCL X	74	GTO 10	125	LASTX
24	AVIEW	75	STO 02	126	<u>LEL 06</u>
25	TONE 73	76	GTO 03	127	"I "
26	TONE 72	77	<u>LEL J</u>	128	DSE X
27	TONE 87	78	CF 04	129	GTO 06
28	TONE 73	79	"DOWN↓"	130	ARCL 02
29	TONE 72	80	AVIEW	131	AVIEW
30	TONE 87	81	-	132	RTN
31	TONE 73	82	X<0?	133	<u>LEL 07</u>
32	"ALERT"	83	GTO 10	134	RCL 01
33	AVIEW	84	STO 02	135	RCL 02
34	SE 25	85	STO 03	136	X<=Y?
35	SE 09	86	<u>LEL I</u>	137	GTO 10
36	4	87	CF 04	138	RDN
37	GTO 00	88	"↑FIRE↑"	139	X=0?
38	5	89	AVIEW	140	XEQ 08
39	STO 01	90	TONE 80	141	RCL 04
40	7	91	TONE 87	142	RCL 07
41	STO 02	92	TONE 80	143	X<=Y?
42	9	93	TONE 87	144	XEQ 00
43	STO 03	94	RCL 02	145	RCL 00
44	STO 08	95	RCL 01	146	STO 01
45	E	96	X<=Y?	147	IO
46	STO 06	97	GTO 03	148	RCL 09
47	3 E3	98	E	149	R-D
48	STO 07	99	BT- 01	150	PRC
49	CLX	100	19	151	STO 09
50	STO 04	101	RCL 02	152	*
51	STO 05	102	-	153	INT

154 STO 00	165 TONE 9	176 RTN
155 RCL 03	166 TONE 8	177 <u>LBL 10</u>
156 STO 08	167 TONE 6	178 "GAME OVER"
157 STO 02	168 TONE 7	179 BEEP
158 <u>LBL 08</u>	169 RTN	180 RCL 06
159 RCL 06	170 <u>LBL 09</u>	181 RCL 05
160 350	171 E	182 *
161 *	172 ST+ 06	183 ST+ 04
162 ST+ 05	173 2	184 PROPT
163 "BONUS"	174 ST* 07	185 STO 01
164 AVIEW	175 RGE 03	186 END

PS: Le module horloge n'est pas nécessaire. Il suffit de remplacer la fonction TIME par une demande de nombre aléatoire. Pour refaire une partie dès qu'une vient de s'achever, il suffit d'appuyer sur R/S.

Codes hexadécimaux de la ligne 02: 31 00 80 00
 arme les flags 26 27 31 40

La commande que vous effectuez au moyen des touches E, I, et J n'est validée qu'à l'apparition de l'indicateur 4.

ARRONDIR

Voici une petite routine qui permet d'arrondir un réel à n décimales sans changer le FIX (i.e sans passer en FIX n)

Placer le nombre à arrondir en y, le nombre de décimales voulues en x.

```
01 LBL "RD"
02 10X
03 STO Z
04 *
05 INT
06 X(Y)
07 /
08 END
```

P. BARTHELEMY
 12 square Hopkinson
 13004 MARSEILLE
 PPC-T 659

UTIL ASTRO

SOTIROPOULOS C.

T434

13.a. A. ARCHAMBAULT
95118 SANNOIS

LES 5 PROGRAMMES UTILI-
TAIRES QUI SUIVENT
PEUVENT SERVIR A EXPLOI-
TER LES ROUTINES D'
ASTRONOMIE DU MODULE
NAVIGATION.

LE PRGM (TSD) FOURNIT LE
TEMPS SIDERAL A GREEN-
WICH. (en H.M.S.)

LE PRGM (ASTRE) DONNE L'
ASCENSION DROITE ET LA
DECLINAISON (D.M.S.)

LE PRGM (REQ) TRANSFORME
LES COORD. RECT. EN EQU.

LE PRGM (EEO) LES COORD.
ECLIPT. EN EQU.
-LO=LONGITUDE CELESTE EN
DMS
b=LATITUDE CELESTE DE
-90 A +90
e=OBLIQUITE (VOIR MANUEL
DE NAVIG. PAGE 43)

LE PRGM (DIF) PEUT ETRE
UTILISE SOIT POUR TROU-
VER LA DIFF. EN ad ET EN
dec. ENTRE DEUX CORPS
CELESTES, SOIT POUR DETER-
MINER LE MOUVEMENT
PROPRE EN ad ET EN dec.
D'UN ASTRE, POUR DEUX
EPOQUES DIFFERENTES.

```

01*LBL *TSD*
02 SF 21
03 37
04 PSIZE
05 XROM *D+T*
06 RCL 30
07 XROM *JD*
08 XROM *GST*
09 15
10 /
11 HMS
12 *TSD=*
13 ARCL X
14 AVIEW
15 END
    
```

```

01*LBL *ASTRE*
02 SF 21
03 54
04 PSIZE
05 XROM *D+T*
06 XROM *ABODY*
07 RCL 45
08 360
09 MOD
10 15
11 /
12 24
13 -
14 ABS
15 HMS
16 *ad=*
17 ARCL X
18 AVIEW
19 *d=*
20 RCL 46
21 HMS
22 ARCL X
23 AVIEW
24 END
    
```

```

01*LBL *REQ*
02 30
03 PSIZE
04 SF 21
05 FIX 4
06 CLST
07 *Y=?*
08 PROMPT
09 *Y=?*
10 PROMPT
11 *Z=?*
12 PROMPT
13 X() Z
14*LBL *REQ*
15 XROM *ZYXdHA*
16 360
17 MOD
    
```

```

18 15
19 /
20 24
21 -
22 ABS
23 HMS
24 *ad=*
25 ARCL X
26 AVIEW
27 X()Y
28 HMS
29 *d=*
30 ARCL X
31 AVIEW
32 END
    
```

```

01*LBL *EEO*
02 30
03 PSIZE
04 *e=?*
05 PROMPT
06 STO 29
07 *LO(DMS)=?*
08 PROMPT
09 HR
10 *b(DMS)=?*
11 PROMPT
12 HR
13 .9832965
14 XROM *LBRYZX*
15 RDN
16 X()Y
17 R#
18 XEQ *REQ*
19 END
    
```

```

01*LBL *DIF*
02 XEQ *ASTRE*
03 STO 04
04 X()Y
05 STO 03
06 XEQ *ASTRE*
07 RCL 04
08 RCL 03
09 R#
10 HMS-
11 ABS
12 RDN
13 HMS-
14 ABS
15 *d/ad=*
16 ARCL Z
17 AVIEW
18 *d/dec=*
19 ARCL X
20 AVIEW
21 END
    
```

LAVILLE Laurent (T.634)
45 Rue Riquet
75019 PARIS

NH HN

Cher Jean-Daniel,

Je pense que le PFC ROM, qui reste le chef d'oeuvre de PFC USA, est une source de sujet intarissable. Je le prouve aujourd'hui en vous proposant une version optimisée des programmes NH et HN. Mais avant d'entrer dans le corps du sujet je ferai un rappel sur un de mes précédents articles concernant le programme F?, qui a été publié dans MR n°6p28. Quand je l'ai écrit, je ne connaissais pas le programme S? qui donne l'adresse absolue décimale du registre contenant le .END. et qui permet de réduire F? de cinq octets. Je ne connais pas la version incorporée au PFC ROM, aussi utilise-je donc une version différente et sans doute plus courte rendue possible grâce au module X-F. En voici la liste:

01	LBL "E?"	06	ASHF	11	5I2
02	RCL c	07	RDN	12	MOD
03	"*	08	256	13	ATOX
04	X<Y	09	ATDX	14	+
05	STO N	10	*	15	END

Revenons à F?. Il suffit donc de supprimer les lignes deux à six ainsi que la huit et de remplacer cette dernière par XEQ "E?", c'est tout.

Après ce petit détour arrivons-nous au premier sujet de ma lettre. Il est à n'en point douté que les versions NH et HN que je vous présente sont de loin des versions réduites si j'en juge par celles publiées dans Calculator Tips & Routines page 21. Néanmoins il subsiste un petit défaut. Les nombres autre que le MNN (Nombre Non Normalisé) présents dans la pile avant l'appel d'une des deux routines sont perdus irrémédiablement. Cela peut être très désagréable dans certaines situations.

Deux petites précisions à vous signaler avant de vous livrer la liste ligne à ligne des deux routines. La première est que NH utilise le s/PRGM QR (Quotient and Remainder) du PFC ROM. Là pas de problème. Par contre, là où il risque d'y en avoir un c'est si l'on décide d'utiliser séparément les deux s/PRGMs car ils utilisent chacun le sous-programme numéroté trois. Aussi attention.

01	LBL "NH"	15	+	29	55	43	GTO 04
02	CLA	16	XTDA	30	FC?C 09	44	X<Y
03	STO M	17	RDN	31	48	45	I6
04	7	18	FS?C IO	32	RTN	46	*
05	CF 09	19	GTO 02	33	LBL "HN"	47	+
06	LBL 01	20	DSE X	34	7	48	XTDA
07	ATOX	21	GTO 01	35	CF 09	49	RDN
08	I6	22	CLST	36	SF IO	50	SF IO
09	XEQ "QR"	23	RTN	37	LBL 04	51	DSE X
10	X<Y	24	LBL 03	38	ATOX	52	GTO 04
11	SF IO	25	X<Y?	39	60	53	RCL M
12	LBL 02	26	SF 09	40	XEQ 03	54	END
13	9	27	RDN	41	-		
14	XEQ 03	28	FS? 09	42	FS?C IO		

Merci de votre éventuelle attention et

Bonne programmation à tous.

M^r UNG SENG POR
5 rue Duvergier
75019 PARIS

à Paris le 2/4/85

GRAPHE

Cher Monsieur Dodin

Suite à votre lettre de rappel, je vous signale que je ne souhaite plus renouveler mon abonnement car je viens d'acquérir un micro-ordinateur et par manque de temps j'ai laissé mon HP41CV de côté. Mais je tiens à remercier beaucoup le club, car il m'a fait découvrir pas mal de possibilités de ma petite machine, hélas ses périphériques sont très coûteuses. Pour témoigner ma reconnaissance, je vous envoie ci-joint 3 cartes magnétiques contenant 1 programme composé de 2 parties. C'est un programme sur l'application de la théorie des graphes. Il s'agit en fait de l'application de l'algorithme de Marshall généralisé à la détermination de certains chemins optimaux. Ce programme permet de déterminer la distance minimale entre 2 sommets. Vous trouverez le mode d'emploi et quelques explications sur une feuille ci-jointe. Pour plus de détails, je vous conseille de consulter le livre "Techniques numériques appliquées au calcul scientifique" par J.P. PELLETIER édition MASSON à partir de la page 237. J'espère que nos membres trouveront d'autres utilisations de ce programme dans leur vie quotidienne. Si un problème se pose, vous pouvez m'envoyer un petit mot, j'essayerai d'y répondre. Avant de vous quitter, je souhaite une longue vie à PPC-T et joyeuse programmation à nos membres.

Veuillez agréer, Monsieur, l'expression de ma considération distinguée.

Toutes mes amitiés (T 286)



DM1 et 2 (Distance Minimale)

Le programme est constitué de 2 modules indépendants mais complémentaires : DM1 et DM2.

DM1 est un module de saisie et d'affichage des résultats.

DM2 est un module de calcul dont les données nécessaires au calcul sont fournies par DM1.

Pour augmenter la capacité mémoire libre pour le traitement, vous pouvez d'abord charger "DM1", vous faites la saisie, vous l'effacez et chargez "DM2" pour le calcul et vous l'effacez et finalement rechargez "DM1" pour lire les résultats.

Le programme utilise 2 matrices d'adjacence :

- 1) pour indiquer le chemin existant entre 2 sommets
- 2) pour indiquer sa distance

Mode d'emploi :

- XEQ "DM1", vous êtes maintenant en mode USER
- La machine vous demande "nb de SOMMETS?" et votre réponse devra être un entier positif > 1 sinon la machine vous redemande.

Par la suite, pour changer le nb de SOMMETS, il suffit d'appuyer sur **[A]** en mode USER.

- La machine vous demande "SOMMET Y X?"

Vous devrez entrer d'abord le sommet de départ suivi de **[ENTER]** puis le sommet d'arrivée et **[R/S]**. Si les données sont erronées, la machine vous redemande. Si c'est correcte, elle vous demande "DIST. = ?".

Vous devrez entrer un nb positif ; si vous appuyez sur **[R/S]** sans rien entrer, on quitte alors le programme de saisie en conservant l'ancienne information.

Pour faire la suite de la saisie ou la correction de la distance, il suffit d'appuyer sur **[B]** en mode USER.

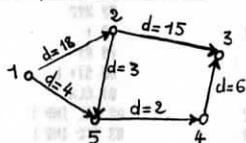
- Appuyer sur **[C]** en mode USER pour vérifier la saisie ou bien lire les résultats si on a fait avant XEQ "DM2".

Remarques : • Au départ, après un XEQ "DM1", si on répond correctement à la demande de "nb de SOMMETS?", la machine initialise les 2 matrices d'adjacence.

Dans le cas contraire ou sans réponse (en appuyant sur **[R/S]** seulement) les anciennes valeurs sont conservées.

- Les sommets doivent être numérotés de 1 à nb de SOMMETS.

Exemple d'utilisation :



XEQ "DM1"

nb de SOMMETS ? Réponse : 5, **[R/S]**

SOMMET Y^X ? Réponse : 1 **[ENTER]** 2 **[R/S]**

DIST. = ? Réponse : 18 **[R/S]**

...

SOMMET Y^X ? Réponse : 5 **[ENTER]** 4 **[R/S]**

DIST. = ? Réponse : 2 **[R/S]**

SOMMET Y^X ? Réponse : **[R/S]** pour quitter la saisie et on entend un BEEP sonore.

XEQ "DM2" pour le calcul

Dès que la calcul est terminé, on fait XEQ "DM1"

nb de SOMMETS ? Réponse : appuyer sur **[C]** en mode USER pour voir les résultats

On veut par ex. savoir le chemin minimal entre 2 et 3 :

SOMMET Y^X ? Réponse : 2 **[ENTER]** 3 **[R/S]**

DIST. = 11 c'est la distance minimale entre 2 et 3.

Faites **[R/S]** pour avoir les sommets rencontrés

2 **[R/S]**

5 **[R/S]**

4 **[R/S]**

3 **[R/S]**

et on entend un BEEP sonore qui indique la fin

Pour un autre chemin, il suffit d'appuyer sur **[C]**.

Suggestion : • Par ex., à Paris, vous voulez vous rendre d'une station à un autre et il y a plusieurs possibilités et vous connaissez la durée du trajet entre les stations. Alors ce programme vous permet d'avoir le chemin qui prend le moins de temps en vous indiquant la durée totale et les stations rencontrées. Il faut numéroté au préalable les stations qu'on a besoin et la distance sera remplacée par la durée.

- Par ex. pour ses grandes vacances, vous partez en voiture et vous voulez savoir le trajet optimal.

Bien sûr, le trajet ne sera pas en pratique optimal car il faut tenir compte de l'embouteillage, de la vitesse qu'on roule, de l'état des routes, du météo, etc...

PRP "DM1"	55*LBL 01		PRP "DM2"	
01*LBL "DM1"	56 RDW		01*LBL "DM2"	60 +
02 "DIST. MINIMUM"	57 CF 22		02 RCL 00	61 FC?C 25
03 AVIEW	58 PROMPT	116*LBL C	03 E3	62 XEQ 06
04 SF 27	59 FC?C 22	117 XEQ 02	04 /	63 RCL IND Z
05 CF 29	60 GTO 06	118 FC?C 22	05 ISC X	64 X=Y?
	61 X<0?	119 GTO 06	06 **	65 GTO 04
06*LBL A	62 GTO 01	120 RCL IND L	07 STO I	66 X<Y
07 CLST	63 X=0?	121 RCL d	08 STO \	67 STO IND T
08 "Nb SOMMETS ?"	64 RCL 01	122 FIX 2	09 STO J	68 RT.
09 CF 22	65 STO IND L	123 ARCL Y		69 RT
10 PROMPT	66 CLX	124 STO d	10*LBL 00	70 RCL \
11 FC?C 22	67 RCL 00	125 CLX	11 RCL I	71 RCL J
12 GTO 06	68 ST* X	126 9,999999999 E99	12 FRC	72 -
13 SIGN	69 ST+ L	127 X=Y?	13 ISC X	73 Rt
14 LASTX	70 RDW	128 GTO 06	14 **	74 X<Y
15 X<=Y?	71 STO IND L	129 PROMPT	15 STO I	75 -
16 GTO A	72 GTO B	130 Rt	16 SIGN	76 RCL 00
17 FRC		131 Rt		77 X+2
18 X=0?	73*LBL 02		17*LBL 01	78 +
19 GTO A	74 CLST	132 "SOMMETS RENCONT"	18 RCL I	79 Rt
20 LASTX	75 "SOMMETS YX?"	133 "FRES"	19 RCL J	80 ST+ L
21 CLRG	76 CF 22	134 AVIEW	20 X=Y?	81 CLX
22 ENTER+	77 PROMPT	135 X<Y	21 GTO 02	82 RCL IND Y
23 X+2	78 FC? 22		22 RCL 00	83 STO IND L
24 2	79 RTN	136*LBL 05	23 ST+ T	
25 X<Y	80 XEQ 03	137 RCL d	24 Rt	84*LBL 04
26 *	81 FC?C 25	138 FIX 0	25 GTO 05	85 Rt
27 ISC X	82 GTO 02	139 CLA	26*LBL 06	86 Rt
28 **	83 X<Y	140 ARCL Y	27 X<Y?	87 ISC Y
29 SF 25	84 XEQ 03	141 STO d	28 X<Y	88 **
30 RCL IND X	85 FC?C 25	142 RDW	29 RDY	89 ISC \
31 FS?C 25	86 GTO 02	143 PROMPT	30 RTN	90 GTO 03
32 GTO 07	87 XEQ 04	144 X=Y?		91 X<Y
33 "SET SIZE "	88 "DIST.=="	145 GTO 06	31*LBL 02	
34 RCL d	89 RTN	146 XEQ 04	32 -	92*LBL 05
35 FIX 0		147 RCL 00	33 RCL 00	93 ISC I
36 ARCL Y	90*LBL 03	148 ST* X	34 *	94 GTO 01
37 STO d	91 CF 25	149 ST+ L	35 RCL \	95 ISC J
38 CLST	92 FRC	150 RDW	36 FRC	96 GTO 00
39 PROMPT	93 X=0?	151 RCL IND L	37 ISC X	97 CLA
40 GTO A	94 RTN	152 GTO 05	38 **	98 CLST
	95 X<Y L		39 GTO \	99 END
41*LBL 07	96 X<=0?	153*LBL 06	40 RDW	
42 X<Z	97 RTN	154 CLA		
43 STO 00	98 RCL 00	155 CLST	41*LBL 03	
44 9,999999999 E99	99 X<Y?	156 BEEP	42 RCL \	
	100 RTN	157 .END.	43 RCL J	
45*LBL 00	101 RDW		44 X=Y?	
46 STO IND L	102 X=Y?		45 GTO 04	
47 DSE L	103 RTN		46 -	
48 GTO 00	104 SF 25		47 Rt	
	105 RTN		48 X<Y	
49*LBL B			49 -	
50 XEQ 02	106*LBL 04		50 RCL Y	
51 FC?C 22	107 STO Z		51 Rt	
52 GTO 06	108 DSE X		52 ST- Y	
53 "I?"	109 **		53 RDW	
54 Rt	110 RCL 00		54 CHS	
	111 *		55 Rt	
	112 X<Y		56 Rt	
	113 +		57 RCL IND T	
	114 X<Y L		58 RCL IND T	
	115 RTN		59 SF 25	

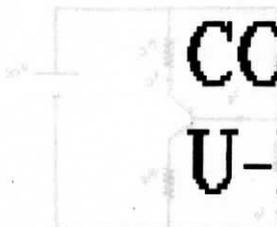
ELECTRONIQUE

REDACTEUR EN CHEF: J. L. LAFITE

EDITEUR: J. LAFITE
10, rue de Valenciennes
92000 NANTERRE

LEADER ELECTRONIQUE DE FRANCE

Le leader électronique de France est un magazine qui se consacre à l'actualité de l'électronique de puissance et de l'électronique de commande. Il est destiné à tous les professionnels de ces domaines et à tous les amateurs éclairés. Le leader électronique de France est un magazine qui se consacre à l'actualité de l'électronique de puissance et de l'électronique de commande. Il est destiné à tous les professionnels de ces domaines et à tous les amateurs éclairés.



COIN DES U-PROS

ELECTRONIQUE

JEGOUZO Christian
5, rue de l'ardèche
44800 St HERBLAIN

NANTES, le 1 JANVIER 1986

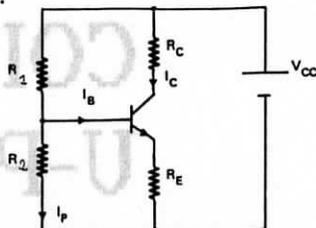
DOSSIER ELECTRONIQUE DE BASE

Les 4 programmes que je vous propose, devraient vous permettre de vérifier vos calculs en matière d'électronique de base. J'ai essayé de garder les memes spécifications dans les 5 programmes afin de faciliter leur emploi.

P-PB : Polarisation par Pont de Base

regs pgm : 223 octets
regs données : 8

La polarisation par pont diviseur de tension fait partie des polarisations classiques. Ce montage consomme de l'énergie, meme au repos à cause du pont. Le courant I_B est fixe définitivement par les valeurs de R_1 et R_2 .



Connaissant I_{C0} , V_{CE0} , V_{BE0} , $BETA$, V_{cc} , les rapports $X=R_C/R_E$ et $Y=I_P/I_B$, le programme donne I_B , I_P , R_1 , R_2 , $R_B=R_1//R_2$.

Exemple :

"Ic0 ?"	E-3	R/S (1mA)
"VCE0 ?"	5	R/S (5 volts)
"VBE0 ?"	0,6	R/S (0,6 volts)
"BETA ?"	100	R/S (BETA=H21)
"Vcc ?"	10	R/S (Tension d'alimentation)
"RE ?"		R/S (mettre 0 si RE n'existe pas)
"RC=X.RE X?"	4	R/S
"RE=1000"		R/S (RE 1 kohms)
"RC=4000"		R/S (RC 4 kohms)
"Ib=10 µA"		R/S (10 micro-ampères)
"IP=Y.Ib Y?"	10	R/S
"IP=100 µA"		R/S (courant dans R2 100 micro-ampères)
"R2=16100"		R/S (16,1 kohms)
"R1=76272"		R/S (76,2 kohms)
"RB=13294"		(R1//R2 = 13,2 kohms)

TR-Ec : Transistor monté en Emetteur Commun

regs pgm : 895 octets
regs données : 22

Le programme se décompose en 4 grandes options :

A : Initialisation

B,C,D,E : Calcul des amplifications en tension (A_v) et en courant (A_i), ainsi que des impédances d'entrée (Z_e) et de

sortie (ZS).

F : Détermination des équations des droites d'attaque et de charge statiques et dynamiques.

G : Calcul des coefficients de stabilité thermique

Exemple :

A : R1^R2 ? 76.4 E3 ^ 16 E3 R/S (si R/S la machine demande RB)

RC^RE^RU ? 4 E3 ^ E3 ^ 4 E3 R/S (en ohms)

H11 ? 1000 R/S

H21 ? 100 R/S

H22 ? E-5 R/S (H12 toujours pris égal à 0)

OK

a : Permet de changer la valeur de H22

H22 ? 0 R/S

OK

B,C,D,E : Les 4 options prennent en compte la valeur de H22, ainsi que la valeur de la résistance d'émetteur de couplage.

3 cas sont possibles :

- Condensateur aux bornes de RE : RE de couplage = 0 = RE1
- Condensateur partiellement en // avec RE : RE couplage = RE1
- Pas de condensateur : RE = RE1

B : RE COUPL ?	H22<>0 R/S	H22=0 R/S	Couplage 222,00 H22=0 (forcée)
AV=	-196,08	-200,00	-8,54
C: AI=	45,57	46,49	18,05
D: Ze=	929,72	929,72	8454,24 (ohms)
E: ZS=	3846,15	4000,00	4000,00 (ohms)

F : R1^R2 ? 76,4 E3 ^ 16 E3 R/S (demande si valeurs non entrées

BETA ? 100 R/S dans option A)

Ic0 ? E-3 R/S (1mA)

VBE0^VCE0 ? 0,6 ^ 5 R/S

Vcc ? 10 R/S

Vbb = 1,73 R/S (potentiel de la base)

M0 VBE = 1,73 R/S (droite d'attaque statique)

M0 Ib = 0,00 µA R/S (couple M0,M1)

M1 VBE = 0,00 R/S

M1 Ib = 15,29 µA R/S

D CHAR STAT0 R/S (droite de charge statique)

M2 VCE = 10,00 R/S (couple M2,M3)

M2 Ic = 0,00 MA R/S (milliampères)

M3 VCE = 0,00 R/S (volts)

M3 Ic = 2,00 MA R/S (milliampères)

P0 VCE0 Ic0 (rappelle coordonnées du point de fonctionnement)

D CHARG DYN R/S (droite de charge dynamique)

M4 VCE = 7,00 R/S (couple M4,M5)

M4 Ic = 0,00 MA R/S

M5 VCE = 0,00 R/S (volts)

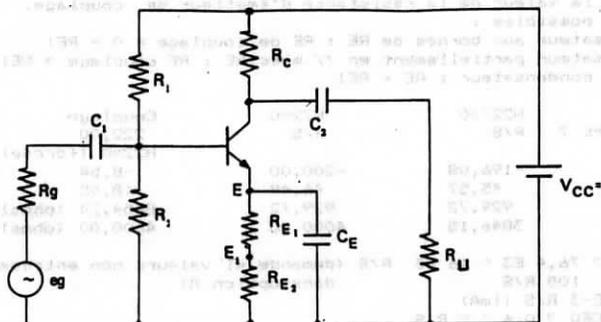
M5 Ic = 3,50 MA R/S (milliampères)

EXCURS MAXI (amplitudes maximales >0 et <0 des

VCE+2,00 -5,00 grandeurs de sortie. Elles sont

$I_c +2,50 -1,00$ dissymétriques)
 EXCURS SYM R/S (Pour obtenir des excursions symétriques
 $M_6 VCE= 2,86$ R/S il faut placer le point de repos au
 $M_6 I_c= 1,43$ MA R/S milieu de la droite de charge dynamique
 $I_{b1}= 14,29$ μA R/S On détermine alors les nouvelles coor-
 GRAPH $V_{BE1} ? 0,65$ R/S données M_6 du point de repos et la
 $I_{P1}= 89,40$ μA R/S nouvelle valeur de I_{b1} . Pour cela on
 $R'2= 23250,78$ fait varier R_2 d'ou $R'2$.)

G : $S1=dI_c/dI_{cb0}$ R/S (rappel de la formule)
 $S1= 12,58$ R/S
 $S2=dI_c/dV_{BE}$ R/S
 $S2=-8,75E-4$ R/S
 $S3=dI_c/dBETA$ R/S
 $S3=1,26E-6$



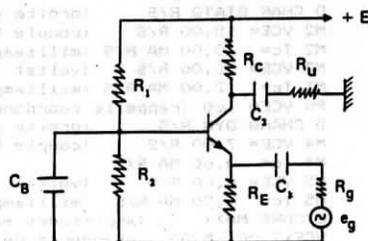
TR-Bc : Transistor monté en base commune

regs pgm : 448 octets
 regs données : 10

On considère que le transistor est utilisé en base fréquence. Les condensateurs sont des court-circuits. On transforme le générateur de Norton en générateur de Thévenin dépendant de V_{BE} et $e=\mu V_{BE}$. ($\mu=H21/H11.H22$)

Le programme donne l'amplification en tension (A_v), en courant (A_i), les impédances d'entrée (Z_e) et de sortie (Z_s)

Les options b,c,d,e rappelle les formules (utiliser le pas à pas SST).



Exemple :

A : RC? 3335 R/S (ohms)
RE? 667 R/S
RG? 50 R/S
RU? 3335 R/S
H11? 800 R/S
H21? 80 R/S
H22? 5 E-5 R/S
 $\mu=2000$
 $\mu \& H21/H11$ ($\&=1/H22$)

a : pour changer H22

B : AV= 153,99

b : formule par SST $Av=(\mu+1) RL/\&+RL$ $RL=RC//RU$

C : AI= -0,49

c : $Ai=(-RC/RC=RU) (1/(1+(\&+H11+RE) . (\&+RL)/RE(u+1) . H11$

D : Ze= 10,52 (ohms)

d : $1/Ze= 1/RE + (u+1/\&+RL) + 1/H11$

E : ZS= 3235,06 (ohms)

e : $1/ZS= 1/RC + (1/(\&+(u+1)R0))$ $R0=RG//RE//H11$

TR-Cc : Transistor monté en collecteur commun

regs pgm : 201 octets

regs données : 7

Comme pour les autres montages le pg calcule Av, Ai, Ze et Zs. On remarquera que l'étage collecteur commun n'amplifie pas en tension mais amplifie en courant, et possède une résistance d'entrée assez forte et une faible impédance de sortie. Il peut-être utilisé comme étage adaptateur d'impédances.

Exemple :

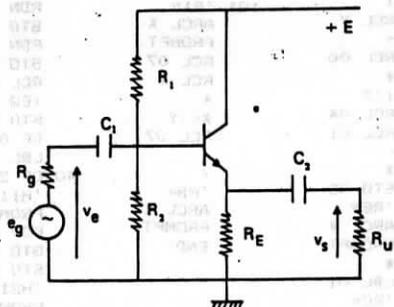
A : R1^R2 ? R/S
RB ? 23700 R/S (demande si R1 et R2 non entrés)
RU^RE ? 5 E3 ^ 5 E3 R/S (ohms)
RG ? 50 R/S (ohms)
H11^H21 ? 1200 ^ 100 R/S
OK

B : AV= 0,995 R/S

C : AI= 4,718 R/S

D : Ze= 21675,16 R/S (ohms)

E : ZS= 12,34 (ohms)



	*		FS? 22
	'IP=		GTO 00
	ARCL X		RCL 10
	'uA		70 SF 25
	70 PROMPT		1/X
	E6		RCL 07
	/		FC? 00
1 LBL P-PB	1/X	1 LBL TR-EC	XEQ 20
FIX 0	RCL 03	SIZE?	RCL 08
'Ic0 ?	1	22	/
PROMPT	+	X>Y?	LBL 01
STD 00	RCL 05	PSIZE	RCL 09
'VCE0 ?	*	LBL A	*
PROMPT	RCL 06	0	
STD 01	*	X<>F	80 CHS
'VBE0 ?	80 *	SF 02	'AV=
10 PROMPT	RCL 02	FIX 2	ARCL X
STD 02	+	CF 22	PROMPT
'BETA ?	*	'R1^R2 ?	GTO B
PROMPT	STD 07	PROMPT	LBL 00
STD 03	'R2=	FS? 22	STD 11
'Vcc ?	ARCL X	GTO 06	SF 01
PROMPT	PROMPT	'RB ?	'H22=0
STD 04	RCL Y	PROMPT	AVIEW
'RE ?	*	STD 03	90 PSE
PROMPT	90 RCL 06	GTO 07	RCL 09
20 X=0?	*	GTO 07	1
GTO 00	CHS	LBL 06	+
RCL 04	RCL 04	CF 02	*
RCL 01	+	STD 02	RCL 08
-	X<>Y	X<>Y	+
RCL 00	1	STD 01	1/X
/	+	XEQ 20	RCL 07
GTO 01	RCL 06	STD 03	*
LBL 00	*	LBL 07	100 GTO 01
'RC=X.RE X?	/	'RC^RE^RU ?	
30 PROMPT		PROMPT	
1	101 'R1=	30 STD 06	101 LBL C
RCL Y	ARCL X	RDN	FS? 01
+	PROMPT	STD 05	GTO 02
RCL 00	RCL 07	RDN	RCL 03
*	RCL Y	STD 04	RCL 08
1/X	*	RCL Z	+
RCL 04	X<>Y	XEQ 20	1/X
RCL 01	RCL 07	STD 07	FS? 00
-	+	SF 03	GTO 03
40 *	/	LBL J	110 RCL 10
STD 05	'RB=	CF 22	1/X
'RE=	ARCL X	'H11 ?	RCL 07
ARCL X	PROMPT	PROMPT	RCL Y
PROMPT	END	FC? 22	+
*		GTO F	/
LBL 01		STD 08	*
'RC=		'H21 ?	GTO 03
ARCLX		PROMPT	LBL 02
PROMPT		STD 09	RCL 09
50 RCL 00		LBL a	120 1
		50 CF 00	+
		51 'H22 ?	RCL 11
		PROMPT	*
		STD 10	RCL 08
		X=0?	+
		SF 00	RCL 03
		CF 03	+
		'OK	1/X
		PROMPT	LBL 03
		GTO A	RCL 07
		60 LBL B	130 RCL 06
		FS? 03	/
		GTO J	RCL 09
		CF 01	*
		CF 22	RCL 03
		'RE COUPL ?	*
		PROMPT	*
			'AI=

				*EXCURS SYM	ARCL Y
	201	PROMPT		PROMPT	PROMPT
		'D ATTO STATQ		SF 04	'M
140		CF 04		RCL 15	FIX 0
		0		RCL 07	ARCL X
		RCL 16		*	FC? 04
		0		LAST X	
		XEQ 21		RCL 04	351 'I Ib=
		RCL 16		+	FS? 04
		RCL 00	280	RCL 05	'I Ic=
	210	RCL 05		+	RCL Z
		*		/	FS? 04
		RCL 03		RCL X	E3
		+		RCL 07	FC? 04
150	RCL 11	/		/	E6
		0		STD 19	*
		1		X<>Y	360 FIX 2
151	*	XEQ 21		6	ARCL X
	+	'D CHAR STATQ	290	XEQ 21	FC? 04
		PROMPT		E3	'I uA
		SF 04	220	*	FS? 04
		0		RCL 00	'I MA
		RCL 15		/	PROMPT
		2		'Ib1=	RTN
		XEQ 21		ARCL X	LBL G
160	RCL 10	RCL 15		STD 20	'S1=dIc/dIcb0
	SF 25	RCL 04		'I uA	370 PROMPT
	1/X	RCL 05		PROMPT	RCL 05
		+	300	'GRAPH VBE1 ?	RCL 00
	RCL 04	/	301	PROMPT	1
	FS? 00	0		RCL 05	+
	GTO 05	3		RCL 19	RCL Y
	FS? 01	XEQ 21		*	RCL Y
	GTO 05	'PO VCEO Ic0		+	*
	XEQ 20	PROMPT		RCL X	RCL 03
	LBL 05	'D CHARG DYN		CHS	+
170	'ZS=	PROMPT		RCL 15	380 LAST X
	ARCL X	RCL 07		+	R^
	PROMPT	RCL 12	310	RCL 01	+
	GTO E	*		/	RCL Z
	LBL F	240	STD 17	E6	*
	FS? 02	RCL 14		*	X<>Y
	GTO A	+		RCL 20	/
	BETA ?	0		-	STD 21
	PROMPT	X<>Y		'IP1=	'S1=
	STD 00	4		ARCL X	ARCL X
180	'Ic0 ?	XEQ 21		'I uA	390 PROMPT
	PROMPT	RCL 14		PROMPT	RCL 00
	STD 12	RCL 07	320	/	'S2=dIc/dVbe
	VBE0*VCEO ?	/		E6	PROMPT
	PROMPT	250	STD 18	*	RCL 05
	STD 14			'R*2=	RCL Y
	RDN	251	RCL 12	ARCL X	1
	STD 13	+		PROMPT	+
	Vcc ?	0		GTO F	*
	PROMPT	5		LBL 20	RCL 03
190	STD 15	XEQ 21		RCL Y	400 +
	RCL 02	'EXCURS MAXI		RCL Y	401 /
	*	PROMPT		+	CHS
	LAST X	E3	330	RDN	'S2=
	RCL 01	ST*18		*	ARCL X
	+	ST*12		R^	PROMPT
	/	'VCE+		/	'S3=dIc/dBETA
	STD 16	ARCL 17		RTN	PROMPT
	Vbb=	'I -		LBL 21	RCL 00
	FIX 2	ARCL 14		'M	RCL 05
200	ARCL X	PROMPT		FIX 0	410 RCL Y
		'IC +		ARCL X	*
		ARCL 18	340	FC? 04	RCL 03
		'I -		'I VBE=	+
		ARCL 12		FS? 04	RCL Y
		ST/12	270	'I VCE=	*
		PROMPT		FIX 2	X<>Y

	51	CF 00	*	XEQ 20
X^2		LBL 06	LBL 05	RCL 05
RCL 05		'OK	'AI=	XEQ 20
RCL 03		PROMPT	ARCL X	RCL 06
420 +		GTO A	PROMPT	200 *
/		LBL 20	GTO C	
1/X		STD Z	LBL 04	201 RCL 08
RCL 12		RCL Y	130 RCL 05	1/X
*		+	RCL 02	+
RCL 16	60	1/X	+	1/X
+		RDN	LAST X	RCL 01
RCL 13		*	/	1/X
-		R^	RCL 07	+
/		/	/	1/X
430 1/X		RTN	1	LBL 10
RCL 21		LBL B	+	210 'ZS=
*		FS? 00	-2	ARCL X
'S3=		GTO 02	140 *	PROMPT
ARCL X		RCL 06	1/X	LBL 09
PROMPT	70	RCL 08	GTO 05	RCL 01
436 END		1/X	LBL c	GTO 10
		RCL 09	'-RC/RC+RU 1/	LBL e
1 LBL TR-Bc		+	AVIEW	'1/ZS=1/RC +1
LBL A		LAST X	'1+ H11+RE&&+	AVIEW
0		/	AVIEW	'/&+ μ +1.R0
X<>F		1/X	RL/RE. μ +1.H11	220 AVIEW 20
'RC		*	PROMPT	'R0=RG/RE/H11
PROMPT		LBL 03	150 GTO c	PROMPT
STD 01		AV=	151 LBL D	END
'RC	80	FIX 2	FS? 00	
PROMPT		ARCL X	GTO 07	
10 STD 02		PROMPT	RCL 08	
'RG		GTO B	1/X	
PROMPT		LBL 02	RCL 09	
STD 03		RCL 07	+	
'RU		RCL 09	1/X	
PROMPT		*	RCL 06	
STD 04		RCL 05	160 *	
'H11		/	RCL 05	
PROMPT	90	GTO 03	1/X	
STD 05		LBL b	+	
20 'H21?		'AV= μ +1 RL/	RCL 02	
PROMPT		AVIEW	1/X	
STD 07		'&+RL	+	
RCL 04		PROMPT	1/X	
RCL 01		LBL C	LBL 08	
XEQ 20		FS? 00	'Ze=	
STD 09		GTO 04	170 FIX 2	
LBL a		RCL 06	ARCL X	
SF 00	100	RCL 05	PROMPT	
'H22?		*	LBL 07	
30 PROMPT	101	*	RCL 07	
STD 08		RCL 02	1	
X=0?		*	+	
GTO 06		1/X	RCL 05	
RCL 05		RCL 08	/	
RCL 08		1/X	RCL 02	
*		RCL 09	180 1/X	
1/X		+	+	
RCL 07	110	*	1/X	
*		110 RCL 02	GTO 08	
40 'p=		RCL 05	LBL d	
ARCL X		+	'1/Ze=1/RE +	
AVIEW		*	AVIEW	
PSE		1	'1/H11+ μ +1/	
1		1/X	AVIEW	
+		CHS	'&+RL	
STD 06		RCL 01	190 PROMPT	
μ =&H21/H11		RCL 04	LBL E	
AVIEW	120	+	FS? 00	
PSE		/	GTO 09	
50 PSE		RCL 01	RCL 03	
			RCL 02	

PROFIL

PROFIL DE VOIES URBAINES.

Voici les significations succinctes des termes employés et le N° du registre les concernant.

"A"... "CHOIX (1-3)?" entrer le numéro de la classe du revêtement désiré, pour voir le menu: "R/S";

"1" revêtements lisses: béton de ciment bétons bitumineux, granulats de synthèse asphalte coulé ou pavés d'asphalte;

"2" pavage en granit: mosaïques petites (0,10) ou grosses (0,14);

"3" chaussée pavée en échantillon.

Ce choix place "d" en RO1, d étant l'altitude d'un point appelé défense, posé à 1,00 mètre des bordures et formant le caniveau.

"R/S"... "LARGEUR?" entrer la largeur de la chaussée qui sera stockée en RO2.

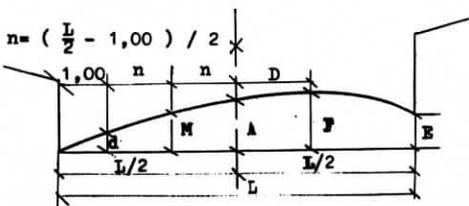
"R/S"... "EPAULEMENT?" E= différence des hauteurs des fils d'eau, stocké en RO3.

Les 3 éléments de calcul ainsi entrés permettent par "R/S" d'afficher D en RO5 c'est le DESAXEMENT du point haut de la voie, appelé "F" en RO4. Ce déport de la flèche permet de rétablir de part et d'autre deux demis profils normaux, en les traitant séparément comme tels.

"R/S"... affiche la hauteur "A" à l'axe de la chaussée calculée, et "M" étant la cote du point équidistant à "A" et "d".

Si les caniveaux sont de même hauteur la cote de A sera également celle de F.

Ces calcul sont admis au cahier des Clauses, Conditions et Charges Générales Imposées aux entrepreneurs des Travaux publics de la VILLE DE PARIS. (titre III du C.C.C.G.)



09:13	09/05		
01+LBL "PROFIL"	64 X<>Y	106 RCL 01	
02+LBL A	65 E	107 -	
03 FIX 3	66 -	108 625 E-3	
04+LBL 00	67 /	109 *	
05 CF 22	68 RCL 01	110 RCL 01	
06 "CHOIX(1-3)?"	69 4	111 +	
07 TONE 9	70 /	112 "H="	
08 PROMPT	71 *	113 ARCL X	
09 FS?C 22	72 ST+ 04	114 PROMPT	
10 GTO IND X	73 ARCL 04	115 GTO A	
11 AON	74 PROMPT	116+LBL 05	
12 "1-ASPHALTE"	75 RCL 02	117 "E. MAX="	
13 PSE	76 2	118 ARCL 03	
14 "2-MOSAIQUE"	77 /	119 PROMPT	
15 PSE	78 RCL 05	120 END	
16 "3-PIERRE"	79 -		
17 PSE	80 X12		
18 OFF	81 RCL 02		
19 GTO 00	82 RCL 05		
20+LBL 01	83 2		
21 5 E-2	84 *		
22 GTO 04	85 -		
23+LBL 02	86 E		
24 6 E-2	87 -		
25 GTO 04	88 /		
26+LBL 03	89 RCL 05		
27 72 E-3	90 X12		
28+LBL 04	91 RCL 05		
29 STO 01	92 2		
30 TONE 9	93 *		
31 "LARGEUR?"	94 RCL 02		
32 PROMPT	95 +		
33 STO 02	96 E		
34 2	97 -		
35 *	98 /		
36 5	99 -		
37 -	100 RCL 01		
38 X<>Y	101 *		
39 4	102 RCL 03		
40 /	103 +		
41 *	104 "A="		
42 STO 03	105 ARCL X		
43 TONE 9			
44 "EPAULEMENT?"			
45 PROMPT			
46 X<>Y?			
47 GTO 05			
48 STO 04			
49 STO 03			
50 "B="			
51 RCL 01	XEQ A		
52 /	CHOIX(1-3)?	2,000 **	
53 STO 05			
54 ARCL 05	R/S		
55 "H="	LARGEUR?	14,000 **	
56 2			
57 *	R/S		
58 RCL 02	EPAULEMENT?		
59 X<>Y		0,310 **	
60 -	R/S		
61 ENTER	D=5,167 F=0,306		
62 STO 07	R/S		
63 X12	A=0,317 N=0,221		

BLAISE

UN NOUVEAU
LANGAGE

APPLICATIONS

UN NOUVEAU LANGAGE

BLAISE

Introduction

Cet article ouvre une nouvelle rubrique dans MICRO-REVUE, qui traitera des langages de programmation. Nous allons donc étudier comment définir un langage et réaliser un compilateur pour ce langage.

1^{er} épisode : la description syntagmatique d'un langage.

Nous allons nous intéresser pour l'instant uniquement à la description de la syntaxe d'un langage : le niveau sémantique n'intervient pratiquement que dans l'étape finale, la réalisation du compilateur.

Pour décrire un langage, nous allons avoir besoin d'un autre langage, un métalangage. Nous utiliserons ici la notation BN* (Backus Naur form), valable principalement pour les langages "classiques" et structurés, donc inappropriée pour le forth ou le basic. Sa première utilisation fut pour définir l'Algol 60 : fortran, par exemple; avait les caractéristiques de son compilateur et non l'inverse.

Un langage L se définit alors par $L(T, N, P, S)$, T étant l'ensemble des éléments terminaux (par exemple GOTO), N l'ensemble des éléments non terminaux (les catégories grammaticales), P l'ensemble des productions (règles de grammaire) et S un élément de N, l'élément de départ.

On utilise pour décrire P les métasymboles suivants :

:= pour une production : A := γ signifie "réécrire γ par A".
pour le "ou"
<-> pour symboliser un non terminal
| - | pour une répétition incluant le séquence vide ϵ .
Prenons un exemple : soit le langage

<phrase> ::= < sujet > < verbe >
< sujet > ::= le chat | le chien
< verbe > ::= mange | dort

Les 4 phrases syntaxiquement correctes sont donc :

"le chat mange", "le chien mange", "le chat dort", "le **chien dort**".

On a $S = \langle \text{phrase} \rangle$, $T = \{ \text{le chien, le chat, dort, mange} \}$ et $N = \{ \langle \text{phrase} \rangle, \langle \text{sujet} \rangle, \langle \text{verbe} \rangle \}$.

Dans la suite, les majuscules désigneront les non terminaux, les minuscules les terminaux et les caractères grecs des séquences d'éléments de TUN (des éléments de TUN*).

On dit qu'une séquence η peut être directement générée à partir d'une séquence γ si il existe des séquences α, β, γ' telles que $\eta = \alpha \gamma \beta$, $\eta = \alpha \gamma' \beta$ et P contient $\gamma' := \eta$. On note alors $\gamma \rightarrow \eta$.

Une séquence η peut alors être générée à partir d'une séquence γ , si il existe η_1, η_2 telles que $\forall i \in [1..n], \eta_i \rightarrow \eta_{i+1}$. On note $\eta \xrightarrow{*} \eta_n$.

Nous pouvons maintenant définir le langage comme l'ensemble des séquences de terminaux générables à partir de S :

$$L = \{ \gamma / S \xrightarrow{*} \gamma \text{ et } \gamma \in T^* \} .$$

L'étape finale de la génération ne contient plus que des terminaux, d'où leur nom.

On dit qu'un langage est sensible au contexte si la substitution d'un élément non terminal tient compte des autres symboles présents, par exemple $\langle A \rangle ::= \alpha \gamma \beta$.

Au contraire, si les productions sont de la forme $A ::= \gamma$, il est dit hors contexte, ou non sensible au contexte. Il est très simple d'analyser les phrases d'un tel langage, pour vérifier si elles appartiennent effectivement au langage. On utilise l'analyse descendante, avec lecture d'un symbole à la fois (LL1) : on reconstruit la phrase à partir du symbole de départ. Par exemple :

< phrase >	le chien mange
< sujet >	le chien mange
le chien < verbe >	le chien mange
< verbe >	mange
mange	mange

Mais il faut prendre quelques précautions si l'on ne veut pas que l'analyse échoue par erreur d'aiguillage et éviter les retours en arrière : considérons le langage

$S ::= A|B$
 $A ::= xA|y$
 $B ::= xB|z$
 analysons xz :

S	xz
A	xz
xA	xz
A	z

et notre algorithme se plante. Le problème réside dans la première étape où il est impossible de décider entre A et B puisqu'on ne lit qu'un symbole à la fois. Une solution serait de remonter et de prendre B, mais on a dit qu'on ne voulait pas de retour en arrière. Il va donc falloir prendre des précautions dans les définitions des langages : énonçons la loi :

Si $A ::= x_1 | x_2 | \dots | x_n$
 on doit avoir $i \neq j \Rightarrow \text{first}(x_i) \cap \text{first}(x_j) = \emptyset$
 $\text{first}(x)$ est l'ensemble des symboles terminaux qui peuvent apparaître en première position dans les phrases dérivées de x .

Dans l'exemple précédent, redéfinir le langage par

$S ::= \epsilon | xS$
 $\epsilon ::= y | z$
 élimine tout problème.

Le langage suivant pose un autre problème :

$S ::= ax$
 $A ::= x|\epsilon$ où ϵ représente la chaîne vide.
 Si on analyse x, on obtient

S	x
Ax	x
xx	x
x	-

C'est le problème de la chaîne nulle. Pour l'éviter, énonçons :

Si $A \rightarrow \epsilon$
 il faut $\text{first}(A) \cap \text{follow}(A) = \emptyset$
 avec, en considérant toutes les productions P_i : $X_i ::= y_i A_i$
 $\text{follow}(A) = \cup_i \text{first}(y_i)$.

Ainsi, pour générer B, BB, BBB, ...

$A ::= B|AB$ viole la loi 1
 $A ::= \epsilon|AB$ viole la loi 2

On voit donc qu'il est impossible de faire des définitions récursives à gauche, seule la récursivité à droite est possible, c'est à dire $A ::= \epsilon|BA$, qui équivaut d'ailleurs à $A ::= \{B\}$.

Ces deux lois sont suffisantes pour éviter tout problème au cours de l'analyse.

Remarquons que, tout à l'heure, nous avons remodelé un langage pour lui faire respecter nos deux lois. Mais attention : considérons l'exemple suivant :

$S ::= A \mid S-A$

$A ::= a \mid b \mid c$

et $S' ::= A \mid A-S'$

On a $S \xrightarrow{*} a-b-c$ et $S' \xrightarrow{*} a-b-c$, mais le premier correspond en fait à $(a-b)-c$ et le second à $a-(b-c)$, ce qui est différent. Quand on définit un langage par sa structure syntaxique, il faut quand même penser à sa structure sémantique : la première régit la seconde.

Vous voyez maintenant pourquoi je vous disais que cette notation ne s'appiquait pas au forth : ce la donnerait quelque chose du genre $\langle \text{mot} \rangle ::= : \{ \langle \text{mot} \rangle \} ;$.

Dans le prochain numéro, je vous parlerai des diagrammes syntaxiques, une autre façon de décrire un langage.

T 366 Eric AUQUAT

2eme Episode

2^{eme} épisode : les diagrammes syntaxiques, réalisation d'un analyseur.

Une seconde représentation d'une grammaire est un diagramme syntaxique, ou diagramme de Conway. Cette représentation est beaucoup plus parlante que la notation BNF, mais ces deux systèmes sont tout à fait équivalents. Les lois de représentations sont les suivantes:

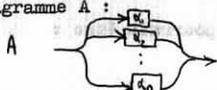
- Un symbole terminal x est représenté dans "un cercle, ou un ovale:



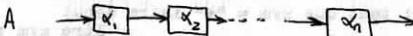
- Un symbole non terminal est représenté dans "un rectangle:



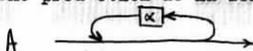
- Une production de la forme $A ::= \alpha_1 | \alpha_2 \dots \alpha_n$ est traduite par le diagramme A :



- Une production de la forme $A ::= \alpha_1 \alpha_2 \dots \alpha_n$ est traduite par



- Une production de la forme $A ::= \{ \alpha \}$ par



A partir de maintenant, il est aisé de construire un analyseur pour le langage ainsi défini : le diagramme syntaxique représente quasiment l'organigramme de l'analyseur.

Celui-ci, comme son nom l'indique, se contente d'analyser, c'est à dire de vérifier la syntaxe des programmes. Les langages les mieux adaptés pour cela sont les langages structurés : on décompose les diagrammes syntaxiques en blocs de petite taille, à chacun desquels va correspondre une procédure.

Le Pascal, par exemple convient très bien, ainsi que le Fortran : nous auront l'occasion d'y revenir. Pour le moment, nous utiliserons le langage nat'nel.

Supposons alors que nous ayons dans une variable sym le prochain symbole lu (nous travaillons en LL1), et que nous symbolisons par T(S) la procédure analysant le symbole non terminal S.

Une séquence $A ::= s_1 s_2 \dots s_n$ est traduite par
procédure A : T(S1) ; T(S2) ; ... ; T(Sn) fin A

Un choix $A ::= s_1 | s_2 | \dots | s_n$ par, avec first (s_i)
procédure A : selon que sym l1 faire T(S1)

...
sym l1 faire T(Sn)
autrement erreur fin A

Une itération $A ::= \{ S \}$ par
procédure A : tant que sym l faire T(S) fin A

Le traitement d'un terminal x est tout simplement
si sym = x alors lire symbole suivant sinon erreur.

5ème Episode

Et le traitement d'un non terminal consiste en l'appel de la procédure correspondante.

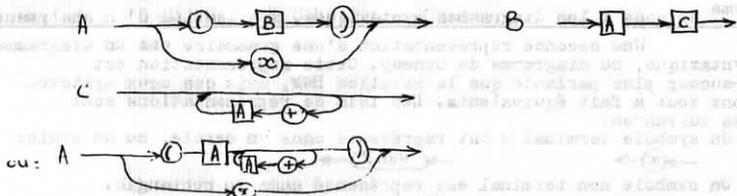
Prenons un exemple : le langage défini par

$A ::= x \mid (B)$

$B ::= AC$

$C ::= \{+a\}$

produisant par exemple $x, (x), (x \cdot x) \dots$ On a :



En général, sauf pour des exemples aussi simples, on préfère la première forme.

Un analyseur pour ce langage pourrait être :

procédure A

 si sym = "x" alors lire sym sinon

 si sym = "(" alors

 début

 lire sym ; A ; tant que sym = "+" faire début

 lire sym ; A

 fin;

 si sym = ")" lire sym sinon erreur

 fin

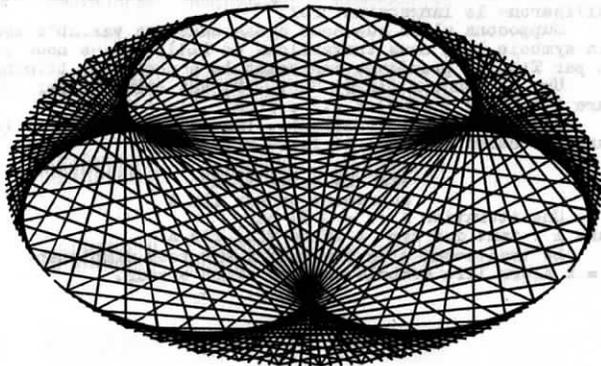
 sinon erreur

fin A.

Et le programme est lire sym ; A fin.

Dans le prochain numéro, nous passerons à la pratique : nous définirons un véritable langage, le Blaise, et nous écrirons un analyseur en Forth pour ce langage.

Tout ça Eric AUBOURG



3eme Episode

Enfin!

3^{ème} épisode : Le Blaise

Pour mettre en pratique les deux premiers épisodes, j'ai défini un langage inspiré de Pascal, le Blaise.

Il est assez aisé de vérifier sur les diagrammes syntaxiques que nos deux lois sont respectées. Pour écrire l'analyseur, j'ai choisi le format : ce langage s'y prête très bien. Pour pouvoir utiliser les routines du noyau, j'ai imposé la condition que tous les identificateurs soient séparés par au moins un espace (généralement, les compilateurs acceptent que l'on accole un signe de ponctuation et un identificateur, ceux-ci ne comportant alors que des caractères alphanumériques).

Le programme source doit être tapé dans un bloc buffer, l'usage de → étant admis. L'analyse est lancée par n PASCALBLAISE, n étant le numéro du premier écran du programme (comme LOGO).

Le code des erreurs est le suivant :

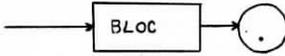
```
1 utiliser = au lieu de :=
2 nombre attendu après =
3 = attendu
4 identificateur attendu après const, var, procedure
5 , ou ; attendu
9 . attendu
12 variable attendue
13 := attendu
14 identificateur de procédure attendu
16 then attendu
17 ; ou end attendu
18 do attendu
20 opérateur relationnel attendu
22 ) attendu
23 variable attendue
25 ( attendu
```

Dans le prochain numéro, nous aborderons la génération du code objet, et nous écrirons un véritable compilateur pour le Blaise.

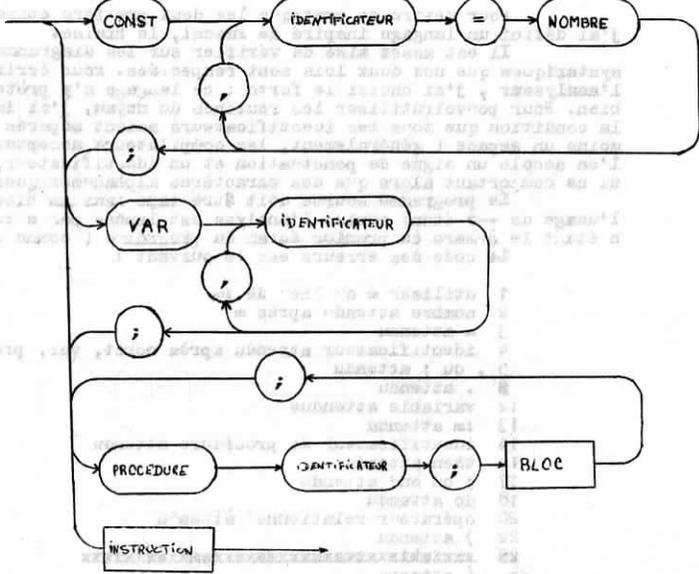


3ème Episode Enfin!

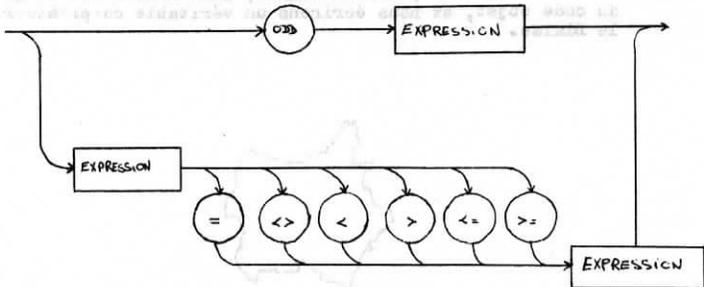
PROGRAMME



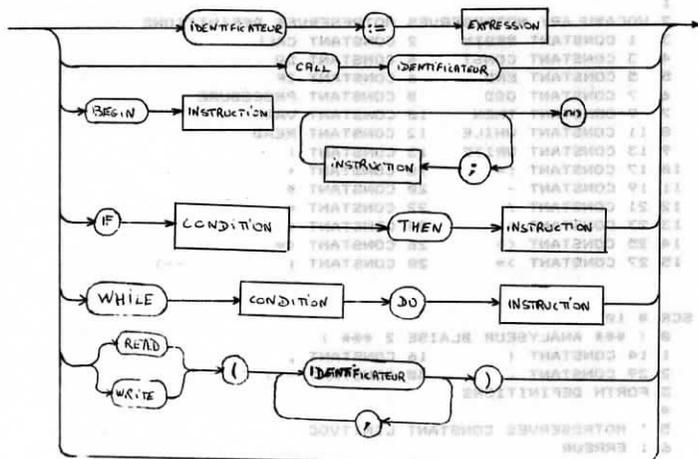
BLOC



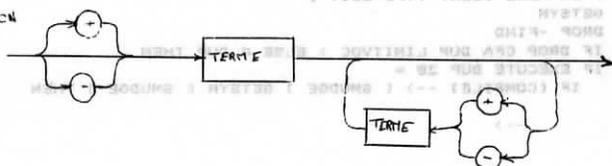
CONDITION



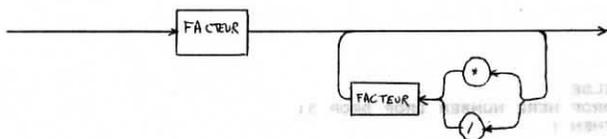
INSTRUCTION



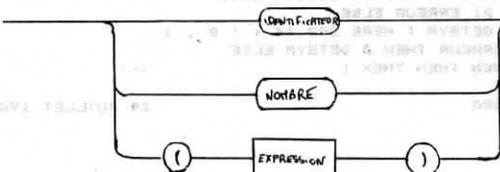
EXPRESSION



TERME



FACTEUR



1
SCR # 9

```
Ø ( *** ANALYSEUR BLAISE 1 *** )  
1  
2 VOCABULARY MOTRESERVES MOTRESERVES DEFINITIONS  
3 1 CONSTANT BEGIN 2 CONSTANT CALL  
4 3 CONSTANT CONST 4 CONSTANT DO  
5 5 CONSTANT END 6 CONSTANT IF  
6 7 CONSTANT ODD 8 CONSTANT PROCEDURE  
7 9 CONSTANT THEN 10 CONSTANT VAR  
8 11 CONSTANT WHILE 12 CONSTANT READ  
9 13 CONSTANT WRITE 15 CONSTANT )  
10 17 CONSTANT := 18 CONSTANT +  
11 19 CONSTANT - 20 CONSTANT *  
12 21 CONSTANT / 22 CONSTANT =  
13 23 CONSTANT > 24 CONSTANT <  
14 25 CONSTANT <> 26 CONSTANT <=  
15 27 CONSTANT >= 28 CONSTANT ! -->
```

SCR # 10

```
Ø ( *** ANALYSEUR BLAISE 2 *** )  
1 14 CONSTANT ( 16 CONSTANT ,  
2 29 CONSTANT . 30 CONSTANT -->  
3 FORTH DEFINITIONS  
4  
5 ' MOTRESERVES CONSTANT LIMITVOC  
6 : ERREUR  
7 ." ERREUR #" . BLK @ ." BLOC #" . IN @ 64 / ." LIGNE #"  
8 . CR HERE COUNT TYPE QUIT ;  
9 : GETSYM  
10 DROP -FIND  
11 IF DROP CFA DUP LIMITVOC > ELSE Ø DUP THEN  
12 IF EXECUTE DUP 30 =  
13 IF [COMPILE] --> [ SMUDGE ] GETSYM [ SMUDGE ] THEN  
14  
15 -->
```

SCR # 11

```
Ø ( *** ANALYSEUR BLAISE 3 *** )  
1  
2  
3  
4  
5  
6 ELSE  
7 DROP HERE NUMBER DROP DROP 31  
8 THEN ;  
9 Ø  
10 : FACTEUR  
11 DUP 30 > OVER 34 < AND IF GETSYM ELSE  
12 DUP 34 = IF 21 ERREUR ELSE  
13 DUP 14 = IF GETSYM [ HERE SP@ 14 + ! Ø , ]  
14 15 - IF 22 ERREUR THEN Ø GETSYM ELSE  
15 23 ERREUR THEN THEN THEN ; -->
```

T 366 ERIC AUBOURG

24 JUILLET 1985

```

SCR # 12
0 ( *** ANALYSEUR BLAISE 4 *** )
1 : TERME
2 BEGIN
3 FACTEUR DUP 20 = OVER 21 = OR
4 WHILE
5 GETSYM
6 REPEAT I
7 : EXPRESSION
8 DUP 18 = OVER 19 = OR
9 IF GETSYM THEN
10 BEGIN
11 TERME DUP 18 = OVER 19 = OR
12 WHILE
13 GETSYM
14 REPEAT I
15 ' EXPRESSION CFA SWAP !

```

```

SCR # 13
0 ( *** ANALYSEUR BLAISE 5 *** )
1 : CONDITION
2 DUP 7 =
3 IF GETSYM EXPRESSION
4 ELSE EXPRESSION DUP 21 > OVER 28 < AND
5 IF GETSYM EXPRESSION
6 ELSE 20 ERREUR
7 THEN THEN I
8 : INSTRUCTION [ SMUDGE ]
9 DUP CASE
10 32 OF GETSYM DUP 17 =
11 IF GETSYM EXPRESSION
12 ELSE 13 ERREUR THEN ENDOF
13 33 OF 12 ERREUR ENDOF
14 34 OF 12 ERREUR ENDOF
15 2 OF GETSYM DUP 34 =

```

```

SCR # 14
0 ( *** ANALYSEUR BLAISE 6 *** )
1 IF GETSYM
2 ELSE 14 ERREUR THEN ENDOF
3 6 OF GETSYM CONDITION DUP 9 =
4 IF GETSYM INSTRUCTION
5 ELSE 16 ERREUR THEN ENDOF
6 11 OF GETSYM CONDITION DUP 4 =
7 IF GETSYM INSTRUCTION
8 ELSE 18 ERREUR THEN ENDOF
9 1 OF GETSYM BEGIN INSTRUCTION DUP 28 =
10 WHILE GETSYM REPEAT DUP 5 -
11 IF 17 ERREUR THEN
12 GETSYM ENDOF
13 ENDCASE DUP 12 = OVER 13 = OR IF GETSYM DUP 14 -
14 IF 25 ERREUR THEN GETSYM BEGIN DUP 32 = OVER 33 = OR 0=
15 IF 12 ERREUR THEN GETSYM DUP 16 = WHILE GETSYM ---

```

T 366 ERIC AUBOURG

24 JUILLET 1985

```

SCR # 15
0 ( *** ANALYSEUR BLAISE 7 *** )
1 REPEAT
2 DUP 15 - IF 22 ERREUR THEN GETSYM
3 THEN ; SMUDGE
4 : BLOC [ SMUDGE ]
5 DUP 3 =
6 IF BEGIN
7 33 CONSTANT GETSYM DUP 22 -
8 IF 3 ERREUR THEN GETSYM DUP 31 -
9 IF 2 ERREUR THEN GETSYM DUP 16 -
10 UNTIL
11
12
13 DUP 28 - IF 5 ERREUR THEN GETSYM THEN
14 DUP 10 =
15 IF BEGIN

```

```

SCR # 16
0 ( *** ANALYSEUR BLAISE 8 *** )
1 32 CONSTANT GETSYM DUP 16 -
2
3 UNTIL
4
5 DUP 28 - IF 5 ERREUR THEN
6 GETSYM THEN
7 BEGIN
8 DUP 8 =
9 WHILE
10 34 CONSTANT GETSYM 28 - IF 5 ERREUR THEN
11 LATEST 0 GETSYM BLOC >R CURRENT 0 ! R>
12 DUP 28 - IF 5 ERREUR THEN GETSYM
13 REPEAT INSTRUCTION ; SMUDGE
14 : PROGRAMME
15 MOTRESERVES DEFINITIONS CR

```

```

SCR # 17
0 ( *** ANALYSEUR BLAISE 9 *** )
1 BLK 0 >R IN 0 >R B/SCR * BLK ! 0 IN !
2 0 GETSYM BLOC 29 - IF 9 ERREUR THEN
3 [COMPILE] FORTH DEFINITIONS
4 R> IN ! R> BLK ! ." ANALYSE TERMINEE " ;
5
6
7
8
9
10
11
12
13
14
15

```

T 366 ERIC AUBOURG

24 JUILLET 1985

LE COMPILATEUR

4^{ème} épisode : réalisation d'un compilateur Blaise

Ce compilateur sera réalisé en utilisant l'analyseur précédemment développé comme squelette, et en insérant les instructions de génération de code objet. Celui-ci sera du forth compilé.

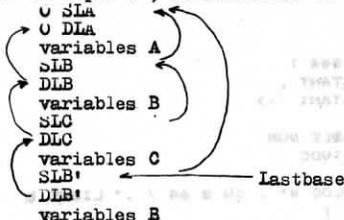
On utilise alors deux variables supplémentaires :
ADR, qui contient le RVA du dernier identificateur de procédure lu, ou le numéro de la dernière variable.
NUM, qui contient la valeur du dernier littéral ou de la dernière constante, ou bien le niveau de la dernière variable.

Le seul point appelant des commentaires est la gestion des niveaux de procédures, avec les variables locales et la possibilité de récursivité.

Les variables sont stockées sur la pile, pour permettre une gestion dynamique (les variables existent seulement pendant l'activation de la procédure correspondante : il faut donc bien les distinguer du mot déclaré par DEFVAR qui ne sert que durant la compilation), avec une structure de liste chaînée. On a en fait deux chaînes de pointeurs : les pointeurs dynamiques (DL = Dynamic Link), qui rendent compte de l'ordre chronologique des allocations, et les pointeurs statiques (SL) qui rendent compte des niveaux de procédures. Prenons un exemple : soit le programme

```
procédure A ;  
  procédure B ;  
    procédure C ;  
      Begin ( C )  
        ... B ( récursif ) ...  
      End ( C )  
    Begin ( B )  
      ... Call C ...  
    End ( B )  
  Begin ( A )  
    ... Call B ...  
  End ( A )
```

A est de niveau 1, B de niveau 2, et C de niveau 3. Après l'appel récursif de B par C, la structure de la pile est la suivante :



Une adresse de variables contient donc la différence de niveau (par exemple 1 pour une variable de B appelée par C) et le

numéro de cette variable dans son bloc. La procédure Base permet de remonter la chaîne des SL.

Il est possible maintenant d'exécuter un petit programme Blaise : les tours de Hanoi. Pour ceux qui ne connaissent pas ce grand classique, en voici le principe : il faut déplacer n anneaux de diamètres décroissant d'un piquet sur un autre, en utilisant un troisième piquet. On ne peut déplacer qu'un anneau à la fois, et ne le poser que sur un anneau de plus grand diamètre.

Après avoir rentré le programme, il suffit de faire le PROGRAMME HANOI pour le compiler, puis après le message " COMPILATION TERMINEE " faire HANOI n pour le lancer. Les anneaux sont initialement sur le piquet numéro 1, et à la fin sur le numéro 2. Les couples affichés donnent le piquet de départ et le piquet d'arrivée de l'anneau à déplacer.

Le blaise n'accepte pas d'arguments pour les procédures, mais l'utilisation de variables globales permet de pallier ce défaut, comme vous le voyez dans Hanoi.

WIN (pour l'instant ...)
T366 Eric AUBOURG

PROFF

SCR # 18

```
0 ( *** COMPILATEUR BLAISE 1 *** )
1
2 VOCABULARY MOTRESERVES MOTRESERVES DEFINITIONS
3 1 CONSTANT BEGIN      2 CONSTANT CALL
4 3 CONSTANT CONST      4 CONSTANT DO
5 5 CONSTANT END        6 CONSTANT IF
6 7 CONSTANT ODD        8 CONSTANT PROCEDURE
7 9 CONSTANT THEN      10 CONSTANT VAR
8 11 CONSTANT WHILE     12 CONSTANT READ
9 13 CONSTANT WRITE     15 CONSTANT )
10 17 CONSTANT !"       18 CONSTANT +
11 19 CONSTANT -        20 CONSTANT *
12 21 CONSTANT /        22 CONSTANT =
13 23 CONSTANT >        24 CONSTANT <
14 25 CONSTANT <>      26 CONSTANT <=
15 27 CONSTANT >=      28 CONSTANT ! -->
```

SCR # 19

```
0 ( *** COMPILATEUR BLAISE 2 *** )
1 14 CONSTANT (        16 CONSTANT ,
2 29 CONSTANT .        30 CONSTANT -->
3 FORTH DEFINITIONS
4 0 VARIABLE ADR      0 VARIABLE NUM
5 ' MOTRESERVES CONSTANT LIMITVOC
6 : ERREUR
7 ." ERREUR #" . BLK @ ." BLOC #" . IN @ 64 / ." LIGNE #"
8 . CR HERE COUNT TYPE QUIT ;
9 : GETSYM
10 DROP -FIND
11 IF DROP CFA DUP LIMITVOC > ELSE 0 DUP THEN
12 IF EXECUTE DUP 30 =
13 IF (COMPILE) --> [ SMUDGE ] GETSYM [ SMUDGE ] THEN
14
15 -->
```

```

SCR # 20
0 ( *** COMPILATEUR BLAISE 3 *** )
1 ELSE
2 DROP HERE NUMBER DROP NUM ! 31
3 THEN ;
4 : DEFCONST ( LEVEL SYM --- LEVEL PFA SYM )
5 <BUILDS HERE SWAP 0 ,
6 DOES> 0 NUM ! 33 ; ( VALEUR -> NUM, --- 33 )
7 : DEFVAR ( LEVEL # SYM )
8 <BUILDS >R OVER OVER , , R> ( EXEC : # -> ADR, )
9 DOES> DUP 0 ADR ! 2+ 0 NUM ! 32 ; ( LEVEL -> NUM, --- 32 )
10 : DEFPROC ( LEVEL SYM -- PFA LEVEL SYM )
11 <BUILDS HERE ROT ROT 0 , OVER , 0 ,
12 DOES> ADR ! 34 ; ( PFA -> ADR, --- 34 )
13 0 VARIABLE LASTBASE
14 CREATE SPSTO ( N --- , N->SP )
15 HEX F9E1 , E9FD , DECIMAL SMUDGE -->

```

```

SCR # 21
0 ( *** COMPILATEUR BLAISE 4 *** )
1 : BASE ( LEVEL APPELANT-APPELE --- ADRESSE )
2 LASTBASE 0 SWAP DUP 0 >
3 IF 0 DO 0 LOOP
4 ELSE DROP THEN ;
5 : APPEL ( ARGUMENT COMPILER PFA , LEVEL APPELANT )
6 R> DUP 4 + >R DUP 2+ 0 SWAP 0 ( LEVEL PFA )
7 DUP >R 2+ 0 - BASE LASTBASE 0 2 - ( POINTEURS SL DL )
8 SP0 DUP 2+ LASTBASE ! R 4 + 0 - SPSTO
9 R> 0 EXECUTE
10 LASTBASE 0 2 - SPSTO 2+ LASTBASE ! DROP ;
11 : (RCLVAR) ( ADR DLEV --- VALEUR )
12 BASE SWAP - 0 ;
13 : RCLVAR ( LEVEL SYM )
14 COMPILER LIT ADR 0 , COMPILER LIT OVER NUM 0 - ,
15 COMPILER (RCLVAR) ; -->

```

```

SCR # 22
0 ( *** COMPILATEUR BLAISE 5 *** )
1 : STOVAR ( VAL ADR LEV --- )
2 BASE SWAP - ! ;
3 : INPUT
4 BEGIN
5 BL WORD HERE 1+ C0 0=
6 WHILE
7 QUERY
8 REPEAT
9 HERE NUMBER DROP ;
10 0
11 : FACTEUR ( LEVEL SYM )
12 DUP 31 = OVER 33 = OR
13 IF COMPILER LIT NUM 0 , GETSYM ELSE
14 DUP 32 = IF RCLVAR.GETSYM ELSE
15 DUP 34 = IF 21 ERREUR ELSE -->

```

SCR # 23

```

0 ( *** COMPILATEUR BLAISE 6 *** )
1 DUP 14 = IF GETSYM [ HERE SP0 18 + ! 0 , ]
2 15 - IF 22 ERREUR THEN DUP GETSYM
3 ELSE 23 ERREUR
4 THEN THEN THEN ;
5 : TERME ( LEVEL SYM )
6 FACTEUR
7 BEGIN
8 DUP 20 = IF 1 >R R ELSE
9 DUP 21 = IF 0 >R 1 ELSE
10 0 THEN THEN
11 WHILE
12 GETSYM FACTEUR R>
13 IF COMPILE *
14 ELSE COMPILE / THEN
15 REPEAT ; --)

```

SCR # 24

```

0 ( *** COMPILATEUR BLAISE 7 *** )
1 : EXPRESSION ( LEVEL SYM )
2 DUP 18 = IF GETSYM 0 ELSE
3 DUP 19 = IF GETSYM 1 ELSE
4 0 THEN THEN >P
5 TERME
6 R> IF COMPILE MINUS THEN
7 BEGIN
8 DUP 18 = IF 1 >R I ELSE
9 DUP 19 = IF 0 >R 1 ELSE
10 0 THEN THEN
11 WHILE
12 GETSYM TERME R>
13 IF COMPILE +
14 ELSE COMPILE - THEN
15 REPEAT ; ' EXPRESSION CFA SWAP ! ; --)

```

SCR # 25

```

0 ( *** COMPILATEUR BLAISE 8 *** )
1 : CONDITION ( LEVEL SYM )
2 DUP 7 =
3 IF GETSYM EXPRESSION COMPILE 2 COMPILE MOD
4 ELSE EXPRESSION
5 DUP 22 < OVER 27 > OR
6 IF 20 ERREUR THEN >R
7 DUP GETSYM EXPRESSION R>
8 CASE
9 22 OF COMPILE = ENDOF
10 23 OF COMPILE < ENDOF
11 24 OF COMPILE < ENDOF
12 25 OF COMPILE = COMPILE 0 = ENDOF
13 26 OF COMPILE > COMPILE 0 = ENDOF
14 27 OF COMPILE < COMPILE 0 = ENDOF
15 ENDCASE THEN ; -->

```

```

SCR # 26
0 ( *** COMPILATEUR BLAISE 9 *** )
1 : INSTRUCTION ( LEVEL SYM )
2 [ SMUDGE ] DUP
3 CASE
4 32 OF ADR @ NUM @ >R >R GETSYM 17 -
5 IF 13 ERREUR THEN
6 DUP GETSYM EXPRESSION COMPILE LIT R> ,
7 COMPILE LIT OVER R> - , COMPILE STOVAR ENDOF
8 33 OF 12 ERREUR ENDOF
9 34 OF 12 ERREUR ENDOF
10 2 OF GETSYM DUP 34 =
11 IF COMPILE APPEL ADR @ , OVER , GETSYM
12 ELSE 14 ERREUR THEN ENDOF
13 6 OF GETSYM CONDITION [COMPILE] IF >R >R DUP 9 =
14 IF GETSYM INSTRUCTION R> R> [COMPILE] ENDIF
15 ELSE 16 ERREUR THEN ENDOF -->

```

```

SCR # 27
0 ( *** COMPILATEUR BLAISE 10 *** )
1 11 OF [COMPILE] BEGIN >R >R GETSYM CONDITION DUP 4 = IF
2 GETSYM [COMPILE] WHILE >R >R INSTRUCTION R> R> ROT
3 ROT R> ROT POT [COMPILE] REPEAT
4 ELSE 18 ERREUR THEN ENDOF
5 1 OF GETSYM BEGIN INSTRUCTION DUP 28 = WHILE GETSYM
6 REPEAT DUP 5 - IF 17 ERREUR THEN GETSYM ENDOF
7 ENDCASE
8 DUP 12 = IF 1 1 ELSE DUP 13 = IF 0 1 ELSE 0 THEN THEN
9 IF >R GETSYM DUP 14 - IF 25 ERREUR THEN
10 GETSYM
11 BEGIN DUP 32 - IF 12 ERREUR THEN
12 I IF COMPILE INPUT COMPILE LIT ADR @ ,
13 COMPILE LIT OVER NUM @ - , COMPILE STOVAR
14 ELSE RCLVAR COMPILE . THEN
15 GETSYM DUP 16 = -->

```

```

SCR # 28
0 ( *** COMPILATEUR BLAISE 11 *** )
1 WHILE GETSYM
2 REPEAT R> DROP
3 DUP 15 - IF 22 ERREUR THEN
4 GETSYM COMPILE CR THEN ; SMUDGE
5 : BLOC ( LEVEL SYM )
6 SWAP 1+ SWAP [ SMUDGE ]
7 DUP 3 = ( CONSTANCE )
8 IF BEGIN
9 DEFCONST ( LEVEL PFA SYM )
10 GETSYM DUP 22 - IF 3 ERREUR THEN
11 GETSYM 31 - IF 2 ERREUR THEN
12 NUM @ SWAP ! 0 GETSYM DUP 16 -
13 UNTIL
14 DUP 28 - IF 5 ERREUR THEN GETSYM
15 THEN 2 >R -->

```

```

SCR # 29
0 ( *** COMPILATEUR BLAISE 12 *** )
1 DUP 10 = ( VARIABLE )
2 IF R> SWAP ( LEVEL # SYM )
3 BEGIN
4 SWAP 2+ SWAP DEFVAR GETSYM DUP 16 -
5 UNTIL SWAP >R ( # VAR SUR RS )
6 DUP 28 - IF 5 ERREUR THEN GETSYM
7 THEN
8 BEGIN
9 DUP 8 = ( PROCEDURE )
10 WHILE
11 DEFPROC ( PFA LEVEL SYM )
12 GETSYM 28 - IF 5 ERREUR THEN
13 LATEST >R 0 GETSYM BLOC R> CURRENT 0 !
14 DUP 28 - IF 5 ERREUR THEN GETSYM
15 REPEAT -->

```

```

SCR # 30
0 ( *** COMPILATEUR BLAISE 13 *** )
1 ROT HERE OVER ! 4 + R> 2 - SWAP !
2 [ ' : 18 + ] LITERAL , [COMPILE] ]
3 INSTRUCTION COMPILE !S
4 SWAP 1 - SWAP ! SMUDGE
5 : INIT 0 SP0 LASTBASE ! 0 R> DUP 2+ >R 0 >R SP0 R 4 + 0 - SPSTO
6 R> 0 EXECUTE CR ." EXECUTION TERMINEE" SP !
7 : PROGRAMME ( #BLOC --- PROGRAMME NOM )
8 HERE 0 , 0 , 0 , [COMPILE] : COMPILE INIT DUP ,
9 [COMPILE] ! SWAP
10 BLK 0 >R IN 0 >R B/SCR * BLK ! 0 IN !
11 MOTRESERVES DEFINITIONS CR [COMPILE] ]
12 0 DUP GETSYM BLOC 29 - IF 9 ERREUR THEN
13 DROP [COMPILE] FORTH DEFINITIONS [COMPILE] [
14 R> IN ! R> BLK ! ." COMPILATION TERMINEE "
15 !

```

PROGRAMME BLAISE : TOURS DE HANDI

```

SCR # 1
0 VAR N , I , J ;
1 PROCEDURE DEPLACER ;
2 VAR N1 , I1 , J1 ;
3 BEGIN
4 IF N = 1 THEN WRITE ( I , J ) ;
5 IF N > 1 THEN
6 BEGIN
7 I1 := I ; J1 := J ; N1 := N ;
8 N := N1 - 1 ; J := 6 - I - J ;
9 CALL DEPLACER ;
10 N1 := 1 ; J := J1 ; I := I1 ;
11 CALL DEPLACER ;
12 N := N1 - 1 ; I := 6 - I1 - J1 ; J := J1 ;
13 CALL DEPLACER
14 END
15 END ! -->

```

```

SCR # 2
0 BEGIN
1 READ ( N ) ; I := 1 ; J := 2 ;
2 CALL DEPLACER
3 END .
4
5

```


HP-41-CX FUNCTIONS


XROM 25,48 -CX EXT FCN


XROM 25,49 ASROOM


XROM 25,50 CLRGX


XROM 25,51 ED


XROM 25,52 EMDIRX


XROM 25,53 EMROOM


XROM 25,54 GETKEYX


XROM 25,55 RESZFL


XROM 25,56 ΣREG?


XROM 25,57 X=NN?


XROM 25,58 X≠NN?


XROM 25,59 X<NN?


XROM 25,60 X≤NN?


XROM 25,61 X>NN?


XROM 25,62 X≥NN?


XROM 26,30 -CX TIME


XROM 26,31 CLALMA


XROM 26,32 CLALMX


XROM 26,33 CLRALMB


XROM 26,34 RCLALM


XROM 26,35 SWPT

Eric Ferguson

MODULE HP-II/DEV

RS

A-BUF


DDL


MONITOR


RG-BUF?


WFRM


HEXVIEW


A=BUF?


DDT


NRD


RREG


WREG


NOT


A=BUF?X?


FRAV?


NRE


SAI


X-BUF


OR


AAD


FRNS?


ORAV?


SCOPE


X=BUF?


OCTIN


AAU


GET


OUTBIN


SDA


X<>FLAG


OCTVIEW


AIPT


GTL


OUTBINY


SDC


AND


ROMCHKX


BSIZE?


IDY


OUTBUF?X


SDI


ASIZE?


ROTXY


BSIZEX


IFCR?


PT=


SF33


A-XL


XOR


BUF-AX	IFC	PT?	SRQR?	A-XR	X-AL
BUF-RGX	INBIN	PRBYTES	SST	A-XX	X-AR
BUF-XA	INBUF	PRFRMS	TAD	BININ	Y-AX
BUF-XB	LAD	REN	TCT	BINVIEW	
CF33	LPD	RFRM	UNL	BIT?	
CMD	MIPT	RG-BUFX	UNT	HEXIN	
AND	BCD>BIN	CVIEW	LOADP	X>ROM	1CMP
OR	BIN>BCD	VIEWA	MLDL?	ROM?	2CMP
APPFN	CF55	DISS	NEXTFN	RXL	1-D
ASSEM	SF55	GETPC	NRCL	RXR	2-D
DISASM	CLROM	PUTPC	NSTO	SXL	3-D
A>X	CODE	HEXKB	PCWRT	SXR	4-D
X>A	DECODE	INSBYTE	REG>ROM	X>\$	
	COMPILE	RCLBYTE	ROM>REG	X+Y	
	COPYROM	STOBYTE	ROM>X	Y-X	

RS (178)

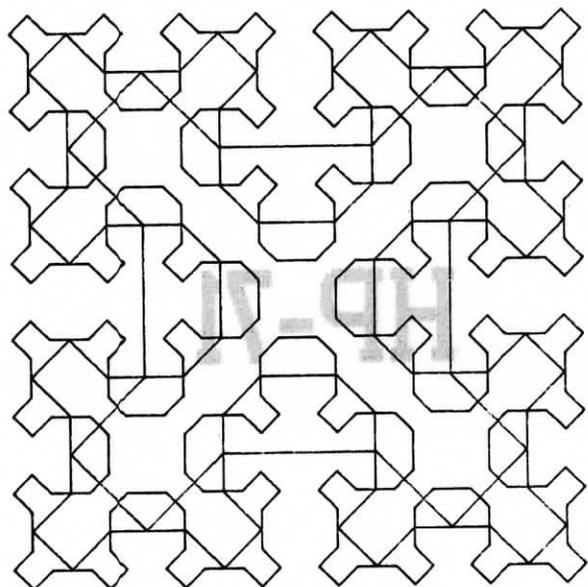
MODULE IL

-IMP IL	PRP	-LECT K7	SEC	-CTL FNS	MANIO
ACA	'PRPLOT	CREATE	SEEKR	AUTOIO	OUTA
ACCHR	'PRPLOT P	DIR	UNSEC	FINDIO	PWRDN
ACCOL	PRREG	NEWM	VERIFY	INA	PWRUP
ACSPEC	PRREGX	PURGE	WRTA	IND	REMOTE
ACX	PR Σ	READA	WRTK	INSTAT	SELECT
BLDSPEC	PRSTK	READK	WRTP	LISTEN	STOPIO
LIST	PRX	READP	WRTPV	LOCAL	TRIGGER
PRA	REGPLOT	READR	WRTR		
'PRAXIS	SKPCHR	READRX	WRTRX		
PRBUF	SKPCOL	READS	WRTS		
PRFLAGS	STKPLOT	READSUB	ZERO		
PRKEYS	FMT	RENAME	--		

R.S.
(TAM)



HP-71



KAS ET PLISTALL

KAS ET PLISTALL

KAS est une version du classique jeu qui consiste à casser des murs avec une balle et une raquette. Il est en BASIC, donc pas d'une rapidité extraordinaire (d'autant plus que le système d'adressage point à point de l'écran n'est pas des plus simples!), mais permet déjà de s'amuser. Le mode d'emploi est rudimentaire : run kas, et c'est parti. . . Ce sont les touches "+" et "/" qui commandent la raquette.

PLISTALL servira à tous ceux qui, comme moi, n'ont accès à une imprimante qu'épisodiquement. Cet utilitaire leur permettra de lister tous les fichiers BASIC de leur machine en une seule fois. Ce programme est un peu acrobatique, à cause du traitement spécial qu'il faut accorder au WORKFILE (sortie du mode programme, déclaration du WORKFILE comme fichier courant, listage de celui-ci, et retour au programme, le tout commandé par le programme lui-même) mais son utilisation est de tout repos, sauf pour l'imprimante, bien sûr. . .

Le listing des deux programmes fourni ici est un exemple d'utilisation de PLISTALL.

b. piguet (7 152)

```
Liste effectuée le 01/01/1900 à 18:10:30
workfile BASIC 0 01/01/00 00:00
PLISTALL BASIC 768 05/12/84 17:30 0
5 DESTROY ALL @ DISP 'Listing en cours'
10 PRINT 'Liste effectuée le' & DATE$(7,8) & DATE$(13,6) & ' 19' & DATE$(1,2) & ' à' & TIME$
20 A=1 @ B=-.01
30 A#=CAT$(A)[1,16]
40 IF A#='' THEN 100
50 IF A#[12,16]# 'BASIC' THEN A=A+1 @ GOTO 30
60 PRINT CAT$(A)
70 IF A#[10,10]='P' OR A#[10,10]='E' THEN PRINT 'Fichier privé' @ A=A+1 @ GOTO 30
80 IF A#[1,8]='workfile' THEN GOTO 'SPEC' ELSE PLIST A#[1,8]
90 A=A+1 @ GOTO 30
100 B=B+.01 @ IF FP(B)=.04 THEN B=IP(B)+1
110 IF B>4.03 THEN END
120 A=1 @ ON ERROR GOSUB 'ERR'
130 B#=':PORT(' & STR$(B) & ') ' @ A#=CAT$(A,B#[1,16]
140 IF A#='' THEN 100
150 IF A#[12,16]# 'BASIC' THEN A=A+1 @ GOTO 130
160 PRINT CAT$(A,B#)
170 IF A#[10,10]='P' OR A#[10,10]='E' THEN PRINT 'Fichier privé' ELSE C#=#[1,8] & B# @ PLIST C#
180 A=A+1 @ GOTO 130
200 'ERR': IF ERR#64 THEN DISP ERR# @ RETURN
210 PRINT 'Le ' & B# & ' n'existe pas'
220 POP @ GOTO 100
300 'SPEC': D=FLAG(-9,1)
310 DEF KEY '#155', 'EDIT@PLIST@EDITPLISTALL@RUN400':
320 IF KEY# THEN 320 ELSE PUT '#155' @ PAUSE
400 D=FLAG(-9,D) @ DEF KEY '#155' @ DISP 'Listing en cours...' @ GOTO 90
```

KAS ET PLISTAL

```
KAS BASIC 925 05/12/84 18:53 0
10 DESTROY ALL @ CONTRAST 10 @ DISP TAB(9);'KAS'
20 DIM A$(41) @ POKE '2F441','F'
30 V=.4 @ T=RND*6+1.6 @ K=7
40 ON TIMER #1,380 GOTO 50
50 X=5 @ H=1
60 CFLAG N @ N=N+1 @ SFLAG N
70 IF N)1 THEN DISP 'BRAVO! Tableau suivant'
80 IF N=4 THEN OFF TIMER #1
100 DISP @ A$=GDISP$(1,40)
110 FOR I=17 TO 31 STEP 7
120 FOR J=1 TO 4
130 A$(I+J,I+J)=CHR$(255)
140 NEXT J
150 NEXT I
160 A$(41)=CHR$(255)
200 IF KEYDOWN('14') THEN K=MAX(K/2,7)
210 IF KEYDOWN('56') THEN K=MIN(K*2,224)
220 A$(5,5)=CHR$(K)
300 Y=X+H @ U=T+V @ W=INT(U)-1
310 IF W=7 OR NOT W THEN V=-V
320 P=FLAG(63,BIT(NUM(A$(Y)),W))
330 IF X<4 THEN 'FIN'
340 A$(Y,Y)=CHR$(BINIOR(NUM(A$(Y)),2^W))
350 A$(X,X)=CHR$(NUM(A$(X))-2^(INT(T)-1)*(X#5 OR H<0))
400 GDISP A$
410 IF Y)39 OR FLAG(63) THEN H=-H @ BEEP 1320,.007 @ S=S+1
420 X=Y @ T=U
430 GOTO 200
900 'FIN': OFF TIMER #1 @ BEEP 220,1 @ CFLAG N @ CONTRAST 7 @ ASSIGN #1 TO
HIGH @ READ #1:A
910 IF S)A THEN DISP 'RECORD !';S @ RESTORE #1 @ PRINT #1;S ELSE DISP 'SCOR
E';S
920 POKE '2F441','0' @ END
1320 P=FLAG(63,MOD(INT(NUM(A$(Y))/2^W),2))
1340 A$(Y,Y)=CHR$(NUM(A$(Y))+2^W*NOT FLAG(63))
2000 ! V:Increment Vertical
2010 ! T:Position Verticale actuelle
2020 ! U:Nouvelle Position Verticale
2030 ! W:INT(U)-1
2040 !
2050 ! H:Increment Horizontal
2060 ! X:Position Horizontale actuelle
2070 ! X:Nouvelle position Horizontale
2080 !
2090 ! El Font a voir Font : CREATE SOATA HIGH.1
```

STATISTIQUES

INTRODUCTION :

■ Pourquoi un programme de statistique ,

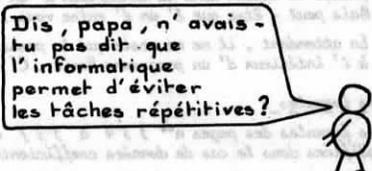
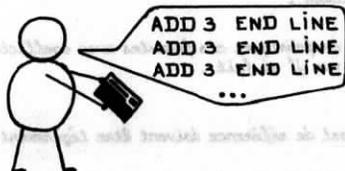
alors que le micropochette HP 71 B possède de nombreux et puissants mots-clés statistiques ?

1°) Parce qu'un programme, qui utilise ces mots-clés, qui facilite l'entrée des données, la correction des erreurs, et la sortie des résultats, qui est adapté aux besoins usuels de la personne qui possède ce micropochette, qui soulage la mémoire de cet utilisateur (l' n' a plus besoin de se rappeler la syntaxe de chacun des mots-clés statistiques), est une aide appréciable .

2°) Parce que (Quelle surprenante absence parmi tant d'abondance !), dans la version de base du HP 71 B, les instructions statistiques ne sont prévues que pour des calculs statistiques sans coefficients et des calculs de probabilité où toutes les valeurs sont équiprobables . Ce qui est plutôt restrictif ;
(imaginez un instant que vous soyez obligé(e) d'introduire le tableau statistique suivant :

x_i	3	4	5	6
n_i	127	432	258	183

Trouvez - vous normal de devoir faire cent vingt - sept fois ADD 3 , et cetera ?



Un programme, permettant de saisir chaque donnée (une simple valeur x , ou un couple (x,y) , ou ...) d'un coefficient ou d'une probabilité, est donc une nécessité .
D' où le programme ci - joint .

■ Ce programme doit être compatible à 100 % avec ce qui existe :

Quand on arrête ce programme, on peut manuellement compléter ou modifier le tableau des sommes statistiques, ou sortir des résultats à l'aide des instructions et des fonctions contenues dans le Basic du HP 71 B .

PREDV, par exemple, permet de trouver une valeur estimée d'une variable ...

Il y a toutefois trois restrictions dans le programme ci - joint :

1°) Le tableau des sommes statistiques s'appelle arbitrairement SØ .

Et il ne semble pas être simple de laisser, à chaque exécution du programme, le choix à l'utilisateur du nom de ce tableau . Mais, est - ce bien gênant ?

2°) Le programme proposé est limité à des calculs de statistique (ou de probabilité) à une ou deux variables notées x et y .

Il faudrait modifier ce programme si des calculs à plus de deux variables étaient nécessaires .

3°) La seule régression, effectuée lors de la sortie des résultats, approxime le nuage de points pondérés par une droite affine d'équation $y = ax + b$, où les notations a et b utilisées pour les coefficients sont celles en usage en France .

Pour obtenir d'autres régressions (puissances, logarithmes, exponentielles, ...), il faudrait compléter ce programme, en s'aidant des pages n° 86 à 89 du manuel d'utilisation .

■ Avec un cadeau :

Un exemple de gestion d' " erreurs ", voici ce qui vous est offert en prime dans ce programme .

STATISTIQUES AVEC COEFFICIENTS SUR HP 71 B :

1^{ère} solution :

1^{ère} solution :

Inclure dans un programme une boucle du genre :

FOR M=1 TO N @ ADD X, Y @ NEXT M .

Mais ceci aurait deux inconvénients :

1°) Cette solution nécessiterait que le coefficient N soit entier .

Ce qui, pour un probabiliste, limiterait l'utilisation au cas où les probabilités sont des nombres rationnels, et l'obligerait à multiplier ces probabilités par un dénominateur commun .

2°) Si, comme cela est fréquemment le cas en statistique, le coefficient N a une grande valeur, le temps d'exécution de cette boucle deviendrait trop grand, avec en plus la possibilité d'un cumul non négligeable des approximations éventuellement effectuées lors de ces nombreux ADD .

2^{ème} solution :

Remplacer les formules sans coefficients par des formules avec coefficients .

L'idéal serait, bien sûr, un fichier LEX qui autoriserait les ordres ADD et DROP avec une syntaxe comprenant un coefficient .

J'ignore malheureusement si un tel fichier LEX existe .

Et mes connaissances relatives au HP 71 B ne sont pas encore suffisantes pour que je puisse me lancer dans la conception d'un tel fichier .

Mais peut-être que l'un d'entre vous ...

En attendant, il ne me reste que la possibilité de fabriquer ces formules avec coefficients à l'intérieur d'un programme Basic . C'est ce que j'ai fait .

Les formules :

Les formules des pages n°s 334 à 337 du manuel de référence doivent être légèrement modifiées dans le cas de données coefficientées .

Une fois les données introduites en totalité, les éléments du tableau des sommes statistiques doivent avoir les valeurs suivantes (modification des formules de la page n° 335) :

$$T(\Phi) = \sum_i n_i$$

$$T(j) = T(\Phi) \cdot \text{MEAN}(j) = \sum_i n_i \cdot x_{ij}$$

$$S(j, k) = T(\Phi) \cdot \text{covariance}(j, k) = \sum_i n_i \cdot x_{ij} \cdot x_{ik} - \frac{T(j)T(k)}{T(\Phi)}$$

$$= \sum_i n_i \cdot (x_{ij} - \text{MEAN}(j)) \cdot (x_{ik} - \text{MEAN}(k)) .$$

Pour obtenir ces sommes, on démontrerait qu'il faut, à chaque introduction de donnée(s), appliquer les formules suivantes (modification des formules de la page n° 336) :

$$\text{new } S(j, k) = \text{old } S(j, k) + \frac{n_i (\text{old } T(\Phi) \cdot x_{ij} - \text{old } T(j)) (\text{old } T(\Phi) \cdot x_{ik} - \text{old } T(k))}{\text{old } T(\Phi) \cdot \text{new } T(\Phi)}$$

$$\text{new } T(j) = \text{old } T(j) + n_i \cdot x_{ij}$$

$$\text{new } T(\Phi) = \text{old } T(\Phi) + n_i .$$

Ce sont ces formules qui sont utilisées dans les lignes n° 340 à 440, à quelques différences de notations près .

Vu l'ordre dans lequel les sommes statistiques sont rangées dans le tableau (noté SΦ dans le programme), les formules de correspondance sont :

$$\text{si } \Phi \leq j, \text{ alors } T(j) = S\Phi \left(\frac{j(\Phi+1)}{2} \right); \text{ et, si } \Phi < j \leq k, \text{ alors } S(j, k) = S\Phi \left(j + \frac{k(\Phi+1)}{2} \right) .$$

De plus, dans ce programme, n_i est noté N, $x_{i,1}$ est noté X, et $x_{i,2}$ est noté Y .

Comment retirer, des sommes statistiques, une donnée erronée :

En cas de statistiques avec coefficients, ceci est simple.
Pour retirer la donnée coefficientée (x, n) ou (x, y, n) précédemment introduite, il suffit d'introduire la donnée coefficientée $(x, -n)$ ou $(x, y, -n)$.

Mais, comme ceci est impossible dans le cas de statistiques sans coefficients, j'ai inclus dans ce programme la possibilité de supprimer une donnée déjà introduite, à l'aide du mot de passe "exclure" (voir le mode d'emploi sur la page suivante).

Remarque : Si l'une des deux valeurs $x_{\min} = 58$ et $x_{\max} = 59$ avait été modifiée par une entrée, et si cette entrée est ensuite supprimée, alors les anciennes valeurs de x_{\min} et x_{\max} ne sont pas rétablies.

Une impossibilité éventuelle, qui est la conséquence d'un choix fait par H.P. :

Parce qu'il supprime une donnée erronée, ou parce qu'il fait un calcul barycentrique avec des coefficients munis de signes (par exemple : recherche des coordonnées du barycentre d'un ensemble de charges électriques ponctuelles), l'utilisateur emploiera des coefficients négatifs. La somme des coefficients peut donc, au cours de l'introduction des données (ne serait-ce qu'au début !), s'annuler.

Etant donnée la formule donnant $new S(j, k)$ (voir sur la page précédente), il y a une "division par zéro", si l'ancienne ou la nouvelle somme des coefficients est nulle.

On démontrerait que, à une condition signalée plus bas, ce problème est correctement résolu par le saut conditionnel de la ligne n° 360 à la ligne n° 420.

Par contre, dans ce cas, la remise à zéro systématique des $S(j, k)$ par Hewlett - Packard (indiquée page n° 336 du manuel de référence) n'est nullement (c'est le cas de le dire) mathématiquement justifiée.

Il est normal qu'il puisse rester dans ces circonstances un reliquat de (co)variance ; et ceci n'est pas gênant.

La présence de ce bogue dans un logiciel Hewlett - Packard est surprenante ...

Je n'ai donc pas inclus cette remise à zéro arbitraire dans mon programme.

Toutefois, il reste un problème : celui du cas où la somme $T(\emptyset)$ des coefficients s'annule sans que tous les totaux $T(j)$ soient nuls.

Dans ce cas, l'un des $S(j, k)$ au moins devrait devenir provisoirement infini.

Et il n'y a pas de solution simple pour qu'ensuite ce $S(j, k)$ reprenne une valeur finie correcte.

Sauf si on abandonne, au moins provisoirement, la définition donnée de $S(j, k)$, pour la remplacer par $\sum_i n_i x_i, j^x_i, k$.

Mais ceci serait contraire au choix fait par Hewlett - Packard (voir la page n° 335 du manuel de référence).

D'où la décision que j'ai prise de faire afficher un message d'erreur, et d'arrêter le programme (lignes n° 470 à 520), dans ce cas sous toute très exceptionnelle :

$$\left(\sum_i n_i = \emptyset \text{ mais } \left(\sum_i n_i x_i \neq \emptyset \text{ ou } \sum_i n_i y_i \neq \emptyset \right) \right)$$

↓
"Impossible sur HP 71 B"

MODE D'EMPLOI :

Comment répondre aux I.N.P.U.T. :

Dans ce programme, chacun des INPUT a une réponse par défaut, pour laquelle il suffit de valider (à l'aide des touches RUN ou END LINE).

Dans le cas d'une question où seules deux réponses contraires sont attendues, toute réponse différente de la réponse par défaut est interprétée comme étant la réponse contraire.

Mise en route :

- Après avoir fait `RUN STAT`, indiquez si vous désirez entrer des données ou sortir des résultats, lorsque cette question vous est posée : entrée ou sortie ?
Pour l'entrée, tapez `END LINE`. Pour la sortie, tapez sortie `END LINE`.

Entrée :

- Indiquez si vous désirez effacer ou non l'éventuel contenu antérieur du tableau des sommes statistiques, et réinitialiser ou non les variables statistiques.
En cas de début de statistique, tapez `END LINE`.
En cas de poursuite de l'entrée de données, tapez non `END LINE`.
- S'il y a effacement et réinitialisation, indiquez si les statistiques doivent être à une (tapez `END LINE`) ou à deux (tapez deux `END LINE`) variable(s).
- Puis indiquez si les statistiques doivent être sans (tapez `END LINE`) ou avec (tapez avec `END LINE`) coefficients.
- Ensuite, après un `BEEP` grave, sont demandées successivement la valeur de la variable x et, éventuellement, celles de la variable y et du coefficient n .
Les valeurs initiales de x , y et n sont respectivement Φ , Φ et 1 .
Leurs valeurs par défaut lors des entrées sont leurs précédentes valeurs.
- Ces entrées de données recommencent indéfiniment, jusqu'à un éventuel `BREAK` (double appui sur `ON` lors d'une entrée), ou jusqu'à l'introduction du mot de passe vide (appui sur `ON` puis sur `END LINE`) ou du mot de passe "exclure" lors d'une entrée de la variable x juste après le `BEEP` grave.

Suppression de données précédemment introduites :

- L'introduction du mot de passe "exclure", signalée ci-dessus, permet de retirer du tableau des sommes statistiques une donnée x ou (x, n) ou (x, y) ou (x, y, n) précédemment introduite.
Pour ce faire, il suffit de taper `exclure END LINE`, juste après le `BEEP` grave ; puis d'introduire la valeur de x , et éventuellement celles de y et de n , à exclure.
Par exemple, pour retirer la dernière donnée introduite, il suffit de taper `exclure END LINE`, juste après le `BEEP` grave ; puis de valider l'ancienne valeur de x , et éventuellement celles de y et de n .
- Mais, dans le cas de statistiques avec coefficients, il est encore plus facile de supprimer une donnée coefficientée erronée (voir sur la page précédente).
- Remarque : Si une donnée n a été que partiellement introduite (par exemple : introduction de la valeur de x , puis `BREAK` par un double appui sur `ON` lors de l'entrée de y ou de n), elle n a pas encore été additionnée dans le tableau des sommes statistiques.
Il n y a donc pas lieu de retirer cette donnée partielle du tableau.

Sortie :

- Quand on vient d'entrer des données, il y a deux façons d'accéder à la sortie des résultats : soit en appuyant sur `ON` puis sur `END LINE` juste après le `BEEP` grave (l'introduction de ce mot de passe vide étant, dans ce programme, interprété comme le signal de la fin des données), soit en arrêtant ce programme (double appui sur `ON` lors d'une entrée), en le faisant redémarrer au début (appui sur `RUN`), et en répondant sortie `END LINE` à la première question.
- La sortie des résultats commence par la somme des coefficients.
Pour monter ou descendre dans le tableau des résultats, il suffit d'utiliser les flèches curseur \uparrow et \downarrow .
- L'arrêt de la sortie des résultats s'obtient par un `BREAK` (simple appui sur `ON`).
Si on désire rentrer d'autres données, ou commencer un autre calcul statistique, il suffit de faire redémarrer ce programme (appui sur `RUN`), et de répondre aux questions conformément à ce qui est désiré.

UN EXEMPLE DE TRAITEMENT D' "ERREUR" SUR HP 71 B :

En probabilité, comme en statistique, l'utilisateur doit pouvoir entrer des données les unes à la suite des autres, sans que le programme lui demande à chaque fois : "Y a-t-il d'autres données à entrer ?".

Mais, bien que la question ne lui soit pas posée, l'utilisateur doit pouvoir indiquer au programme les opérations désirées (suppression d'une entrée erronée, sortie des résultats, ...) à l'aide, par exemple, de mots de passe alphanumériques.

Or l'introduction du mot de passe "exclure" (par exemple), alors que le programme attend la valeur de la variable x, entraîne, sur HP 71 B, une erreur (car "exclure" n'est pas une expression numérique valide).

Donc, pour tester l'éventuelle introduction d'un mot de passe, il suffit de prévoir une routine de traitement d'"erreur".

Et voici l'occasion d'utiliser les possibilités offertes par ce micropoche :

- 1°) A la ligne n° 250, on active le branchement à la routine, en cas d'erreur lors de l'introduction de la valeur de la variable x. Ce branchement est désactivé à la ligne n° 270.
- 2°) Dans la routine d'erreur, à la ligne n° 550, on obtient, avec une facilité déconcertante, la suite X\$ des caractères de la réponse erronée.
- 3°) Aux lignes n° 560 et 570, si cette réponse est l'un des mots de passe, on exécute les instructions correspondantes.

Remarque : S'il le désire, le programmeur peut modifier ou augmenter la liste des mots de passe et des instructions correspondantes (lignes n° 190, 210, 220, 310, 450, 560 et 570).

Premier exemple : Certains préféreront peut-être les mots anglais "add" et "drop" aux mots français "inclure" et "exclure" que j'ai choisis.

Second exemple : A la ligne n° 560, d'autres préféreront peut-être le mot "fin" à la place du mot de passe vide.

- 4°) Et enfin, avec les lignes n° 540, 550, 580, 590 et 260, dans le cas où l'entrée était erronée mais n'est aucun des mots de passe attendus (par exemple, l'utilisateur a tapé par étourderie 1**2), la question demandant la valeur de x est automatiquement reposée, avec comme réponse par défaut la réponse erronée X\$, précédemment introduite, réponse qui est automatiquement validée grâce au FKEY "#38".

Par parenthèse : Ceux qui ne possèdent pas le fichier LEX qui procure l'ordre FKEY peuvent, sans inconvénient ici, mettre à la place : 580 PUT "#38".

Ce qui précède entraîne une erreur (la même que précédemment).

Grâce au OFF ERROR (celui de la ligne n° 540), il n'y a plus de branchement à la routine d'erreur, mais il y a émission d'un BEEP aigu d'erreur et affichage du message correspondant à l'erreur faite.

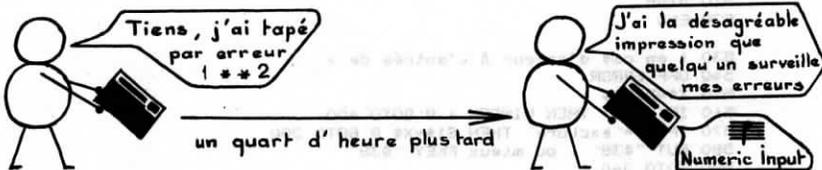
L'utilisateur peut donc lire ce message, appuyer sur g E R R M, puis corriger l'erreur faite.

Remarque : A cet instant, il est trop tard pour introduire un mot de passe, puisque le ON ERROR est désactivé.

Conclusion :

L'utilisateur a donc eu l'impression que son entrée erronée 1**2 a été simplement et normalement refusée par l'INPUT de la ligne n° 260, alors que discrètement cette entrée erronée a été comparée aux mots de passe.

Une telle routine d'"erreur" peut facilement être adaptée à de nombreux autres programmes.



Programme Basic "STAT"

Par Dominique TINEL , sur micropoche Hewlett Packard 71 B

Version du lundi 30 décembre 1985

Longueur = (24.5 + 1719.5) octets

```

10 ! Copyright : D.Tinel et PPC-Toulouse
20 DELAY .2 @ WINDOW 1
30 DISP 'statistique'
40 DELAY INF,INF
50 IF TYPE(S0)#5 THEN GOTO 120
60 LC ON
70 INPUT 'entrée/sortie : ','entrée';S$
80 IF S##'entrée' THEN GOTO 600
    
```

```

90 ! éventuelles initialisations
100 INPUT 'effacer : ','oui';S$
110 IF S##'oui' THEN GOTO 170
120 DESTROY S0 @ STAT S0(2)
130 X=0 @ Y=0 @ N=1
140 S8=INF @ S9=INF
150 INPUT 'variable : ','une';S2$ @ S2=S2##'une'
160 S3$="sans"
170 INPUT 'coefficient : ','S3$;S3$ @ S3=S3##'sans'
    
```

```

180 ! entrée
190 S1$="exclure"
200 DELAY 0,0 @ WINDOW 1
210 DISP S1$
220 DELAY INF,INF @ WINDOW 9
230 BEEP 350,.1
240 X$='x'
250 ON ERROR GOTO 530
260 INPUT 'x = ','X$;X$
270 OFF ERROR
280 IF S2 THEN INPUT 'y = ','y';Y
290 IF S3 THEN INPUT 'n = ','n';N ELSE N=1
    
```

```

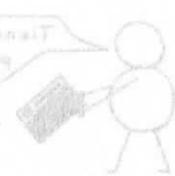
300 ! modification du tableau des sommes statistiques
310 IF S1$="exclure" THEN N=-N
320 S8=MIN(S8,X)
330 S9=MAX(S9,X)
340 O=S0(0)
350 P=O*(O+N)
360 IF P=0 THEN GOTO 420
370 U=O*X-S0(1)
380 V=O*Y-S0(3)
390 S0(2)=S0(2)+N*U*U/P
400 S0(4)=S0(4)+N*U*V/P
410 S0(5)=S0(5)+N*V*V/P
420 S0(1)=S0(1)+N*X
430 S0(3)=S0(3)+N*Y
440 S0(0)=S0(0)+N
450 IF S1$="exclure" THEN N=-N
460 IF S0(0)#0 OR S0(1)=0 AND S0(3)=0 THEN GOTO 180
    
```

```

470 ! en cas d'impossibilité due au choix fait par H.P.
480 WINDOW 1
490 DISP "Impossible sur HP 71 B"
500 BEEP 1400,.1
510 DISP
520 END
    
```

```

530 ! en cas d'erreur à l'entrée de x
540 OFF ERROR
550 X$=DISP$
560 IF X$="" THEN WINDOW 1 @ GOTO 600
570 IF X$="exclure" THEN S1$=X$ @ GOTO 200
580 PUT "#38" ! ou mieux FKEY "#38"
590 GOTO 260
    
```



```

600 I sorties
610 J=3
620 ON J GOSUB 670,690,710,730,750,770,790,810,830,850,880,900
630 K$=KEY$
640 IF K$=#51 THEN J=J+(J(7+5*S2) @ GOTO 620
650 IF K$=#50 THEN J=J-(J(2-S2) @ GOTO 620
660 GOTO 630

670 DISP 'Σ niyi = 'STR$(S0(3))
680 RETURN
690 DISP 'Σ nixi = 'STR$(S0(1))
700 RETURN
710 DISP 'Σ ni = 'STR$(S0(0))
720 RETURN
730 DISP 'x min = 'STR$(S8)
740 RETURN
750 DISP 'x̄ = 'STR$(MEAN(1))
760 RETURN
770 DISP 'x max = 'STR$(S9)
780 RETURN
790 DISP 'σx = 'SQR(S0(2)/S0(0))
800 RETURN
810 DISP 'ȳ = 'STR$(MEAN(2))
820 RETURN
830 DISP 'σy = 'SQR(S0(5)/S0(0))
840 RETURN
850 LR 2,1,B,A
860 DISP 'a = 'STR$(A)
870 RETURN
880 DISP 'b = 'STR$(B)
890 RETURN
900 FIX 1
910 DISP 'corr. = '100*ABS(CORR(1,2))%'
920 STD
930 RETURN

```

SIGNIFICATION DES VARIABLES :

X = 1^{ère} variable x_i .

Y = 2^{ème} variable y_i (nulle s'il n'y a qu'une variable).

N = coefficient n_i (égal à 1 si c' est sans coefficient).

SØ = tableau où sont rangées les sommes statistiques.

S1\$ = "inclure" ou "exclure" suivant que la prochaine donnée entrée devra être incluse dans le tableau, ou exclue de ce tableau.

S2 = Ø (faux) ou 1 (vrai) suivant que les statistiques sont à une ou deux variables).

S3 = Ø (faux) ou 1 (vrai) suivant que les statistiques sont sans ou avec coefficients.

S8 = la valeur minimale actuelle des x_i .

S9 = la valeur maximale actuelle des x_i .

S\$ et S2\$ et S3\$ = réponses alphanumériques à certaines questions.

X\$ = réponse par défaut (ligne n° 240) ou erronée (ligne n° 550) à l'entrée de X.

O = ancienne somme des coefficients.

O + N = nouvelle somme des coefficients.

P = produit de ces deux sommes.

U et V = résultats de calculs intermédiaires.

J = indice du résultat à sortir.

7 + 5 * S2 et 2 - S2 = valeurs extrêmes de l'indice J, qui sont fonctions de S2.

σ_x et σ_y = écarts - types de x et y. Les formules utilisées sont celles valables en probabilité, ou en statistique sur la totalité d'une population. Pour obtenir les écarts - types en statistique sur un échantillon, utiliser SDEV(1) et SDEV(2).

a et b = les coefficients de l'équation $y = ax + b$ de la droite de régression affine.

REMARQUES DIVERSES :

■ Caractères accentués, et autres :

Dans certains messages, ce programme utilise des caractères de codes supérieurs ou égaux à 128, définis avec l'instruction CHARSET.

Il s'agit des lettres accentuées à et é, de la moyenne \bar{y} , et de l'indice i .

Par exemple, le codage du dessin de cet indice i peut être : CHR\$(Φ) & CHR\$(32) & CHR\$(224) & CHR\$(128) & CHR\$(Φ) & CHR\$(Φ).

De plus, sont utilisés les caractères \bar{x} , σ et Σ de codes respectifs 2, 9 et 28.

■ Ligne n° 50 :

Cette ligne évite que l'utilisateur du programme réutilise l'ancien tableau S Φ des sommes statistiques, dans le cas où ce tableau n'a pas été créé, ou a été détruit, ou a été redimensionné pour un autre usage.

Mais cette ligne doit être supprimée si le HP 71 B n'est pas muni du module MATH, car la fonction TYPE est incluse dans ce module.

■ Fenêtre :

Si l'utilisateur arrête le programme lors d'une entrée de donnée statistique (double appui sur la touche ON), l'écran reste sous le WINDOW 9 de la ligne n° 220.

Deux solutions pour revenir au plein écran : taper WINDOW 1 END LINE,

ou appuyer sur RUN pour exécuter le début du programme et en particulier la ligne n° 20.

DERNIÈRE MINUTE : ENCORE UN BOGUE !

En faisant quelques essais avant d'envoyer cet article, je viens de constater la présence d'un autre bogue tout aussi inadmissible que le premier (voir le milieu de la page où figure le paragraphe intitulé "Une impossibilité éventuelle, ..."), vu qu'ils sont l'un et l'autre volontaire !!!

Les fonctions statistiques (en particulier MEAN) donnent le message "Invalid Statistic" (erreur n° 52) quand la somme des coefficients est négative.

Visiblement, ce sont les auteurs de la partie statistique du HP 71 B qui ignoraient qu'on puisse vouloir chercher le barycentre d'un ensemble de points munis de coefficients ayant des signes quelconques.

Pour corriger ce bogue, il faudrait remplacer MEAN(1) par S Φ (1)/S Φ (Φ), MEAN(2) par S Φ (3)/S Φ (Φ), etc..., dans les lignes n° 750, 810, etc...

Auteur : M^{re} Dominique Tinel (PPC-7 n° 695)
9, rue Maurice Jacob, 47000 AGEN.

COMPTA

COMPTA 1966-67

Le présent rapport expose les résultats de l'activité de l'Association pour l'année 1966-67. Elle a été établie en vertu de la loi n° 10 du 15 mai 1966 relative à l'organisation de l'enseignement technique et professionnel. Elle a été établie en vertu de la loi n° 10 du 15 mai 1966 relative à l'organisation de l'enseignement technique et professionnel.

Le présent rapport expose les résultats de l'activité de l'Association pour l'année 1966-67. Elle a été établie en vertu de la loi n° 10 du 15 mai 1966 relative à l'organisation de l'enseignement technique et professionnel. Elle a été établie en vertu de la loi n° 10 du 15 mai 1966 relative à l'organisation de l'enseignement technique et professionnel.

HP-75

Le présent rapport expose les résultats de l'activité de l'Association pour l'année 1966-67. Elle a été établie en vertu de la loi n° 10 du 15 mai 1966 relative à l'organisation de l'enseignement technique et professionnel. Elle a été établie en vertu de la loi n° 10 du 15 mai 1966 relative à l'organisation de l'enseignement technique et professionnel.

Le présent rapport expose les résultats de l'activité de l'Association pour l'année 1966-67. Elle a été établie en vertu de la loi n° 10 du 15 mai 1966 relative à l'organisation de l'enseignement technique et professionnel. Elle a été établie en vertu de la loi n° 10 du 15 mai 1966 relative à l'organisation de l'enseignement technique et professionnel.

Le présent rapport expose les résultats de l'activité de l'Association pour l'année 1966-67. Elle a été établie en vertu de la loi n° 10 du 15 mai 1966 relative à l'organisation de l'enseignement technique et professionnel. Elle a été établie en vertu de la loi n° 10 du 15 mai 1966 relative à l'organisation de l'enseignement technique et professionnel.

Le présent rapport expose les résultats de l'activité de l'Association pour l'année 1966-67. Elle a été établie en vertu de la loi n° 10 du 15 mai 1966 relative à l'organisation de l'enseignement technique et professionnel. Elle a été établie en vertu de la loi n° 10 du 15 mai 1966 relative à l'organisation de l'enseignement technique et professionnel.

Le présent rapport expose les résultats de l'activité de l'Association pour l'année 1966-67. Elle a été établie en vertu de la loi n° 10 du 15 mai 1966 relative à l'organisation de l'enseignement technique et professionnel. Elle a été établie en vertu de la loi n° 10 du 15 mai 1966 relative à l'organisation de l'enseignement technique et professionnel.

COMPTA

COMPTA HP-75

Je vous présente cette fois-ci un programme qui m'a servi pendant près d'un an pour faire la comptabilité des Editions du Cagire. Ce programme est loin d'être génial, il a le plus souvent été écrit sous la pression des circonstances.

Malgré ses limitations, je crois que c'est un assez bon exemple de programmation structurée. Il fait appel au module I/O, et serait plus "lourd" à programmer sans lui, mais ce ne serait pas très difficile. Son principal inconvénient est l'utilisation considérable du lecteur de cassette qui le rend très lent. Avec le lecteur de disquettes, ce doit être parfait. Ceci est dû à la faible mémoire du HP-75 qui oblige à travailler en "overlay". Avec une disquette (et MASSLEX) il serait possible d'utiliser directement un fichier de données sur disquette.

Le premier programme est appelé "COMPTA". Il sert simplement de menu et dirige l'exécution des autres. Il a été écrit pour usage avec une interface vidéo, mais il est simple de s'en passer.

Ligne 10 c'est le nom du programme, ligne 20 le dimensionnement des variables. Ce dimensionnement est souvent négligé dans les programmes, c'est dommage car non seulement il permet un gain appréciable de place mémoire (surtout pour les variables alphanumériques, en BASIC HP) mais il donne une liste des variables utilisées (ceci dit, c'est toujours la dernière chose écrite dans un programme).

Le programme propose une désignation de la mémoire de masse et de l'imprimante, que vous êtes libre de modifier si votre configuration a changé. L'imprimante 80 colonnes est presque indispensable.

La mienne étant une imprimante adaptée, sa désignation est celle du convertisseur (:11).

Les lignes 60 à 120 assurent l'affichage rapide du menu sur la vidéo, elle sont simplement à supprimer (et à garder sur une fiche cartonnée) si vous n'avez pas la vidéo.

les lignes suivantes assurent l'exécution des sous programmes. Cette disposition permet très facilement l'adjonction d'éléments nouveaux au menu.

L'utilisation de CALL permet le chargement direct depuis la mémoire de masse du programme appelé. Le paramètre passé est le nom de l'unité de mémoire. Après usage, le programme est effacé et il y a retour au menu. Le programme d'impression est constitué de trois modules.

Les différents modules disponibles permettent de CRBER une fiche de saisie de données. Chaque fiche est constituée de lignes de cahier de compte (voir plus loin). Il est possible d'interrompre la saisie quand on veut et de mélanger les mois. Il n'y a pas de classement par date, mais il serait facile de l'ajouter.

Les programmes ASS et REG permettent de regrouper les écritures d'un mois donné, quelque soit la fiche dans laquelle elles ont été entrées. IMP imprime le grand livre, IMP2 le livre de banque et IMP3 la ventilation de la TVA, le but principal du programme. MOD est une ébauche de programme de correction, très insuffisante.

Voyons tout cela en détails.

CRE:

La chaîne B\$ est le "masque de saisie" par ligne, ligne 40 se place la saisie du nom du fichier à créer (attention, si le fichier existe déjà sur cassette il sera effacé). EDC signifie Editions Du Cagire, remplacez le par votre nom, 8 est le début de l'année (84 85...). x est un numéro d'ordre. Remarquez la décom-

position de la chaîne d'appel qui permet de placer le curseur sur le numéro de l'année. Les caractères 30 et 31 (ligne 50) font passer en double largeur, ils sont à adapter à l'imprimante (ou à supprimer). La ligne 70 ouvre le fichier et la ligne 80 y place un modèle de la ligne d'appel.

La ligne 90 est un bon exemple de la puissance du module I/O. L'instruction MAP remplace dans la chaîne B\$ les caractères : et - par leur version à code augmenté de 128 (soulignés à l'affichage du 75, en vidéo inverse à la vidéo) qui en fait des caractères intouchables à l'affichage par TEMPLATE (le curseur les saute). Quand TEMPLATE retourne la chaîne ces caractères sont "redevenus normaux". Pour extraire les informations de la chaîne, il faut donc garder une trace précise (au brouillon) de la position des diverses zones. Les lignes 110 à 130 vérifient si la sortie de la chaîne est normale ou vide (cas de fin de traitement). La ligne est aussitôt imprimée pour contrôle. Si la saisie est terminée, le fichier est recopié sur cassette.

ASS :

Ce programme doit "assembler" deux fichiers. Je l'avais prévu au cas où la saisie des données serait fréquemment interrompue. En fait je ne m'en suis presque pas servi.

Il commence par demander le nom des deux fichiers à assembler, puis les lit en mémoire. Il renumérote le fichier 2 à partir de 1000 (il y a sûrement moins de 1000 lignes dans le fichier 1). Je ne sais plus à quoi sert le PRINT \$ 1,1 ! il faut bien vous laisser chercher un peu ! Il suffit ensuite de renumérotter (mais à partir de 1) le fichier numéro 1 et de les assembler. Un dernier renumérotage, un peu de nettoyage et (facultatif) une "impression de contrôle" sur la vidéo. Souvenez-vous que le gestion d'écran du 75 est lente et qu'il vaut mieux PRINTER sur la vidéo que DISPLAYer.

REG :

REG sert à regrouper dans un même fichier toutes les opérations ayant rapport à une même année et à un même mois. A ce propos, faites attention que les mêmes noms de fichier sont utilisés d'une année sur l'autre, n'oubliez pas de changer de cassette en même temps que d'année !

Ce regroupement s'effectue en faisant un balayage sur les fichiers de saisie, il faut que les numéros desdits fichiers se suivent, sinon il y aura un arrêt sur erreur et il n'est pas sûr que frapper "CONT" résoudra le problème.

Nous allons donc traiter successivement tous les fichiers de saisie, tester le contenu de toutes les écritures et ajouter au fichier définitif les écritures concernées.

IMP :

IMP assure l'impression des comptes du mois en cours. La ligne 30 permet le tracé du cadre, les CHR\$(30) et CHR\$(31) des lignes 120 et 140 ne visent qu'à faire passer en double largeur le texte intermédiaire, ils peuvent avantageusement être supprimés si vous ne savez pas quelle imprimante sera utilisée en définitif. Ensuite tous le fichier est balayé et les lignes formatées pour l'impression. Je n'ai pas trouvé le moyen, avec l'instruction "IMAGE" de ne pas faire imprimer les zéros du côté où il n'y a pas de chiffre.

N'oubliez pas, si ce n'était pas encore évident pour vous que l'introduction des données dans ce programme n'est pas usuelle en comptabilité. Traditionnellement, on compte 5 écritures comptables par opération : il y a l'inscription au compte du client ou du fournisseur lors de l'achat ou de la vente, l'inscription en ventilation comptable (compte 6 ou 7) et celle de la TVA. Lors du règlement il faut encore mouvementer le compte du client ou du fournisseur et le compte en banque ou la caisse. Ce type d'écritures est un peu laborieux mais permet de nombreux contrôles qui

réduisent très fort les risques d'erreur.

Il ne m'a pas paru envisageable d'effectuer 5 écritures par opération au clavier du 75. J'ai donc réduit de 5 à 1. Mais dans l'unique ligne d'écriture je porte tous les renseignements, avec même un petit supplément pour contrôle. La ligne comporte en effet (vous le voyez par exemple ligne 30 du programme CRE) la date, la référence de l'opération, le commentaire, le montant (accompagné de son signe, il faut donc taper si c'est une dépense), le numéro de ventilation comptable, le code de compte en banque ou caisse et le taux de TVA.

Le "petit supplément" vient de ce que la plus part des comptes de ventilation n'admettent que des opérations en crédit (ou en débit, mais pas les deux) et qu'il est donc possible de contrôler ainsi le signe du montant. Remarquez que je n'ai pas prévu le code client/fournisseur. Si vous voulez ultérieurement ventiler selon ces comptes, rajoutez cette rubrique.

Par contre, lors de l'impression des comptes, j'utilise la forme usuelle en débit et crédit sans signe (ici appelés Fournisseurs et clients), comme vous pouvez le voir ligne 110.

Ceci permet, dans la boucle 200200 de tester un seul emplacement pour le montant, de calculer les totaux débit et crédit ligne 230, d'imprimer l'opération (lignes 240 et 250), puis dès la sortie de la boucle d'imprimer ces totaux et le solde.

IMP2 :

IMP2 imprime la ventilation par compte en banque et caisse. Ceci permet, par rapprochement d'avec les relevés de banque, de vérifier que rien n'a été oublié. Cela permet aussi de vérifier la caisse. Vous remarquerez ici que la caisse s'appelle JDD, j'ai tellement investi dans ma société que je ne risque pas de lui devoir de l'argent ! Comme dit précédemment, ces programmes ont été vite écrits, sous la pression du besoin et j'ai employé une méthode "brute" qui n'en est pas moins efficace. Il n'y a guère de commentaires à faire à ce propos.

IMP3 :

IMP3 est le but ultime qui a justifié la réalisation des programmes : obtenir la ventilation de la TVA pour la déclaration d'impôt trimestrielle. Ceci est fait simplement en calculant à partir du montant TTC le montant de la TVA. Il n'y a pas de commentaire à faire sur le programme. Disons quand même que cette méthode a un défaut sérieux et pour moi inattendu : On est parfois amené à réaliser des opérations où le taux de TVA ne correspond à aucun des taux réglementaires. J'ai rencontré ce cas lors d'opérations avec l'étranger. Vous recevez un colis venu de l'étranger. Ce colis est ouvert par la douane, son contenu inspecté, sa valeur comparée avec la facture qui doit obligatoirement être jointe. Première difficulté, la facture n'est pas toujours dans le colis, vous devez donc la produire avant que l'on vous donne le colis. La valeur du colis est donc estimée sur facture ou d'autorité, en général dans la monnaie du pays d'origine (par exemple en \$US), ce montant est traduit en francs au cours du jour et la TVA calculée sur le résultat, on y ajoute les frais de douane et payez. Mais il vous faut ensuite transmettre le tout à votre banquier pour règlement au fournisseur, règlement qui peut avoir lieu plusieurs semaines après (ou avant !) la réception du colis. Vous savez à quel point les taux de change sont fluctuants, et voilà pourquoi le montant de la TVA ne correspond pas au montant de la facture. J'ai résolu ce problème par des astuces (en comptant une partie de la facture à 18.6% de TVA et une autre partie hors taxes, mais il m'est arrivé de payer plus de 18.6%, comment faire !). Je vous laisse voir si vous serez jamais dans ce cas. Pour mon compte, j'ai laissé le HP75 d'abord pour DBASEIII sur SANYO avec un fonctionnement presque identique, puis pour un vrai programme de comptabilité, beaucoup plus performant (et beaucoup plus cher !).

JeanDaniel Dodin

```

>copy":ml"
>display is *
10 ! COMPTA enchainement prog compta
20 DIM I$(31),M$(3),M1$(4)
30 INTEGER R
40 INPUT "La memoire de masse est ",":ml";M$
50 INPUT "L'imprimante est ",":i1";I$
60 PRINTER IS ":d1"
70 PRINT
80 PRINT "1 Creer une fiche" @ PRINT
90 PRINT "2 Assembler deux fiches" @ PRINT
100 PRINT "3 regrouper par mois" @ PRINT
110 PRINT "4 Imprimer un mois" @ PRINT
120 PRINT "5 Modifier un fichier" @ PRINT
130 PRINTER IS I$
140 INPUT "Votre reponse 1, 2, 3, 4 ou 5.:";R
150 ON R GOSUB 160,170,180,190,220
160 CALL "CRE"&M$ ; ( M$ ) @ PURGE "cre" @ GOTO 60
170 CALL "ASS"&M$ ; ( M$ ) @ PURGE "ass" @ GOTO 60
180 CALL "REG"&M$ ; ( M$ ) @ PURGE "reg" @ GOTO 60
190 CALL "IMP"&M$ ; ( M$,M1$ ) @ PURGE "IMP"
200 CALL "IMP2"&M$ ; ( M$,M1$ ) @ PURGE "IMP2"
210 CALL "imp3"&M$ ; ( M$,M1$ ) @ PURGE "IMP3" @ GOTO 60
220 CALL "MOD"&M$ ; ( M$ ) @ PURGE "MOD" @ GOTO 60

```

```

10 SUB 'CRE' ( M$ )
15 ! creation de fichier
20 DIM A$(64),B$(64)
30 Bs="!jj-mm-aa:ref :com :mont :num :sur :tv :l"
40 INPUT "Nom du fichier : EDC8","4x"; N$ @ N$="EDC8"&N$
50 PRINT TAB(20);CHR$(31);N$;CHR$(30)
60 PRINT
70 ASSIGN # 1 TO N$
80 PRINT # 1 ; B$
90 Bs=MAPS('!-','n','B$(1,64)')
100 As=TEMPLATES(Bs,'p')
110 IF As=' ' THEN GOTO 160
120 IF As=B$ THEN 100
130 PRINT # 1 ; As
140 PRINT As
150 DISP "Ligne suivante" @ GOTO 100
160 DISP "Je recopie "&N$ @ COPY N$ TO M$
170 PURGE N$
180 DISP "termine"

```

```

10 SUB 'ASS' ( M$ )
20 DIM A$(96)
30 INPUT "Premier fichier ? EDC8","4x"; N1$ @ N1$="EDC8"&N1$
40 INPUT "Deuxieme fichier : EDC8","4x"; N2$ @ N2$="EDC8"&N2$
50 DISP "Je lis "&N1$
60 COPY M$ TO N1$
70 DISP "Je lis "&N2$
80 COPY M$ TO N2$
90 EDIT N2$
100 ASSIGN # 1 TO N2$ @ PRINT # 1,1
110 RENUMBER 1000,1
120 DISP "J'ajoute "&N1$&" a "&N2$
130 EDIT N1$
140 RENUMBER 1,1 @ MERGE N2$ @ RENUMBER 1,1
150 DISP "J'efface "&N2$
160 PURGE N2$
170 PURGE N2$&M$
180 DISP "Je recopie "&N1$
190 COPY N1$ TO M$
200 PRINTER IS ":d1" @ PLIST N1$ @ ASSIGN LOOP
210 DISP "Je finis le nettoyage !!!"
220 PURGE N1$
230 DISP "Termine"

```



```

10 SUB 'IMP2' ( M$,M1$ )
20 DIM A$(64),B$(80)
30 REAL D,E,F,D1,D2,E1,E2,F1,F2,D3,E3,F3
40 INTEGER I
50 PRINT TAB(20);CHR$(31);"ETAT BANCAIRE";CHR$(30)
60 PRINT
70 IMAGE " |-----|-----|-----|-----|-----|-----|-----|-----|
-|"
80 PRINT USING 70
90 PRINT " : Banque populaire : S B M P : Associe JDD
:"
100 PRINT " :Ref : credit : debit : credit : debit : credit : debit
:"
110 PRINT USING 70
120 D=0 @ E=0 @ F=0 @ D3=0 @ E3=0 @ F3=0
130 FOR I=1 TO LASTLN?(M1$)
140 READ # 1, I ; A$
150 B1$=UPRC$(A$(55,58))
160 V=VAL(A$(36,45))
170 B1$=RTRIM$( ' ',B1$)
180 B1$=LTRIM$( ' ',B1$)
190 IF B1$(1,2)="BP" AND V>0 THEN D1=V @ D2=0 @ E1=0 @ E2=0 @ F1=0 @ F2=0 @ GOTO
260
200 IF B1$(1,2)="BP" AND V<=0 THEN D1=0 @ D2=-V @ E1=0 @ E2=0 @ F1=0 @ F2=0 @ GO
TO 260
210 IF B1$="SBMP" AND V>0 THEN D1=0 @ D2=0 @ E1=V @ E2=0 @ F1=0 @ F2=0 @ GOTO 26
0
220 IF B1$="SBMP" AND V<=0 THEN D1=0 @ D2=0 @ E1=0 @ E2=-V @ F1=0 @ F2=0 @ GOTO
260
230 IF B1$="JDD" AND V>0 THEN D1=0 @ D2=0 @ E1=0 @ E2=0 @ F1=V @ F2=0 @ GOTO 260
240 IF B1$="JDD" AND V<=0 THEN D1=0 @ D2=0 @ E1=0 @ E2=0 @ F1=0 @ F2=-V @ GOTO 2
60
250 BEEP @ DISP "Erreur" @ STOP
260 D=D+D1 @ D3=D3+D2 @ E=E+E1 @ E3=E3+E2 @ F=F+F1 @ F3=F3+F2
270 PRINT USING 360 ; A$(10,15),D1,D2,E1,E2,F1,F2
280 NEXT I
290 PRINT USING 70
300 PRINT USING 360 ; " !tot! ", D, D3, E, E3, F, F3
310 PRINT USING 70
320 D=D-D3 @ E=E-E3 @ F=F-F3
330 PRINT USING 360 ; " !Res.!", MAX(0,D),-MIN(0,D),MAX(0,E),-MIN(0,E),MAX(0,F),-M
IN(0,F)
340 PRINT USING 70
350 PRINT @ PRINT
360 IMAGE 6a,7dr2d,"!",7dr2d,"!",7dr2d,"!",7dr2d,"!",7dr2d,"!",7dr2d,"!"

```


Mois No5

GRAND CAHIER

jj-mm-aa	ref	commentaire	Fournis.	Clients	numero	sur	tva
01-05-84	136	ACHAT FIG	338,45	0,00	60710	BP	00
01-05-84	137	PAPIER IMP	72,00	0,00	6079	JDD	18.6
02-05-84	138	PHOTOCOPIES	148,30	0,00	61811	JDD	18.6
03-05-84	139	ETRANGER	0,00	150,00	7014	SBMP	EXP
03-05-84	139	LIVRES	0,00	325,00	7012	SBMP	7
04-05-84	140	PETIT	1056,53	0,00	6079	BP	18.6
06-05-84	141	LIVRES	0,00	1128,00	7012	SBMP	7
06-05-84	141	PROGRAMMES	0,00	150,00	7072	SBMP	18.6
06-05-84	141	CELF	0,00	60,75	7014	SBMP	EXP
04-05-84	142	FNAC	0,00	325,00	7012	SBMP	7
06-05-84	143	EYROLLES	431,49	0,00	6073	BP	7
06-05-84	144	TBI CASSETTES	2893,84	0,00	6079	BP	18.6
07-05-84	145	NAVILLE	0,00	121,50	7014	JDD	EXP
07-05-84	146	PAYOT	0,00	138,79	7014	JDD	EXP
07-05-84	147	LIB DU MIDI	0,00	381,79	7014	JDD	EXP
09-05-84	148	SYNTHETIX	51,09	0,00	60710	BP	00
09-05-84	148	FRAIS BANQUE	59,65	0,00	627	BP	18.6
09-05-84	148	SYNTHETIX	1989,25	0,00	6079	BP	18.6
09-05-84	149	PAYOT	0,00	151,87	7014	JDD	EXP
10-05-84	150	PUB OP	3676,60	0,00	6231	BP	18.6
12-05-84	151	PHOTOCOPIES	378,25	0,00	61811	JDD	18.6
12-05-84	152	PUB TRANSACTION	109,20	0,00	6231	BP	18.6
12-05-84	153	URSSAF	599,00	0,00	646	JDD	00
14-05-84	154	LIVRES	0,00	398,00	7012	BP	7
14-05-84	154	PROGRAMMES	0,00	600,00	7072	BP	18.6
14-05-84	154	LIVRES	0,00	32,50	7012	BP	7
14-05-84	155	PTT	1993,93	0,00	6261	BP	00
15-05-84	156	BERANGER	0,00	903,13	7014	JDD	EXP
15-05-84	157	LEPEBVRE	0,00	30,37	7014	JDD	EXP
17-05-84	158	SCRIPTORIA NV	0,00	121,50	7014	JDD	EXP
17-05-84	159	PPC	0,00	1616,05	7073	BP	00
17-05-84	159	PPC	0,00	2893,84	7072	BP	18.6
17-05-84	159	LIVRES	0,00	198,00	7012	BP	7
17-05-84	159	LIVRES	0,00	408,50	7012	BP	7
17-05-84	159	LIVRES	0,00	150,00	7014	BP	EXP
18-05-84	160	PHOTOCOPIES	606,00	0,00	61811	JDD	18.6
21-05-84	161	LIVRES	0,00	498,00	7012	SBMP	7
21-05-84	162	LIVRES	0,00	130,00	7012	SBMP	7
28-05-84	163	LIVRES	0,00	1697,50	7012	BP	7
28-05-84	163	PROGRAMME	0,00	180,00	7072	BP	18.6
26-05-84	165	PHOTOCOPIES	148,30	0,00	61811	JDD	18.6
28-05-84	166	PHOTOCOPIES	127,20	0,00	61811	JDD	18.6
28-05-84	167	LIVRES	0,00	2601,00	7012	BP	7
28-05-84	167	ETRANGER	0,00	65,00	7014	BP	EXP
jj-mm-aa	ref	commentaire	Fournis.	Clients	numero	sur	tva
		TOTAL	14679,08	15456,09			
		BALANCE	0,00	777,01			

ETAT BANCAIRE

Ref	Banque populaires		S B M P		Associe JDD	
	credit	debit	credit	debit	credit	debit
136	0,00	338,45	0,00	0,00	0,00	0,00
137	0,00	0,00	0,00	0,00	0,00	72,00
138	0,00	0,00	0,00	0,00	0,00	148,30
139	0,00	0,00	150,00	0,00	0,00	0,00
139	0,00	0,00	325,00	0,00	0,00	0,00
140	0,00	1056,53	0,00	0,00	0,00	0,00
141	0,00	0,00	1128,00	0,00	0,00	0,00
141	0,00	0,00	150,00	0,00	0,00	0,00
141	0,00	0,00	60,75	0,00	0,00	0,00
142	0,00	0,00	325,00	0,00	0,00	0,00
143	0,00	431,49	0,00	0,00	0,00	0,00
144	0,00	2893,84	0,00	0,00	0,00	0,00
145	0,00	0,00	0,00	0,00	121,50	0,00
146	0,00	0,00	0,00	0,00	138,79	0,00
147	0,00	0,00	0,00	0,00	381,79	0,00
148	0,00	51,09	0,00	0,00	0,00	0,00
148	0,00	59,65	0,00	0,00	0,00	0,00
148	0,00	1989,25	0,00	0,00	0,00	0,00
149	0,00	0,00	0,00	0,00	151,87	0,00
150	0,00	3676,60	0,00	0,00	0,00	0,00
151	0,00	0,00	0,00	0,00	0,00	378,25
152	0,00	109,20	0,00	0,00	0,00	0,00
153	0,00	0,00	0,00	0,00	0,00	599,00
154	398,00	0,00	0,00	0,00	0,00	0,00
154	600,00	0,00	0,00	0,00	0,00	0,00
154	32,50	0,00	0,00	0,00	0,00	0,00
155	0,00	1993,93	0,00	0,00	0,00	0,00
156	0,00	0,00	0,00	0,00	903,13	0,00
157	0,00	0,00	0,00	0,00	30,37	0,00
158	0,00	0,00	0,00	0,00	121,50	0,00
159	1616,05	0,00	0,00	0,00	0,00	0,00
159	2893,84	0,00	0,00	0,00	0,00	0,00
159	198,00	0,00	0,00	0,00	0,00	0,00
159	408,50	0,00	0,00	0,00	0,00	0,00
159	150,00	0,00	0,00	0,00	0,00	0,00
160	0,00	0,00	0,00	0,00	0,00	606,00
161	0,00	0,00	498,00	0,00	0,00	0,00
162	0,00	0,00	130,00	0,00	0,00	0,00
163	1697,50	0,00	0,00	0,00	0,00	0,00
163	180,00	0,00	0,00	0,00	0,00	0,00
165	0,00	0,00	0,00	0,00	0,00	148,30
166	0,00	0,00	0,00	0,00	0,00	127,20
167	2601,00	0,00	0,00	0,00	0,00	0,00
167	65,00	0,00	0,00	0,00	0,00	0,00
tot:	10840,39	12600,03	2766,75	0,00	1848,95	2079,05
Res:	0,00	1759,64	2766,75	0,00	0,00	230,10

VENTILATION TVA

	CLIENTS			FOURNISSEURS		
	TTC	HT	TVA	TTC	HT	TVA
TVA 18,6%	3823,84	3224,15	599,69	11265,12	9498,41	1766,71
TVA 7%	7741,50	7235,05	506,45	431,49	403,26	28,23
TVA 0%	1616,05	1616,05	0,00	2982,47	2982,47	0,00
EXPORT	2274,70	2274,70	0,00	0,00	0,00	0,00
TOTAL	15456,09	14340,95	1106,14	14679,08	12884,15	1794,93

PEDAGOGIE

INFORMATION
INITIATION AU LANGAGE FORTH
A L'EGRÈS DES JOURNÉES
PAR J.-D. BOUILLON
L'Équipe pédagogique de l'École Supérieure

1. Introduction

Le langage FORTH est un langage de programmation orienté objet, qui permet de définir des procédures et des données de manière très concise. Il est particulièrement adapté aux systèmes embarqués et aux applications temps réel.

Il existe de nombreux dialectes de FORTH, mais le dialecte le plus répandu est le FORTH standard. Ce dialecte est défini dans le livre de référence "FORTH: The Practical Approach" de Charles H. Moore.

FORTH

Le langage FORTH est un langage de programmation orienté objet, qui permet de définir des procédures et des données de manière très concise. Il est particulièrement adapté aux systèmes embarqués et aux applications temps réel.

Il existe de nombreux dialectes de FORTH, mais le dialecte le plus répandu est le FORTH standard. Ce dialecte est défini dans le livre de référence "FORTH: The Practical Approach" de Charles H. Moore.

Le langage FORTH est un langage de programmation orienté objet, qui permet de définir des procédures et des données de manière très concise. Il est particulièrement adapté aux systèmes embarqués et aux applications temps réel.

Il existe de nombreux dialectes de FORTH, mais le dialecte le plus répandu est le FORTH standard. Ce dialecte est défini dans le livre de référence "FORTH: The Practical Approach" de Charles H. Moore.

Le langage FORTH est un langage de programmation orienté objet, qui permet de définir des procédures et des données de manière très concise. Il est particulièrement adapté aux systèmes embarqués et aux applications temps réel.

PEDAGOGIE

INFORMATIQUE
INITIATION AU LANGAGE FORTH
A L'USAGE DES LYCEENS
par J-D Dodin
(lignes générales et idées forces)

1) Généralités

Un Ordinateur, pour ce qui nous concerne, est composé de :

- * un écran
- * un clavier
- * une "caisse" baptisée unité centrale
- * une mémoire de masse (disquettes)

L'unité de mémoire étant l'octet (8 éléments binaires ou bits) et un octet ou caractère correspondant à peu près à une lettre de l'alphabet, une disquette peut contenir entre 64000 octets (21 pages de livre) et 1 200 000 octets (un livre de 400 pages) selon l'ordinateur qui l'utilise.

Cette mémoire de masse est rendue nécessaire par le fait que les ordinateurs oublient tout ce qu'ils savent quand on coupe le courant, alors que la disquette, comme un magnétophone, conserve les informations.

* une mémoire vive (celle qui meurt quand on coupe le courant!) qui, sur notre appareil vaut 64000 octets.

Pour que cet ordinateur puisse fonctionner, il faut des logiciels (encore appelés programmes). Rien n'est plus bête ni plus inutile qu'un ordinateur sans programme !

Quand l'ordinateur est éteint, sa mémoire vive est morte (!) et ne contient donc pas de programme. Dans ces conditions, allumer l'appareil ne sert à rien. On utilise donc un autre type de mémoire, dite mémoire morte qui, contrairement à ce qu'on pourrait penser ne perd jamais son contenu (elle doit son nom au fait qu'il n'est pas possible de modifier ce contenu).

Cette mémoire morte contient un programme appelé chargeur, très bête, qui sait juste lire une partie bien précise de la disquette et transmettre l'exécution à ce qu'il vient de lire qui doit donc être un autre programme. Cet autre programme continue le travail et finit de charger ce que l'on appelle le système d'exploitation des disquettes. Ce nouveau programme permet d'examiner le contenu des disquettes, de copier des fichiers d'une disquette sur une autre, etc... et enfin (et surtout) d'exécuter un programme par simple appel de son nom.

Avec un ordinateur professionnel, en effet, les programmes s'exécutent seuls, ils sont dits exécutoires.

Mais ces programmes, il faut les fabriquer. Ceci se fait en chargeant un langage (BASIC, PASCAL, LSE, PL1, LISP, PROLOG, programme exécutable).

Il y a 3 ou 4 couches de programmes qui se recouvrent en mémoire : Chargeur, SED, langage ou progiciel (logiciel professionnel). Notre but, ici, est d'étudier le langage FORTH.

2) Le langage FORTH

Tout programme (et un langage est un programme) prends des données, travaille sur ces données selon un algorithme puis rends de nouvelles données, selon le schéma :

Données	programme	Données
source	----->	résultat

Tout programme peut donc être caractérisé par la façon dont il prends les données sources et par la façon dont il les traite.

2.1) Organisation des données : La pile

En Forth les données sont traitées sur une pile (comme une pile d'assiettes) la dernière donnée introduite est la première utilisable on dit : dernier entré, premier sorti.

2.2) Organisation des programmes : Les mots

En FORTH, toutes les instructions données à un programme sont des MOTS, comme en français, séparés par un blanc. La nature même du mot (son mode de fonctionnement) n'a pas à être connue de l'utilisateur. De ce fait, un me mot peut, au cours de la mise au point du programme, voir sa nature changer sans que le resce du programme soit à modifier.

Dans ces conditions, un mot est défini par le contenu de la pile avant et après son exécution, il y a peu ou pas de règles de syntaxe.

Les mots du langage sont rangés dans ... un dictionnaire.

Exemple : ESSAI (n1 --- n2)

 u u u u
Le mot -----' u u ----- après exécution, il y a n2
n1 se trouve sur-----sépare sur la pile
la pile au début avant d'après

2.3) Les modes de fonctionnement :

Un langage peut fonctionner sous deux modes : ou bien interpréteur : il lit le texte du programme ou ce qui est tapé au clavier et l'exécute immédiatement ou bien compilateur : il lit le texte du programme et le transforme en code exécutable.

En générale ces deux modes sont incompatibles, un langage est interpréteur ou compilateur. Le FORTH peulture les deux de façon mélangée et il allie les avantages des deux méthodes.

2.4) Applications, première leçon :

Nous allons d'abord mettre en route les appareils :
Allumer, taper (en majuscules) B @ (on voit B:@) et

presser la touche de validation.

Première épreuve : trouver comment répondre à la question sur la date. A la demande, taper :

MSUP

MSUP

LINA

ZFORTHNG

Quand apparaît OK, vous êtes sous FORTH.

Deuxième épreuve : utiliser le mot FORTH (point). Mettre un nombre sur la pile :

12 valider

taper .

voir

12

- Quel est le plus grand et le plus petit nombre que l'on puisse entrer dans la pile FORTH et afficher avec , sans qu'il soit modifié ? (réponse :

- Même question avec l'instruction U.
résultat :

- Me question avec D. (mais il faut alors que, lors de la frappe du nombre au moins un point soit tapé :

.123456

12.3456

123456.

.1.2.3.456

Ont pour forth la même signification et s'affichent 123456 avec D.

réponse :

Conclusion ?

2.5) Applications, deuxième leçon :

Cette leçon va être consacrée à la manipulation de la pile et à l'utilisation de l'éditeur plein écran.

Nous avons vu lors de la leçon précédente que le forth enregistre les données sur une pile. Nous avons vu également que selon que l'on utilise , U. ou D. pour lire le nombre, l'interprétation du contenu de la pile est différent. On utilise plusieurs mots permettant de manipuler le contenu de cette pile. Par exemple :

ROT rotation

SWAP échange

PICK prelever (doit être précédé d'un nombre comme 1 ou 5)

DUP dupliquer

Objet de la manipulation : trouver l'action des mots précédents sur la pile.

Méthode : entrer 5 ou 6 nombres sur la pile, par exem-

ple :

12 23 35 54 78 89 valider
Utiliser S. pour examiner le contenu de la pile
S. 12 23 35 54 78 89
exécuter le mot :
ROT

examiner à nouveau la pile :

S. 12 23 35 78 89 54

Répéter l'opération plusieurs fois, en tirant les conclusions et essayer les autres mots. Ne pas oublier que PICK doit être utilisé sous la forme "1 PICK" ou "5 PICK" (1, 5 ou n'importe quel nombre, mais quelle est la validité du résultat ?).

L'EDITEUR

Le mot "Editeur", dans le sens où il est utilisé ici est un anglicisme horrible. En anglais, "to edit" signifie "corriger". Les tout premiers programmes étaient affreusement difficiles à corriger, du coup les programmeurs ont réalisé des programmes de "correction" de programmes. Ces programmes ont évolué vers ce que l'on appelle maintenant des traitements de texte. Mais les programmeurs, dans leur jargon continuent à parler d'éditeur...

Avec le ZFORTH du MICRAL 80-22 (auteur Olivier Singla), l'éditeur est un "overlay". Il faut donc que le chargement du langage ait été correctement effectué (pas d'affichage du message "il faut recompiler les overlay" au chargement du FORTH.

Il faut alors ouvrir un chemin entre vous et la disquette. Ceci se fait par :

" A:NOMGROUP" 2 ASSIGN

Vous pouvez alors commencer à travailler :

1 EDIT

Cette opération amène à l'affichage le dessin d'un carré de 16 lignes et 64 colonnes entouré de l'affichage d'un certain nombre de renseignements utiles.

C'est dans ce carré que nous allons écrire. Il représente une unité de stockage FORTH, appelée ECRAN.

Pressez la touche sortie vous verrez l'explication du fonctionnement de l'éditeur.

Vous pouvez quitter l'éditeur en maintenant pressée la touche CTRL (celle qui possède un voyant) et en pressant la touche Q. Un Bip retentit et une phrase s'affiche. Inutile de la lire, validez.

Vous êtes revenus sous FORTH. Pour exécuter ce que vous venez d'écrire dans l'écran, il suffit de faire :

1 LOAD

Mais il faut d'abord avoir écrit quelque chose ...

Donc faites 1 EDIT.

Pour commencer le travail sur éditeur, nous allons utiliser une astuce du FORTH.

Tapeons comme première ligne de l'écran :
FORGET DEBUT : DEBUT ;

Beaucoup de commentaires sur cette ligne. FORGET signifie "oublie" en anglais. Cette commande provoque l'effacement de la mémoire du mot qui suit (ici DEBUT), et de tout ce qui avait été fait après.

Quand on fait des essais, il faut souvent effacer les essais précédents. Au tout début du travail, et une seule fois, nous allons taper "à la main" : DEBUT ;

Il faut taper tout, y compris les " ;".

Nous avons vu que le forth peut être interpréteur ou compilateur.

Il est interpréteur en temps normal. Mais quand il rencontre le mot " ;", il passe en mode compilateur. Ce qui suit n'est plus exécuté mais enregistré dans le dictionnaire.

Quand le forth rencontre le mot " ;" il comprend que la compilation est terminée et qu'il faut à nouveau interpréter.

Nous allons utiliser l'éditeur pour découvrir de nouvelles instructions.

D'abord essayez WORDS (pas de commentaire).

Revenons à l'éditeur (1 EDIT)

Nous revoyons ce que nous avons déjà écrit (FORGET...)

Ajoutons dessous :

: SEC 1000 0 DO LOOP ;

faites 1 LOAD

Essayez WORDS, SEC que se passe-t-il ? (sans doute pas grand chose).

Revenez à l'éditeur et ajoutez :

: MINUTE 60 0 DO I . LOOP ;

Sortez de l'éditeur, 1 LOAD, essayez MINUTE. Que se passe-t-il ?

Vous devez voir 0 1 2 3 4 ...

Les instructions DO LOOP peuvent être traduites par FAIRE BOUCLE, I. (attention, un espace entre I et .) affiche le numéro de la boucle.

Le forth compte les boucles entre 0 et 59 et entre 0 et 1000.

Avez-vous compris que SEC est une "temporisation" (fait elle une seconde ? comment obtenir ce résultat ?)

Avez-vous compris ce que fait MINUTE ?

2.6 Application, troisième leçon.

Le but de la troisième et de la quatrième leçon est de mettre en oeuvre ce qui a été vu précédemment, plus quelques mots nouveaux.

Nous allons : afficher notre chronomètre au milieu de l'écran, avec secondes et minutes sous la forme MM:SS, et prévoir la possibilité d'interrompre le fonctionnement en pressant sur une touche.

Les mots nouveaux vont apparaître dans les écrans ci-

dessous. Afficher au milieu de l'écran.

Notre FORTH utilise un mot !CURS (se lit : stocker curseur). Ce mot doit être précédé du numéro de ligne et du numéro de colonne représentant la position de l'écran o l'affichage doit avoir lieu.

La définition est :
!CURS (l n ---)
4 8 !CURS place le curseur à la ligne 4, colonne 8 de l'écran, tout ce qui sera écrit ensuite le sera à partir de cette position.

Interrompt le fonctionnement.

L'instruction LEAVE permet de sortir d'une boucle avant la fin normale. L'instruction ?KEY permet de savoir si une touche a été pressée. L'instruction IF (le SI anglais) permet d'utiliser le renseignement fourni, de la façon suivante :

```
?KEY IF LEAVE ENDIF
IF ... ENDIF fonctionne comme des parenthèses, ce qui est
à l'intérieur est exécuté si la condition est réalisée.
On arrive donc à l'écran suivant :
```

```
FORGET DEBUT : DEBUT ;
: SEC 1000 0 DO LOOP ; < TEMPORISATION ACCELEREE POUR LES ESSAIS >
: MINUTE ( --- ) < affichage de 60 secondes consécutives >
  60 0
  DO ( LA DUREE DE LA BOUCLE )
    DO ( DEBUT DE LA BOUCLE )
      4 8 !CURS ( CURSEUR POSITIONNE PAR EXEMPLE EN 4, 8 )
      I . ( AFFICHAGE )
    ?KEY IF LEAVE ENDIF ( FAUT-IL FINIR ? )
    SEC ( TEMPORISATION DE 1 SECONDE )
  LOOP ( FIN DE LA BOUCLE )
;
--> ( PASSER A L'ECRAN SUIVANT )
```

Dans l'éditeur, on passe d'un écran à l'écran suivant en tapant CTRL D et à l'écran précédent en tapant CTRL S.

```
: MINUTES ( --- ) < affichage des minutes >
  60 0
  DO
    4 4 !CURS
    I .
    ." : " < CECI SIGNIFIE : AFFICHER DEUX POINTS >
  ?KEY IF LEAVE ENDIF
  MINUTE < AFFICHAGE DES SECONDES >
  LOOP
;
-->
```

Troisième écran, un mot de démarrage, pour simplifier le travail :

```
: TOP
  CLS < CECI SIGNIFIE : EFFACER L'ECRAN >
  MINUTES
;
TOP < CECI VA LANCER L'EXECUTION UNE PREMIERE FOIS >
  ( LORS DU CHARGEMENT )
```

INITIATION AU FORTH

COMMENTAIRES ET CONCLUSIONS

Cette initiation au langage FORTH a eu lieu dans deux classes, l'une de seconde T1 et l'autre de première F1 en Octobre et Novembre 1985.

Elle s'est déroulée par demi-classes, sur les horaires des cours d'Agencement pour la classe de 2T1 et sur les horaires de Construction pour la classe de première. Les groupes étaient de 11 et 12 élèves pour la classe de 1ère et de 15 et 16 élèves pour la classe de seconde.

Les matériels utilisés étaient 7 Micral 80.22 non graphiques et le logiciel était le ZFORHNG fourni par Mr Singla de l'Ecole Normale de Montauban, tournant sous système d'exploitation EDEN.

L'initiation était prévue pour chaque groupe sur 4 séances de deux heures, à raison d'une séance par semaine pour les élèves de première et une séance tous les 15 jours pour les élèves de seconde.

La durée a volontairement été réduite pour ne pas empiéter excessivement sur les cours et pour obliger les élèves à un effort prononcé.

En pratique l'initiation a dû être interrompue après la troisième séance pour des raisons matérielles : les ordinateurs utilisés sont à bout de souffle et plusieurs lecteurs de disquettes sont hors d'usage, de plus, pour des raisons dont je ne puis dire si elles sont liées au matériel ou au logiciel, nous avons expérimenté de nombreuses erreurs d'écriture sur disquette rendant la disquette inutilisable avec l'éditeur, à moins d'un reformatage complet avec perte des données (le cas s'est produit au moins 6 fois au cours des séances).

Un contenu résumé des séances est donné en annexe, il a été distribué aux élèves après la séance.

Pour l'essentiel les élèves connaissaient peu l'ordinateur et pas du tout le langage FORTH. Leur motivation a été variable. La classe de première a été aux deux extrêmes : l'un des groupes a remarquablement travaillé, se trouvant en avance sur le planning prévu, l'autre a fait montre de si peu d'empressement que l'initiation a été arrêtée au milieu de la deuxième séance.

Il faut en effet remarquer que, faute de sanction sous forme de note, un minimum (et même un maximum) de motivation est requis des élèves.

La classe de seconde n'a pas manqué de motivation et a même abordé les premières séances avec de meilleurs résultats que la première. Cependant cette avance a rapidement été perdue.

Comme on pourra le remarquer à la lecture des descriptifs, un des buts des premières séances, au delà de l'initiation au FORTH était une pratique de la recherche expérimentale, facilitée par la structure du langage. En effet dans ce langage les données de l'expérimentation (sur les instructions du langage lui-même) ainsi que les résultats sont soit visibles sur l'écran soit situés en un emplacement (la "pile") facile à examiner. Le cycle mise en place des données, exécution de l'instruction, examen des résultats est donc simple.

Sous cet aspect il est apparu que la méthode expérimentale

reste mal comprise de la plus part des élèves. On peut estimer qu'au départ seuls quelques élèves (10% ?) auraient pu expérimenter seuls. Après deux séances, je pense que la moitié des élèves avaient compris la méthode (dans la cadre de cette application). Le travail par groupes de 2 ou 3 élèves ne permet pas de préciser davantage. La proportion est meilleure en première, moins bonne en seconde.

La deuxième remarque importante est que les élèves sont bloqués par la procédure, il est vrai assez contraignante, permettant le démarrage du FORTH. Au cours des premières séances ils n'ont fait aucun effort pour noter ce qui était indiqué au tableau et se sont trouvés dans l'impossibilité de démarrer à la séance suivante. Ceci s'est beaucoup amélioré au fil des séances, mais il a été nécessaire en seconde de réviser le contenu de la séance précédente, ce qui est pour l'essentiel la cause du retard.

Il semble cependant, au vu des commentaires d'élèves ci-joints que les contraintes d'usage de l'ordinateur, mais aussi ses possibilités aient été assez bien saisies par les élèves, on peut donc considérer cette initiation comme une réussite.

En sus de l'évaluation subjective de l'activité des élèves en cours de séance un commentaire écrit a été demandé à un groupe d'élèves de seconde pris au hasard. Il leur a été demandé d'écrire en 1/4 d'heure, sur une page ou 15-20 lignes au maximum ce que cette initiation leur avait inspiré. Aucun sujet n'était imposé, l'anonymat était possible. Il me semble impossible de résumer ces devoirs qui figurent ci-dessous. J'ai essayé de respecter l'orthographe et la mise en page des élèves.

Le professeur
J-D Dodin

les devoirs, présentés sans ordre particulier :

Premier élève, devoir anonyme :

Le stage était trop court et très intéressant, très intéressant très intéressant, très intéressant

Au départ j'ai été déçu de ne pas faire du basic et, à part les tirs d'entraînement sur cible mouvante et que l'on a pas pu faire le dessin d'une montre comme l'autre groupe. On était trop sur une machine car tout le monde se battait pour taper quelque chose sur l'écran

Remarque : les "tirs" ne font pas référence à un jeu informatique, mais à la guerre à la sarbacane qui fait fureur entre élèves à certaines périodes...

Deuxième élève, anonyme

Ce que l'on pense du stage d'informatique

Je pense que l'informatique, de nos jours, est très utile dans beaucoup de domaines; pour résoudre des problèmes au niveau de la gestion, par exemple; dans presque toutes les entreprises, il y a plusieurs ordinateurs qui sont utilisés pour ces raisons. De plus

... c'est une voie de communication, en effet, par ordinateur une communication est possible d'une ville à une autre et les grosses entreprises sont continuellement en ligne avec l'étranger grâce aux ordinateurs.

Je vais maintenant parler des ordinateurs personnels qui ont fait fureur ces dernières années. En effet, il y a beaucoup de particuliers qui ont chez eux un ordinateur pour classer des adresses, pour gérer les dépenses... mais aussi pour s'amuser (cassettes de jeux vidéo)

L'informatique fournit donc beaucoup de possibilités pour s'en servir. Elle devient indispensable de nos jours.

Je trouve donc très intéressant de faire des stages d'informatique au lycée, pour se familiariser avec les ordinateurs. Lors du stage, nous avons monté un petit programme permettant de faire fonctionner un petit chronomètre...

En résumé, c'était une première approche assez intéressante pour des gens qui ne connaissaient pas très bien ce domaine.

troisième élève, anonyme

Ce que l'on pense de ce que l'on a fait en informatique aime bien l'informatique, travailler sur les ordinateurs ça me plaît. J'en avais déjà fait l'an dernier en travaillant sur le Basic, mais mes connaissances étaient très succinctes. ce que l'on a fait, c'était très intéressant, on a appris un nouveau langage le Forth. Les ordinateurs qu'il y a ici sont assez gros donc on peut faire beaucoup de choses. Le Forth est à mon avis moins intéressant que le Basic pour des gens de notre âge mais il est plus difficile à assimiler. Puis de nos jours la plus part des entreprises sont équipée d'ordinateurs. De plus le Forth est un langage que utilise peut-être les entreprises. Maintenant le Français moyen à son mini-ordinateur chez lui. De nos jours on se sert des ordinateurs pour classer les courriers des entreprises. Je pense que le langage Forth est quand-même intéressant

quatrième élève, nom communiqué

Ces séances m'ont plus car j'aime beaucoup l'informatique. La seule remarque à faire se situe sur le langage. j'aurais aimé travailler sur Basic qui est un langage plus usuel que le Forth. Je pense qu'il faudrait approfondir le Forth et voir plus loin. Car pour apprécier un langage il faut avancer assez loin dedans.

cinquième élève, anonyme

C'était intéressant mais vu le nombre de disquette abimé par le groupe devant nous c'est normal que si chacun en grille autant on soit obligé d'arrêter. Il faudrait, si on le peu, recommencer un autre stage de 3 ou 4 semaines pour mieux savoir se servir d'un clavier et mieux exploiter les possibilités de l'ordinateur. Mis à part les tirs d'artillerie tout s'est bien passé et les

ordinateurs ont ouvert, pour ceux qui ne l'avait pas déjà franchit, une autre porte vers l'avenir.

sixième élève, anonyme

Le stage est beaucoup trop court et très intéressant. Au départ j'ai été déçu de ne pas faire du Basic et puis avec les leçons que l'on a faites j'ai compris que c'était beaucoup plus simple

Ce que j'aimerais savoir c'est si l'on peut avoir des ordinateurs dans le commerce

septième élève, anonyme

Je dois dire que je préfère étudier les programmes et apprendre à me servir d'un ordinateur que d'étudier l'agencement. C'est moins fatiguant.

Plus tard quand on travaillera dans une entreprise on ne se servira pas des jeux entre les pièces ni des différentes sortes de liaisons qu'il existe. On travaillera, aidé par les ordinateurs. Je pense qu'on doit nous apprendre à se servir d'un ordinateur sinon on ne sera pas au même niveau que les pays industrialisés de l'Europe.

huitième élève, anonyme

L'informatique me passionne moyennement, je ne suis pas très fort et le Forth est un langage assez difficile. Cependant je préfère étudier l'informatique que l'agencement. Plus le temps passera plus l'informatique va devenir une matière comme les autres alors il faut plutôt s'intéresser sur ce chemin là. Malgré le petit nombre d'ordinateurs qu'il y a à ce lycée je pense que nous pouvons quand même retenir quelque base d'informatique que nous aurons appris dans ce nouveau domaine.

neuvième élève, anonyme

Savoir se servir d'un ordinateur est un avantage dans notre vie actuelle car plus les années passent, plus l'informatique prend une grande place dans notre vie. L'homme ne servira plus qu'à programmer les ordinateurs car les machines feront le travail à notre place.

Le Forth est un langage assez difficile cependant je préfère apprendre l'informatique qu'apprendre des cours d'agencement.

dixième élève, nom communiqué

J'ai appris en 3ème la Basic, cet année ce fut le Forth. Je l'est trouvé plus simple que le Basic. Ca m'a appris à me servir d'un ordinateur différent de celui de l'année dernière. L'ordinateur est une machine qui me plaît et avec ces séances m'a appris peut-être les métiers de l'avenir. Même pour le dessin industrielle l'ordinateur commence à prendre la place de l'homme donc il faudra peut-être savoir s'en servir. Vu que j'étais intéressé par l'informatique je suis tout "ému" de ne pas finir la séance.

onzième élève, nom communiqué

OPINION SUR L'INFORMATIQUE

Je trouve que c'est une très bonne chose d'avoir fait de l'informatique puisque dans toute usine et même dans la vie courante l'ordinateur est toujours présent. Ça serait dommage pour un jeune de notre époque de ne pas avoir taper une seule fois sur le clavier d'un ordinateur et ça serait beaucoup plus enrichissant pour lui de savoir le programmer. Mais de plus en plus je vois que dans n'importe quel travail il est venu s'imposer (exemple : pharmacie et n'importe quel commerce). Je pense que maintenant savoir programmer un ordinateur fait partie du bagage qu'a besoin n'importe quel technique quelque soit sa discipline. Ce qui est une bonne chose pour moi aussi c'est que dès le bas âge on mette dans les mains d'un enfant un petit ordinateur comme le MO5 ou le TO7 (ordinateur de base très connue); ceci est une bonne base pour n'importe quel futur technicien en informatique. Autre idée dans la scolarité qui a apparue il y a a peu près 2 ans qui est très bénéfique c'est d'apprendre à faire de l'informatique dès l'école primaire.

douzième élève, nom communiqué

J'ai beaucoup apprécié ce stage d'informatique car bientôt, cela nous sera très utile dans la vie, l'informatique est le futur. Il est fort dommage que nous n'ayons pas pu faire le double de séances car pour le moment, nous ne savons pas faire grand-chose. Nous sommes des débutants.

Et je pense que ce serait bien qu'en cours d'année, quand la salle sera vide, que nous passions 2 heures à "tapoter" sur les ordinateurs

treizième élève, nom communiqué

Ces séances d'informatique m'ont appris le fonctionnement d'un ordinateur. L'informatique me plaît et peut-être aurai-je bientôt un ordinateur.

Je trouve que l'informatique prend une grande place dans la vie. Ces séances m'ont plu car on a appris un peu d'informatique sur les ordinateurs car certains apprennent l'informatique en théorie.

quatorzième élève, anonyme

Informatique

C'était intéressant car on a pu voir ce qu'est l'informatique. Moi qui n'ai jamais fait d'informatique j'ai pu trouver que ce n'est pas si évident que quand on nous le présente soit dans la publicité soit dans les magazines.

Je préfère le Basic par rapport au Forth. Par exemple avec Logo on peut apprendre à faire des figures géométriques très compliquées. De nos jours l'informatique a une grande place dans la vie.

quinzième élève, anonyme

C'était la première fois que je prenais un ordinateur en main et je dois dire que ça m'a beaucoup plu. Se qui a été le plus intéressant durant ses 2 ou 3 stages c'est le programme. Comment on pouvait faire quelque chose avec si peu de connaissance. J'aurai aimé poursuivre cela et essayer de programmer un jeu simple.

Le seul inconvénient de tout cela s'est que l'on a pas pu avoir un ordinateur pour une personne. pour un début je pense que l'ordinateur que l'on avait été trop perfectionné pour nous. vu que nos capacités sur la connaissance de l'ordinateur étaient minimes.

seizième élève, nom communiqué

Pour moi, l'informatique c'est plutôt quelque chose de compliqué. Les 3 séances ne m'ont rien apporté, j'oublie chaque fois, ce qu'on a fait précédemment, je m'embrouille avec toutes les touches qu'il y a; la seule touche que je souvient à chaque séance, c'est la touche "valider".

Sa montre que j'ai raison de prendre technique, car j'ai hésité en 3e entre informatique et P2.

Je dois dire que je préfère étudier les programmes et apprendre à me servir d'un ordinateur que d'étudier l'agencement. C'est moins fatiguant. Plus tard quand on travaillera dans une entreprise on ne se servira pas des penseurs les pièces ni des ordinateurs, se pense qu'on doit nous apprendre à se servir d'un ordinateur sinon on ne sera pas au même niveau que les pays industrialisés de l'Europe.

C'était intéressant moi on le nombre de diquette ainsi on le groupe devant nous, c'est normal que si ^{il y a} ~~il y a~~ quelle autour on soit obligé de s'arrêter.

Il faudrait, si on le veut, recommencer avec un autre stage de 2 ou 4 semaines en mieux tenir compte de la sensibilité de l'ordinateur.

Mais à part les tris d'attribution tout est très bien passé et les ordinateurs ont marché, pour ceux qui ne l'ont pas déjà franchit, avec une sorte de l'écrit.

CALCULS EN REGLE

UNE REGLE A CALCULS POUR FORTH

Un article de Nathaniel GROSSMAN,
Dpt Mathematics UCLA Los Angeles CA 90024
traduit par Christophe LAVARENNE.

Si la plupart des utilisateurs d'arithmétique en virgule flottante ne ressentent pas toujours le besoin d'une extension complète en virgule flottante, ils peuvent trouver des applications pour une "règle à calculs". De petite taille, une telle extension permettrait le calcul, en virgule flottante simple précision, des fonctions mathématiques usuelles, y compris addition et soustraction. Comme avec une règle à calculs, le domaine de validité des fonctions serait limité, laissant à l'utilisateur le soin des calculs d'échelle. Voici une implémentation de règle à calculs pour Forth, commençant par les quatre opérations de base implémentées par Martin Tracy, les autres fonctions mathématiques étant dérivées de l'algorithme CORDIC unifié.

Du temps de la plume, du papier et des tables numériques, la règle à calculs facilita les travaux de l'Homme. Cet instrument, qu'on pourrait présenter comme un ancêtre, non miniaturisé, d'une puce interactive intégrant les fonctions arithmétiques du calcul flottant, a permis d'économiser un nombre considérable d'heures qui auraient été consacrées à de longues divisions et à d'incessantes utilisations de tables numériques. Quand les calculateurs mécaniques, puis électroniques devinrent disponibles, puis courants, les calculs arithmétiques à grande précision ne devinrent pas plus difficiles que ceux à faible précision, ces derniers se résumant simplement à arrondir ou à tronquer les résultats en grande précision.

Pour le concepteur de matériel ou de logiciel, le passage de faible à grande précision n'est pas une étape triviale. L'utilisateur, pour qui l'implémentation est souvent transparente, peut se plaindre d'avoir à supporter le temps nécessaire à un calcul en haute précision pour obtenir un résultat en faible précision. De même, lors du transport d'une application ne nécessitant que du calcul en faible précision, les efforts nécessaires au transport des outils de calcul en haute précision semblent disproportionnés.

Cet article présente une implémentation générale de calcul de l'algorithme de CORDIC permettant à lui seul de générer les fonctions mathématiques usuelles, y compris multiplication et division si nécessaire. Un algorithme unifié est des plus attirant pour un concepteur de matériel. Les

calculatrices de poche Hewlett-Packard sont équipées d'un processeur CORDIC câblé [1,6]. Pour le concepteur d'outils de calcul en virgule flottante, CORDIC est également intéressant. En général, pour des raisons de vitesse d'exécution, CORDIC est souvent implémenté en virgule fixe, ce qui réclame des mises à l'échelle avant et après exécution. Les opérations de mise à l'échelle sont reportées dans une routine en soi très simple. Ceux qui ont utilisé une règle à calculs se rappelleront que les mises à l'échelle étaient laissées entièrement aux soins de l'utilisateur par calcul mental. L'implémentation présentée ici se situe à mi-chemin, sacrifiant la vitesse d'exécution au profit d'un calcul en virgule flottante simplifiant les problèmes de mise à l'échelle. Elle est tirée d'une extension compacte en simple précision que Martin Tracy a écrite et baptisée "Maths Zen" pour rire, aussi avons nous baptisé "Règle à calculs Zen" les écrans que nous y avons ajoutés. La règle à calculs Zen, comme la "vraie", laisse à l'utilisateur le soin de vérifier le domaine d'utilisation des différentes fonctions.

Le reste de cet article décrit l'algorithme CORDIC puis commente les écrans présentant notre implémentation. Nous n'entrerons pas ici dans les détails de démonstration du fonctionnement de CORDIC: ceux qui s'y intéressent pourront se reporter à la littérature existante. L'introduction simple de Schelin [5] est sans doute le meilleur point de départ. Schelin donne une longue bibliographie sur CORDIC et d'autres sujets s'y rapportant; entre autres, les articles de Schmid et Bogacki [6], Volder (le concepteur de CORDIC) [8], et Walther [9] ont été les plus utilisés pour la préparation de ce document. Les articles de Furman [3] et de Freese [4] dans "Forth Dimensions" contiennent une implémentation CORDIC en virgule fixe pour les fonctions Sinus et Cosinus en Forth. Nous avons intégré pratiquement toutes les fonctions élémentaires usuelles citées par Duncan et Tracy [2] dans leur proposition de standard en virgule flottante.

L'ALGORITHME CORDIC.

Comme le fait remarquer Schelin [5], l'algorithme CORDIC est remarquable par le fait qu'il ne fait pas appel aux techniques du calcul différentiel ou intégral pour calculer les valeurs des fonctions élémentaires. Etant données trois valeurs initiales, appelées soit x_0, y_0, z_0 soit x_1, y_1, z_1 suivant la fonction à calculer, des séquences de triplets x_k, y_k, z_k sont générées suivant le schéma de récurrence suivant:

$$x_{k+1} = x_k - m \cdot \text{delta}_k \cdot y_k \cdot 2^{-k}$$

$$y_{k+1} = y_k + \text{delta}_k \cdot x_k \cdot 2^{-k}$$

$$z_{k+1} = z_k - \text{delta}_k \cdot \text{epsilon}_k$$

Partant de $k=0$ ou 1 comme approprié, les itérations doivent être poursuivies jusqu'à $k=n$. Le paramètre m (appelé le mode) vaut +1 ou -1. Les nombres epsilon_k, se présentent sous deux aspects, présents dans les tableaux +EPS et -EPS:

+EPS: epsilon_k = tan⁻¹(2^{-k}), pour $k=0, 1, \dots, n$

-EPS: epsilon_k = tanh⁻¹(2^{-k}), pour $k=1, \dots, n$

Dans notre implémentation, $n=14$. Les nombres delta_k sont choisis égaux soit à +1 soit à -1 de manière à forcer soit y_k soit z_k vers zéro. Si z_k est forcé vers zéro, le processus est appelé rotation, sinon si y_k est forcé vers zéro il est appelé vectorisation. Une obscure condition technique sur les epsilons exige que les itérations pour $k=4$ et $k=13$ soient répétées et $k=0$ soit omise, pour la vectorisation seulement. Enfin, deux constantes sont nécessaires à l'initialisation des récurrences:

$$1/K = (1+4^{-0})^{1/2} * (1+4^{-1})^{1/2} * \dots$$

$$* (1+4^{-14})^{1/2} = 0.6073$$

$$1/K' = (1-4^{-1})^{1/2} * (1-4^{-2})^{1/2} * \dots$$

$$<4 \text{ répétées} \dots <13 \text{ répétées}$$

$$* (1-4^{-14})^{1/2} = 1.2076$$

Les instructions pour évaluer les fonctions spécifiques sont données dans le tableau ci-dessous, qui est adapté de celui donné par Schelin [5]. Le cas $m=0$ qui peut être utilisé pour effectuer les multiplications et les divisions est omis ici parce qu'il n'est pas utile quand +, -, * et / sont déjà implémentés en virgule flottante.

Le schéma CORDIC en binaire

Rotation ($z_k \rightarrow 0$)

delta_k = +1 si $0 < z_k$, -1 si $z_k < 0$

$m=+1$: $x_0=1/K, y_0=0, z_0=1$ *donnant*

$$x_{n+1} = \cos(t), y_{n+1} = \sin(t)$$

$m=-1$: $x_0=1/K', y_0=0, z_0=1$ *donnant*

$$x_{n+1} = \cosh(t), y_{n+1} = \sinh(t)$$

$$t^k = x_{n+1} + y_{n+1}$$

Vectorisation ($y_k \rightarrow 0$)

delta_k = +1 si $y_k < 0$, -1 si $0 < y_k$

$m=+1$: x_1 donnée, y_1 donnée, $z_1=0$ *donnant*

$$z_{n+1} = \tan^{-1}(y_1/x_1)$$

$$x_{n+1} = K * (x_1^2 + y_1^2)^{1/2}$$

$m=-1$: x_1 donnée, y_1 donnée, $z_1=0$ *donnant*

$$z_{n+1} = \tanh^{-1}(y_1/x_1)$$

$$x_{n+1} = K * (x_1^2 + y_1^2)^{1/2}$$

$$(x_1=t+1 \text{ et } y_1=t-1 \text{ donnant } \ln(t)=2 * z_{n+1})$$

$$(x_1=t+1/4 \text{ et } y_1=t-1/4 \text{ donnant } t^{1/2}=x_{n+1}/K')$$

LES ECRANS FORTH

Nous supposons qu'une implémentation de +, -, * et / est disponible en virgule flottante, accompagnée des utilitaires nécessaires aux entrées-sorties (conversion ascii-binaire). Les nombres sont supposés être manipulés dans la même pile que les entiers avec la caractéristique (l'exposant en base 10) sur un mot au sommet de la pile, et la mantisse juste en-dessous, sur un nombre non prescrit de mots. La caractéristique, comme la mantisse doivent être des entiers décimaux signés (le nombre d'itérations pour la boucle DO..LOOP la plus interne pourrait être réduit considérablement si le calcul se faisait en virgule flottante binaire, c'est-à-dire avec un exposant en base deux).

Les écrans n°6 à 15 contiennent la "règle à calculs Zen". Seul l'écran 6 y est dépendant de l'implémentation, et l'implémentation des quatre fonctions arithmétiques en virgule flottante est celle de Martin Tracy "maths Zen" [7], qui utilise une mantisse en simple précision, occupant donc deux mots avec la caractéristique. Du coup, la plupart des fonctions de manipulation de la pile ne sont que des synonymes des opérateurs double-précision (2DUP, 2!...). Les lignes 4, 5 et 6 de l'écran n°6 sont au cœur du "truc" pour diviser par 2. Si le but était de diviser par 10, le plus simple serait d'utiliser 1- pour retrancher un de la caractéristique du nombre flottant sur la pile. De même, 1- suffirait à diviser le nombre flottant par 2 si les nombre étaient représentés avec une caractéristique en puissances de 2. Ici, comme nous l'avons mentionné, nous travaillons avec un système hybride, exécutant CORDIC en virgule flottante décimale. Nous devons donc payer le désavantage qui consiste à effectuer une division par 2 en multipliant d'abord par 5 puis en divisant par 10. Pour qu'aucun chiffre significatif ne soit perdu par débordement ("overflow") lors de la multiplication, la mantisse en simple précision est tout d'abord étendue en double précision (2 mots), puis le résultat du produit est reconverti en simple précision par la fonction TRIM de Tracy. Le mot F2'N/ (r n --- r/2^n) divise le réel r par l'entier 2^n et doit effectuer les divisions par 2 nécessaires au moyen d'une boucle DO..LOOP (en virgule flottante binaire, on pourrait tout simplement écrire : F2'N/ (r n --- r/2^n) - ;). Le mot FLOAT aux lignes 12 et 14 est le discriminateur de Tracy qui transforme en réel ce qui pourrait être pris pour un entier double : 3.1416 FLOAT laisse 31416 -4 sur la pile. Enfin, FF est un synonyme servant d'abréviation pour FLOAT.

L'écran n°7 contient un assortiment de constantes et de variables utiles. Si la précision devait changer, seules les valeurs de $1/K$ et de $1/K'$ seraient à changer. Ces deux constantes pourraient être calculées avec plus de précision par les formules données ci-dessus, en ajoutant au produit les facteurs suivants, jusqu'à ce que la puissance de 4 soit nulle pour la précision recherchée (pour $1/K'$, l'itération suivante à répéter se trouve à $k=40$). Les trois variables flottantes, XX, YY et ZZ et la variable entière NDX résultent d'un choix délibéré de l'auteur pour améliorer la lisibilité du code. Les opérations d'accès mémoire en sont un peu moins efficaces, mais de toute façon, le plus gros du temps d'exécution se situe au niveau de l'addition en virgule flottante. Il sera facile de remplacer les manipulations de variables par des manipulations au niveau de la pile, mais celles-ci risquent de devenir prohibitives pour des précisions de mantisse plus importantes.

Le mot FARRAY crée une table de constantes réelles flottantes et empile à l'exécution celle sélectionnée. Il est utilisé pour générer les tables +EPS et -EPS. Le k^{ième} élément de +EPS vaut $\tan^{-1}(2^{-k})$, et celui de -EPS vaut $\tan^{-1}(2^{-k})$ (sauf pour $k=0$ pour lequel $\tan^{-1}(1)$ est indéfini, et remplacé par une valeur nulle). Pour sélectionner plus librement une des deux tables pour la valeur des epsilons à utiliser, le choix passe par EPS, un mot différencié (créé par DEFER).

Comme m et delta ne prennent que les valeurs +1 ou -1, la multiplication par l'un ou l'autre peut se résumer à un NEGATE conditionné. Les deux processus de rotation et de vectorisation définissent de manière un peu différente le choix des delta, d'où les deux mots R-DELTA et V-DELTA, et pour les accès aux variables, les mots R-STORE et V-STORE. L'unité de l'algorithme CORDIC fait reporter le choix entre les deux versions à travers le mot différencié ISTACK.

Les trois mots NEW_X, NEW_Y et NEW_Z supportent les itérations de la récurrence et représentent la partie principale des deux mots principaux, ROT'ING et VEC'ING qui forment le noyau, à travers le mot différencié DO-IT, de la boucle centrale du mot CORDIC.

A l'écran n°11, les deux premiers mots décident du mode en plaçant un drapeau et en sélectionnant la table d'epsilons appropriée. Le mot CORDIC est au cœur de tous les calculs de fonctions. La séquence NDX ! DO-IT aurait pu être factorisée, mais a été laissée telle quelle pour plus de clarté.

Les écrans n°12 à 14 proposent une sélection généreuse de fonctions en virgule flottante: cos, sin, tan, atan (trigonométriques), p>, r>p (transformations coordonnées rectangulaires-polaires), exp, cosh, sinh, tanh, atanh (hyperboliques), sqrt (racine carrée), ln (logarithme naturel). Ce qui représente l'ensemble des

fonctions spécifiées par Duncan et Tracy [2], les autres pouvant facilement se déduire de celles-ci. Ayant mentionné qu'une règle à calculs laisse à l'utilisateur la tâche de vérifier les domaines de validité des fonctions, nous donnons ceux-ci suivant Walther [9].

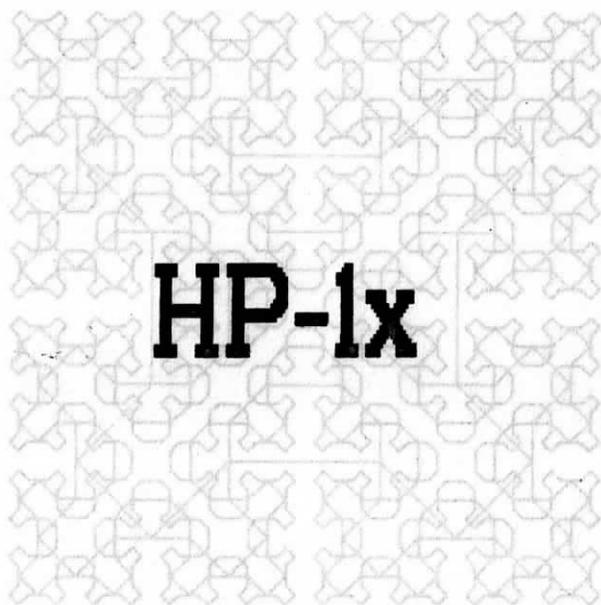
Domaines de convergence CORDIC

fonctions	(domaines)
sin, cos, tan	(-1.74, 1.74)
\tan^{-1}	(-infini, +infini)
sinh, cosh, tanh, exp	(-1.13, 1.13)
\tanh^{-1}	(-0.81, 0.81)
ln	(0.10, 9.58)
sqrt	(0.03, 2.42)

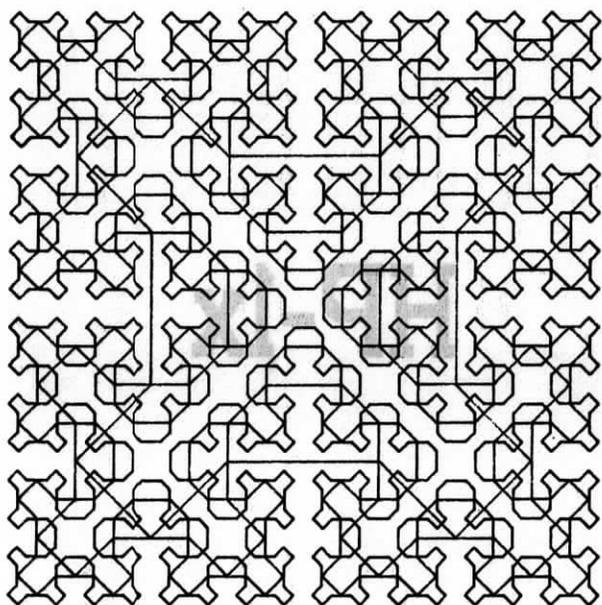
Enfin, notons que les résultats produits par la règle à calculs Zen seront précis à une ou deux unités à la troisième décimale, l'erreur étant due au manque de décimales de garde (c'est pour conserver les effets de tronquature que nous n'avons pas adopté la méthode de Schmid et Bogacki [6] qui réclamerait beaucoup plus d'additions flottantes pour obtenir la même précision). Le code donné tel quel est compilé en 1580 octets sur le MasterForth de Micromotion (une réalisation enrichie de Forth-83) sur Apple IIe. Maths Zen de Tracy réclame 500 octets supplémentaires, soit un total de 2080 octets.

REFERENCES

1. D. Cochran, Algorithms and accuracy in the HP35, HewlettPackard J. (Juin 1972, p10-11)
2. R. Duncan & M. Tracy, The FVG standard floating-point extension, DrDobb's Journal (Septembre 1984, p110-115)
3. D. Fraese, CORDIC algorithm revisited, Forth Dimensions V, n°3 p24-25
4. A.T. Furman, The CORDIC algorithm for fixed-point polar geometry, Forth Dimensions IV, n°1 p14-15
5. C.W. Schelin, Calculator function approximation, American Mathematical Monthly (Mai 1983, p317-25)
6. H. Schmid & A. Bogacki, Use decimal CORDIC for generation of many transcendental functions, Electrical Design News Magazine (Février 1973, p64-73)
7. M. Tracy, informal communication, 1984.
8. J. Volder, The CORDIC computing technique, IRE Transactions On Computers (Septembre 1958, p330-334)
9. J. Walther, A unified algorithm for elementary functions, Joint Computer Conference Proceedings 38 (printemps 1971, p379-85)



HP-1x



CALENDRIER

PHILIPPE HEILBRONN

**6, RUE FURSTENBERG
75006 PARIS**

PARIS, le 21 Décembre 1985

Monsieur Jean-Daniel DODIN,

MICRO-REVUE
77, rue de Cagire,
31100 Toulouse

Cher ami,

Je reprends ma lettre du 11 novembre 1985 qui contenait un programme de "Fonctions calendrier" pour le HP-12C, programme que vous conservez pour parution.

Je m'aperçois que l'entrée des codes N n'est pas facile à mémoriser et qu'il vaut mieux entrer, par exemple, les années 1900 en tapant 19 [ENTER]; de ce fait, je modifie légèrement le programme et je vous prie de bien vouloir noter que la présente version remplace la précédente.

Vous en souhaitant bonne réception, je vous prie de croire, cher ami, à mes sentiments cordiaux et les meilleurs.

P. Heilbronn
Philippe Heilbronn (PPCT-551)

CALENDRIER

----- Fonctions Calendrier du HP-12C -----

Le programme trouve son fondement dans l'examen comparatif des fonctions calendrier du Module TIME (du HP-41) et de celles du HP-12C, ainsi que du désir de réduire autant que possible le nombre d'appuis de touche. Le programme fonctionne aussi bien en mode M.OY qu'en mode D.MY. Pour la clarté de l'exposé, ce dernier mode est choisi.

Module TIME (HP-41) HP-12C

DATE+ DATE

DDAYS ΔDYS

DOW pas de fonction DOW au clavier,

mais accès à cette fonction par la fonction DATE

avec W de 1 (pour Lundi) à 7 (pour Dimanche).

Habituellement, pour obtenir le DOW d'une date donnée, il faut taper:

(d)d,mm,yyyy [ENTER] 0 [g][DATE] -->dd.mm.yyyy W
soit, 12 (ou 13) appuis de touche DOW

Simplification avec le programme: après introduction de celui-ci, passer en mode RUN, par appui de [f][P/R], faire [f]0 (FIX 0), puis choisir un nombre N correspondant au siècle sur lequel porteront les calculs. N est stocké en R0.

S	N	Dates de	à
1	15	1500*	1599
2	16	1600	1699
3	17	1700	1799
4	18	1800	1899
5	19	1900	1999
.....
25	39	3900	3999
26	40	4000	25 Novembre 4046

Les dates sont entrées sous le format:
(d)dmmy (différemment du format d'origine d)d.mm.yyyy
- soit une économie de 3 appuis de touche par date.

* En réalité, n'entrer des dates qu'à partir du 15 Octobre 1582 (début du calendrier grégorien). Le programme est valide pour 24 siècles entiers et pour 2 siècles partiels, le 1er (1500..) et le 26ème (4000..).
Toute introduction de date antérieure au 15 Octobre 1582 ou postérieure au 25 Novembre 4046 entraîne Error 8.

Avant de passer à des dates d'un autre siècle, éteindre et allumer le calculateur, afin de positionner le pointeur en début de programme, puis entrer le nouveau N.

Nombre de Jours entre Dates.

Le calcul est fait sur les deux dernières dates entrées qui seront rendues dans le format dd.mm.yyyy W, W est le DOW.

1. [ON]
2. Entrer le N choisi [ENTER]. (Si N est déjà en machine, aller en 3 directement).
3. Entrer la première date. (Si N est déjà en machine et qu'on vient de l'allumer, entrer la première date, appuyer sur [ENTER], puis 4 et 5).
4. [R/S] 0 apparaît à l'affichage.
5. [SST] Première date avec son DOW.
6. Entrer la deuxième date.
7. [R/S],[SST] Deuxième date avec son DOW.
8. [g][DYS] Nombre de jours entre les dates.

Exemple: Quel est le nombre de jours entre le 3 Juin 1983 et le 15 Octobre 1984. Supposer que N (19) est en machine et qu'on vient d'allumer la machine.

```
30683 [ENTER][R/S][SST]      3.06.1983 5   Vendredi
151084 [R/S][SST]           15.10.1984 1   Lundi
[g][DYS]                      500 Nombre de jours
[X<>Y]                          492 Nombre de jours comptés
                                sur un mois de 30 jours.
```

Dates passées, présentes (1900) ou futures.

1. [ON]
2. Entrer le N choisi [ENTER]. (Si N est déjà en machine, aller en 3 directement).
3. Entrer la date d'origine. (Si N est déjà en machine et qu'on vient de l'allumer, entrer la date, appuyer sur [ENTER] puis 4 et 5).
4. [R/S][SST] Date d'origine avec son DOW.
5. Entrer le nombre de jours (négativement si la date est dans le passé).
6. [g][DATE]

Exemple: Nous sommes le 14 Mai 1981. Dans 120 jours, quelle sera la date?

```
19 [ENTER]
140581 [R/S][SST]           14.05.1981 4   Jeudi
120 [g][DATE]                11.09.1981 5   Vendredi
```

REMARQUE : Il est possible d'entrer un nombre de jours qui fasse passer d'un siècle au(x) suivant(s).

Exemple: Le début du calendrier grégorien se situe le 15 Octobre 1582. Il s'écoulera 899.999 jours. Quelle sera alors la date? (fin du calendrier valide)

```
15 [ENTER]
151082 [R/S][SST]           15.10.1582 5   Vendredi
899.999 [g][DATE]           25.11.4046 7   Dimanche
```

Touches Affichage

[q][x<y]	01- 43 34
[q][GTO]06	02-4333 06
[x<y]	03- 34
[STO]0	04- 44 0
[x<y]	05- 34
[EEX]	06- 26
2	07- 2
[/]	08- 10
[q][FRAC]	09- 43 24
[q][LSTx]	10- 43 36
[q][INTG]	11- 43 25
[EEX]	12- 26
2	13- 2
[+]	14- 20
[RCL]0	15- 45 0
[+]	16- 40
[+]	17- 40
[EEX]	18- 26
4	19- 4
[/]	20- 10
0	21- 0
[R/S]	22- 31
[q][DATE]	23- 43 36
[q][GTO]06	24-4333 06

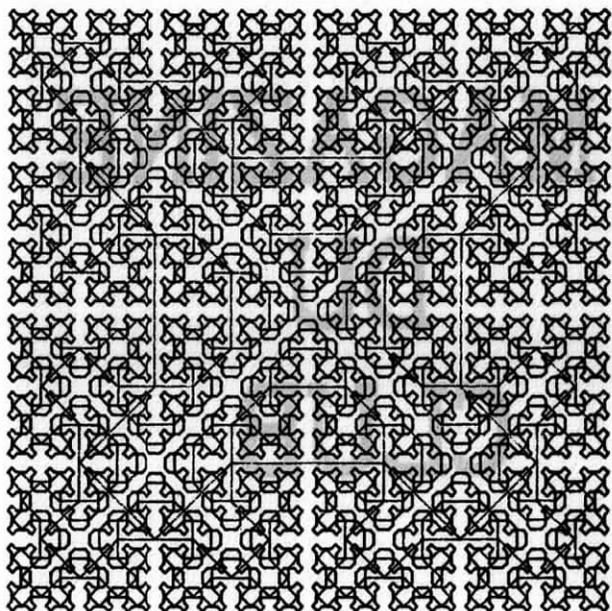
Si l'on avait besoin du registre R0 pour stocker une valeur extérieure au programme, on pourrait se servir, à la place de R0, d'un registre financier comme 'n', en remplaçant respectivement les lignes 04 et 15 par :

[n]	04- 11
.....	
[RCL][n]	15- 45 11

plus de 11
11CT-57



**MAGAZINE
DU
CLUB**



LES JOURNAUX

Reçus, depuis MICRO-REVUE No 9 :

Beaucoup de numéros de PPC-Journal : du numéro V12N6 de Juin au numéro V12N10 d'Octobre 85 (ceci est écrit le 26 Janvier 86). Ces exemplaires ont été reçus en deux envois groupés.

No de Juin : parmi les articles remarquables, "71 keyboards for the 71", une étude détaillée de différents types de claviers pouvant être adaptés au 71, deux programmes pour 71, 6 pour 41 et 3 pour 75.

No de Juillet : diverses applications 41, dont une particularité du CCD module (au fait, quelqu'un a-t-il déjà vu ce module en France ? la diffusion en semble assez mal faite car je n'en ai jamais eu d'écho) ; 4 programmes 41 et autant de programmes pour 75.

V12N8 : voir couverture ci-après
V12N8 : bon, je ne vais pas continuer les statistiques sur PPC-T, c'est sans grande importance. Sachez qu'il y a surtout des programmes, pour HP-41, HP-75 et HP-71. Le niveau de la revue est dans l'ensemble relativement bas (c'est à dire orienté débutant) sans nouveauté très marquante. La nouvelle adresse de PPC (qui annule tout ce qui a pu être publié jusqu'ici est : PPC, Po Box 90579, Long Beach, CA 90809-0579 USA. L'abonnement est de \$37 pour l'Europe (inchangé depuis des années). Il y a maintenant dans chaque numéro une feuille de cotation des articles (il faut mettre une note aux différents articles) assortie de récompenses pour les réponses.

CHHU Chronicle. Je suis arrivé au moins au numéro V2N5 de Septembre. Il me semblait même avoir reçu celui d'Octobre, mais je ne peux pas remettre la main dessus. CHHU est davantage orienté "applications" que PPC, à titre d'exemple vous trouverez ci-après la copie de la table de matières du numéro de Septembre. CHHU 2545 Camden Place Santa Anna, CA 92704 USA, abonnement \$34.75 plus \$6.25 la première année.

JEDI J'ai sous la main les numéros 18 et 20 (Janvier 86, couverture ci-après) Les animateurs font toujours un travail remarquable de régularité et d'intérêt. Je ne sais pas comment ils font pour éditer JEDI, fournir des articles à MICRO-REVUE et alimenter également MICRO-SYSTEME (ce n'est pas le moindre !). Bravo. JEDI 8 rue Poirier de Narçay 75014 Paris.

Club Spiegel No17 : Le club de Suisse alémanique continue sa croissance vers les 130 adhérents avec le 17ème numéro de leur sympathique petit journal. HP club Schweiz Postfach 8152 Glattbrugg Suisse.

HP-USER Nieuws : Ce journal (format A4 en feuilles volantes agraphées) est édité par Boekhandel Prins, l'équivalent hollandais de "La Règle à Calcul" (Binnenwatersloot 30, 2611 BK DELFT, Pays Bas), l'abonnement est de 15 florins par an pour 4 numéros, celui que j'ai entre les mains est le numéro 6 de Décembre 85. J'ai toujours trouvé le flamand plus facile à comprendre que l'allemand, peut être parce que plus proche de l'anglais. Nous procédons avec cette revue à des échanges d'articles, grâce à Piet Van Unen qui se charge des traductions dans les deux sens.

JPC : J'ai reçu les numéros d'Octobre et Novembre de JPC (numéros 28 et 29). Nos amis de PPC-Paris utilisent maintenant une Laserjet et après quelques déboires avec leur imprimeur semblent avoir trouvé quelqu'un qui sait mettre en valeur les qualités de cette imprimante. Ce n'est que justice, car l'équipe de rédaction se paie le luxe (façon de parler car ils font le travail eux même) de retaper tous les articles et de faire leur mise en page sur HP-71. Il faut le faire ! JPC, Philippe Guez, 56 rue JJ fousseau, 75001 Paris.

USER : La revue du club Danois, No5. Cette revue est fort intéressante et il y paraît souvent des articles "en pointe". Le club Danois a près de 500 adhérents et des relations avec HP-Danmark à faire rêver. Je voudrais vous encourager à vous abonner aux revues étrangères qui sont trop peu diffusées en France et beaucoup moins rébarbatives que l'on pourrait le penser, malgré la langue. PPC-Danmark Postboks 2 3500 Vaerlose.

PRISMA Octobre/Novembre et Décembre 85. C'est sans doute la plus luxueuse des revues de club. Le numéro de Décembre publie les plans d'une interface à liaison inductive pour HP-71. Cette interface commande une série de relais en décodant le mouvement de la membrane du beeper. Elle doit marcher aussi bien avec n'importe quelle calculatrice ayant un beeper piezo. Un schéma de ce type avait été publié pour HP-41 il y a bien longtemps, mais il nécessitait un transformateur spécial introuvable. L'intérêt de ce type d'interface est de relier la calculatrice à l'extérieur simplement en posant l'appareil sur le boîtier. PRISMA Postfach 2129, 6242 Kronberg 2 RFA.

DATAFILE Nos amis anglais semblent avoir de sérieux problèmes du fait du manque de disponibilité de D. Burch, l'éditeur. Du coup il y a eu un numéro triple juillet Aout Septembre, puis très récemment un numéro double Octobre/Novembre de... 20 pages. de plus, ce dernier numéro ne comporte pas d'éditorial, mais la rubrique Editeur comporte le mot "Position Vacant" et une rubrique "produced by R. Hellbusch" a été ajoutée. Espérons que nos amis se tireront de ce mauvais pas. Une Assemblée Générale est prévue le 8 Février. HPCC PoBox 59 Windlesham GU20 6EG England.

Je n'en parle généralement pas dans cette rubrique, mais nous recevons régulièrement, au titre de l'échange entre les revues, L'Ordinateur Individuel, Micro-Système, Sciences et Vie et El Ordenador Personal, variante Espagnole de l'OI.

CHHU CHRONICLE

CONTENTS

SEPTEMBER 1985

Both Cover photos show stereo images. See October issue page 2A.

- 2..HP UPDATE, Issue Masthead and hourglass time notice.
- 4..OOPS V2N1 DIS Bugs, Nick Reid's apology to David Birch.
- 4..MEMBER LETTERS - James Jasek - His HP-71 is door stop
 - Ben Fairbank - CHHU phone support Ric Jesch - Wants IL circuit board
 - C.K. Logue - Halfnut/ERAMO problems Dave Partridge - Many questions, HELP!
 - Sten Vesterli - '87 Copenhagen Conf Rene Franquinet - member drive help
 - Paul Harvell - 5 year feedback Graham Fraser - Bug in FINDID?
 - Peter Rossiter - S/C flag blocks John Dearing - 3 41 Tips and Routines
 - Hans Trixer - Better Morse Code JH David Motto - 71 EPROM, LEX ID's pg. 18d
 - Tapani Tarvinen - Returns in print Fernando Lopez-Lopez - Adv. complex nos.
- 11..HP-41 Thinkjet Barcode Gerhard kruse
- 13..Multiple Curve Fitting Program For the HP-41 Bill Kolb
- 18..Catalog Main HP-41 Memory Using The CCD ROM. Brian Walsh
- 18..41-Thinkjet Alternate Controls Mode. Falke Danielsson
- 18..HP-71 Keywords Organized! David Motto
- 23..Keyboards For the HP-71. Gary Friedman
- 28..CHHU HP-71 ROMCHHU members launch custom ROM project
- 32..HP-75 Create and MakehexRene M. Franquinet

OCTOBER 1985

- 2..September & October Issues, their covers & why combined.
- 2..HP-41 and HP-IL Thermal Printer.Bob Laughton
- 2..BoldlistJoseph K. Horn
- 3..Brother EP44 Used with HP-71?.Nick Reid
- 3..COMMERCIAL PRODUCTS - Megg Associates, PIL CLIP and HP-71 programs
- 4..HP-41 MCODE Resource Update.Jeremy Smith
- 5..CCD ROM #3 - Packing Key Assignment Registers.Gerhard Kruse
- 6..HP-IL #2 - SCOPE, LOOPVOC Errata from # 1.Larry Kramer
- 8..PLOT IT # 2 - Coordinates and Plotter Units.Bill Rudersdorf
- 9..REFLECTIONS # 3 - Small print; 9114A; Olson's Nos.; ZENWAND; PCN dies. . rjn
- 12..Robots - An HP-71 GameRune & Stefan Tobiaason
- 14..HP-41 XROM HEX Decimal EquivCarl E. Lewis
- 15..SCJ CORNER # 6 - ACSM/ASPRS conf., Surveying by faith or reason. . . Joe Bell
- 17..MCODE Mantissa Function - HP-41 MCODE program.Constantine Fantanas
- 18..NUT FILE # 1 - HP-41 Memory Checking, -Jeremy Smith takes over column
- 20..75 & 71 Math Pac Comparison.David G. Motto
- 21..EXPLORING THE 71 IDS # 5 - Worms & VirusesJohn R. Baker
- 22..HP-41 Setalm2.Case Van Horsen
- 23..TITAN FILE # 7 - Members Questions & AnswersJoseph K. Horn
- 26..Graphic Display And Screen Dump For 75(& 71)Rene M. Franquinet
- 27..Re-visiting Numeric Data Entered as Alpha (HP-41).Roger K. Heinsohn
- 27..Making Headerless HP-71 FORTH words.Bill Wickes
- 29..Application Tapes. (HP-41)Dave Jordan

PPCTM JOURNAL

The Personal Programming Center is an international users group of People Programming Computers
The original Hewlett Peckard users group

Copyright © 1985 PPC

AUGUST 1985 V12N8

PROGRAMS

Tom Langland (11872)	2	41	SUPER DETECTIVE
Robert Meals (9025)	8	41	TAPE RECORDER COUNTER CONVERSIONS
George Guy (12244)	9	41	HP-41 WRITES CHINESE
PPC-IDC Chapter Contributions	10	41	JURY TEST, L-TYPE MATCHING NETWORKS, GORK AN HP-41C GAME, CUMULATIVE BINOMIAL, "8-BALL" ELECTRONIC EXECUTIVE DECISION MAKER, POISSON DISTRIBUTION, WEIGHTED REGRESSION FOR A STRAIGHT LINE, RANDOM VIBRATION, BINARY CLOCK, LEARNING CURVE PROGRAM, HANDICAPPED RACE STARTER/TIMER, AREA OF A CIRCULAR SECTOR NORMAL DISTRIBUTION, FACTOR CONVERSIONS, BACKGAMMON DICE DUPLICATOR.
Charles Woodall (930)	18	71	REVERSING PRIVATE STATUS ON HP-71 AND HP-75 FILES
Ross Cooling (12433)	18	41	VIEW REGISTERS
Ake Gustafsson (8384)	19	41	ALARM SETTING PROGRAM
Carl Lewis (12652)	19	41	HP-41 XROM HEX DECIMAL EQUIVALENTS
Tom Lewis (12875)	20	41	41 HEXIDECIMAL ARITHMETIC
Niels Brinch (7248)	21	75	VALIRRA
Donald Hall (7144)	23	71	PLISTER
J.W. Warner (11552)	24	41	THE GRADIENT METHOD
H. H. Hays	26	87	CNVRTCDFL

REGULAR COLUMNS

CHAPTER NOTES	25	FEEDBACK.	29
S W A P	25	BITS & PIECES	30
TRADING POST	28	MEMBER LETTER	32

PPC, the Personal Programming Center, is a California non-profit public benefit corporation dedicated to advancing the applications art of personal computing. PPC is the oldest personal computing users group and publishes the PPC Journal. Address all correspondence to PPC, P.O. BOX 90579, Long Beach, California 90809 USA. Telephone (213) 498-2554

JEDI



QUE LE FORTH SOIT AVEC VOUS

JANVIER 1986



HP-USER NIEUWS



NUMMER 6
Dec. 1985

CLUB SPIEGEL

ACOSH
SS
R

'ELIN
'EXEP
'ZLOG
'ZPON
'ZPOLYP
'ZPOLYC
'ZMLRXY
'ZMLRXY2
'ZPTST
'ZTSTAT
'ZXSQEV
'ZEEFXSQ
'ZCTKK
'ZCTKKK
'ZSPEAR
'ZORND
'ZHSQD

In HP
produk
van de
den geb
Ik ben in
programma
Drogelijfe
kunnen woro
nemen.
Rest mij nog

TWEE
1) MATH/STAT PAC.

Deze module is eigen
van de reeds bestaande
de HP-41. Deze beil
liefert behalve bespa
voordeel op. De gecombi
twee modules apart.
Dat de module niet nieuw
open doet. Daarin vindt o
van de statische module en
Een funktieelijst:

NR.

uitgever:

boekhandel PRINS
Computershops delft

BINNEWATERSLOOT 30 TEL. 015. 131551/136989
2611 PV DELFT



**HEWLETT
PACKARD**

Official Dealer

PETITES ANNONCES

Notre proposition de bourse de matériel d'occasion parue dans le dernier numéro n'a suscité aucun intérêt chez les vendeurs mais a attiré quelques appels chez les "chercheurs". J'ai parfois pu les diriger vers les petites annonces non encore parues (et dont certaines, du coup, ne paraîtront jamais, car tout est vendu). Les prix de "L'argus" ont paru élevés à certains. Les "prix neuf" sont ceux du tarif HP Aout 85, sans doute le plus élevé jamais publié. Dès que possible je mettrai l'Argus sur tableur pour pouvoir l'ajuster. Il n'a, bien sûr aucun caractère obligatoire. Mon avis personnel est que, compte tenu de la qualité du matériel HP, il n'y a pas de raison de "brader" du matériel presque neuf. La seule chose qui compte, c'est que chacun y trouve son compte : l'acheteur qui paie moins cher qu'il ne le craignait et le vendeur qui récupère une partie de sa mise. En informatique, compte tenu de la baisse régulière du prix du neuf, la vente d'occasion est ruineuse. Bien sûr si vous vendez bon marché, les acheteurs ne se plaindront pas...

Leroy (T70) 3 rue Jules Ferry, Carency 62144 Mont St Elci tel 21 22 15 52 Vends lecteur de cartes (+ cartes + livre) pour HP-41. (prix ?).

Carlos Bredam, 32 rue du Gal Beuret 75015 Paris, tel 45 30 27 63, vend HP-41C (accélérée 2x), Quadram, modules Thermique, Statistique, X-F, PFC avec Manuel, Jeux, Lecteur de cartes, Imprimante 24 col, programmes, pour 4000F ou échange contre matériel HP-71.

Daniel Lamas, 3 rue des lacs, 91350 Grigny, vend pour cause de 71 : HP-41CX avec accus 2600F, X-Mémoire 600F, lecteur de cartes 1500F, Module Navigation 350F, module aviation 150F (le tout de 1984).

Gilbert Blanchet (T297) 3 rue de la Noë Rido 22190 Plérin, tel 96 74 56 89, vend boucle HP-IL 82160A neuve 640F.

V. Herliq (T139) 3 rue Berteaux Dumas, 92200 Neuilly sur seine 46 40 15 33 vend HP82162A Mai 83 (imprimante IL) 2200F, 40 cartes magnétiques HP-41 80F.

Hugues Oberlé (T443) 1 rue Beaulieu 68440 Zimmersheim vend pour HP-41 : modules X-F, 2 X-M, regroupés en un seul module 1500F, livres "Au fond de la HP-41" 80F, "Calculator tips and routines" 70F.

Franck Lebastard (T35) Appt 1260, 1 rue des Vêrchères 69120 Vaulx en Velin tel 78 80 77 34, vend MLDL pour HP-41, 4K ram+ 4K rom, prix à débattre.

Emmanuel Dupas (T422) 12 rue Pierre Curie 91390 Morsang/Orge, possesseur HP-41CV+X-F+Time, cherche membre connaissant routines pour adapter fonctions de HP-41CX sur HP-41CV.

PETITES ANNONCES

Alain GOUBAULT (441) 27 avenue de BRINONT 78100 CHATOU
 tél 45 61 99 11 (bureau) 39 52 56 01 (domicile)

vends :

- * Calculateur HP41 CX avril 84
avec nouveaux manuels, changeur et batterie 2000,00
- * lecteur de cartes août 82 500,00
- * module HP 1L modèle 82 (neuf) 500,00
- * module ZENRON décembre 84 500,00
- * module Advantage octobre 85 500,00
- * module XM août 82 250,00
- * module STAT août 82 150,00
- * module Structural Analysis août 82 150,00
- * livres (avec codes barres)
Maths de haut niveau, Génie Mécanique,
Fluids dynamics and hydraulics,
Physics, Control systems Chaque 50,00
- * MLDL Sicape 4K0
avec Epson MLE1H et VASM commentés 1200,00
- * Port extender 8 ports 400,00
- * Nombreux ouvrages sur le HP41 :
Synthetic programming (wides) Tips and Routines
(Deering), HP41/HP1L system dictionary (Reinheim)
XF c'est facile et P.S. c'est facile (Jouett)
IND41 (Pilluard) Chaque 50,00

Possibilités de prix groupés. n'hésitez plus!

Patrick Leray, 29253 Ile de Batz, vend HP-71 avec Imprimante Thinkjet, Lecteur de cartes, Module HP-1L, 200 feuilles de papier, 50 cartes magnétiques, le tout 13000F.

REACTIONS

Les logiciels, bénévolat et profession

Les articles publiés dans MICRO-REVUE sont fournis bénévolement par leurs auteurs, qu'ils en soient remerciés. Certains d'entre vous peuvent s'en étonner, sachant que les revues classiques d'informatique paient les articles.

Plusieur remarques s'imposent. La première et la plus évidente est que MICRO-REVUE n'a qu'un tout petit tirage (1000 ex pour le premier numéro) et que, par conséquent les recettes couvrent à peine les frais d'impression et de distribution. Ce petit tirage à lui seul interdit le recours à la publicité, faute d'annonceurs.

Mais il n'est pas certain que nous changerions d'avis si nous avions une diffusion supérieure. Les exemples de la presse informatique ne sont pas si démonstratifs. Hebdomadaire croule sous les programmes, mais ceux-ci ont souvent un relent de "déjà vu" et ce journal s'est récemment fait rouler en publiant des programmes recopiés sur des concurrents par des lecteurs peu scrupuleux. L'Ordinateur de Poche a abandonné ce qui était sa vocation en devenant LIST.

Il est sûr que les vrais programmes originaux (et intéressants) sont rares, dans le domaine familial ou des jeux. Le but de MICRO-REVUE n'est pas en fait de publier des programmes "prêts à l'emploi", même si nous en publions beaucoup, mais de vous aider à vous former en programmation en vous frottant à d'autres programmeurs. Le véritable jeu est dans la programmation, pas dans l'utilisation des programmes. Il y a plus à apprendre en comparant deux programmes d'auteurs différents sur le même sujet qu'en tapant en aveugle 800 lignes incompréhensibles.

Enfin il nous semble qu'il y a deux catégories de programmes : ceux qui sont (souvent) longs, très utiles, demandant une abondante documentation et surtout destinés à des non informaticiens, ces programmes demandent un gros travail de programmation et de documentation, et doivent être fournis tout prêts sur support magnétique (ou codes barre) et par conséquent coûtent cher. Un bon programme coûte souvent plus que la calculatrice qu'il utilise. Ils n'intéressent aussi qu'un public bien défini, souvent en fonction d'une profession précise.

Les autres peuvent avoir demandé des centaines d'heures de travail... pour réduire leur longueur à 10 lignes, ils seront très appréciés des amateurs la réputation de leur auteur y gagnera beaucoup et nous y trouverons des tas d'astuces à réutiliser.

Du coup, la documentation ne porte guère sur la façon d'utiliser le programme, car l'auteur suppose que vous connaissez votre machine, mais elle détaille comment le programme lui-même travaille. Ce programme n'est en général pas vendable, mais quel plaisir de le publier !

Rentrent aussi dans cette catégorie des programmes plus modestes, utiles mais trop courts pour pouvoir être commercialisés, alors il serait dommage de les garder pour soi.

Terminons pour signaler que sont parfois publiés dans les journaux du club des programmes parfaitement vendables mais gracieusement offerts par des adhérents particulièrement désintéressés. Merci à eux.

Selon la catégorie où vous vous placez, vous nous enverrez vos participations pour publication ou un entrefilet à passer dans "Sur le Marché" (gratuit). Les deux attitudes ne sont d'ailleurs nullement incompatibles.

Jean-Daniel Dodin



ERIC ANGELINI T2D 93.98.23.61
LE VALLON DES FLEURS B9
103 AV. HENRI DUNAND 06100 NICE

21/12/85

J.D. Dodin

Je reprends tardivement a ton courrier. En effet, je n'habite plus a l'adresse a laquelle tu me l'as envoyer et ce, depuis plus d'un an. J'avais signale mon changement d'adresse a plusieurs reprises au club mais comme M.R. qui m'arrive bien toutes les autres correspondances sont mal orientees.

Au sujet de mon role de responsable local, je l'assurerais tant bien que mal. Il est bien evident que si une autre personne desirait prendre ma place je lui cede sans aucun regret.

Je n'ai malheureusement pas beaucoup de temps a consacrer au club. Je suis ouvert au tout conseil ou toute reclamation.

Bonne Annee 86 au club

PS: Ce papier a lettre a été realise par PRINT SHOP sur Aple 4e.



SUR LE MARCHE

Le FORTH commence à être disponible dans des versions utilisables sur le marché français. La première version que je connais est celle du GUIC. Il est disponible pour 150F à GUIC, F. Lebastard, apt 1260, 1 rue des Vêrchères 69120 Vaulx en velin. Détails ci-dessous. Ce FORTH (pour IBM-PC ou compatibles) présente deux défauts : il est livré sans le livre qu'il vous convient de commander séparément, il utilise pour ses fichiers une disquette qui n'est pas compatible avec MS-DOS. Mais il n'est pas cher !

Vous pouvez aussi vous procurer le FORTH de Laboratory Micro-Système chez MICRO-SIGMA 10 rue de la Boétie 75008 Paris. Il coûte 2.325F hors taxes (4650F pour la version avec graphisme et déboggeur) ce qui est bien cher, mais c'est une version compatible MS-DOS, standard 83 et soignée dans son implémentation.

J'ai également reçu de A. Pirard, 5, Pirefontaine, 4052 Dolembreux, Belgique, la proposition pour un FORTH qui semble très intéressant, son seul défaut (au vu du descriptif) étant d'être au standard 79. Son prix est de \$90 pour la version complète et de \$45 pour les membres du club (faire référence à notre statut de chapitre du FIG-FORTH) pour une version allégée.

Les caractéristiques simplifiées sont : système de développement complet, peut être utilisé sur toute machine CP/M, CP/M 86, MS-DOS, Apple-DOS, Commodore 64, écrire à l'auteur pour les détails. Standard 79, compatible FIG FORTH, jeu de mots double longueur, extensions comme CASE, ?DO, DUMP, WHERE, .S etc...

Interaction complète avec le système d'exploitation, permettant d'accéder aux fichiers du système hôte en accès séquentiel ou direct (FILE, OPEN, CLOSE, LOOK, GET, PUT, READ, WRITE, READLINE, POINT, NOTE, INRECORD, INFILE, INDATA, SKIPEOR, PUTEOR, MUTEIOER, IOER, DELETE, RENAME, ...). Supporte les arbres MS-DOS.

Editeur d'écran complet, fichiers écran sous forme de fichiers textes standard, donc éditables sur votre traitement de texte favori, l'éditeur interne ne servant qu'à la mise au point. Le texte des écrans est stocké au dessus du dictionnaire (?)

Une commande DEBUG interprète le contenu du buffer d'entrée et en cas d'erreur vous place sous éditeur avec le curseur à l'endroit de l'erreur. COMGEN est un générateur d'applications qui condense votre programme compilé en supprimant les en-têtes... j'abrège. J'ai demandé une disquette d'évaluation.

UN CONVERTISSEUR DE SIGNAL CONTROLE LA BOUCLE HP-IL

(extrait de Electronics, Octobre 21, 1985) La puce NSC851 est un convertisseur de signal qui réalise le contrôle de réseau entre le bus d'un microprocesseur et une boucle d'interface Hewlett-Packard (HP-IL). L'HP-IL est un réseau local à deux fils de

SUR LE MARCHÉ

portée moyenne. Le NSC851 est vu par le micro-processeur comme 8 registres mémoire d'entrée sortie. La lecture ou l'écriture des données dans ces registres contrôle les actions sur la boucle. La puce manipule le codage des bits, la conversion série parallèle et les aspects critiques en fonction du temps du protocole de la boucle HP-IL. Il est disponible maintenant sous forme DIP et vendu à \$12 par 100 et plus.

Plusieurs remarques au sujet de l'entrefilet ci-dessus : D'abord c'est un produit National Semiconductor qui est un des plus gros fabricants de circuits intégrés et c'est à ma connaissance le premier circuit HP-IL non HP, donc une deuxième source : l'HP-IL devient une norme. Ensuite il n'est pas question de transformateur dans la notice. Ce circuit pourrait donc être beaucoup plus performant que l'IL-B3... l'Avenir le dira.

Un nouveau module pour HP-41 est annoncé, sans que je sache ce qu'il contient. Il a été étudié par Redshift Software, Wilson W. Hole, 7614 Lakecliff Way, Parker, Colorado 80134 USA.

On annonce des modules de mémoire vive à prix raisonnables et les lecteurs d'Eproms/modules pour le HP-71 et le HP-75 en plus de la HP-41. Tout ceci probablement par Corvallis MicroTechnologie 33815 Eastgate Circle Corvallis, OR 97333 USA.

Ceux d'entre vous qui cherchent du papier pour leur imprimante Thinkjet peuvent maintenant s'adresser également à Moore Paragon, 21 de Buxérioux BP235 36004 Chateauroux Cedex tel vert 16 05 27 78 11 (c'était avant le changement de numérotation, je ne connais pas le nouveau numéro). C'est un magasin de vente par correspondance de matériel informatique, vous pouvez leur demander leur catalogue (gratuit).

Un nouveau catalogue d'EDUCALC est paru (N°29), avec plein de nouveautés (mon stock de ce catalogue a fondu en une semaine, c'est un signe, ne me le demandez pas). Il y a même deux livres sur l'usage du module Advantage ! Par exemple il y a un nouveau module qui place au clavier, accessible en moins de 3 pressions de touches toutes les fonctions d'une 41CX. Prenez la peine de demander le catalogue à Educalc 27953 Cabot Road, Laguna Niguel, CA 92677 USA. Le \$ baisse, profitez-en !

FORTH GUIC

Cette version de FORTH fonctionne sur SANYO MBC 550/555 avec pour système d'exploitation MS-DOS. Le fichier FORTH.EXE constitue le point de départ qui permet d'implanter FORTH sur cette machine.

Le listing original est tiré du livre 'MVP-FORTH SOURCE LISTINGS' (volume 2) de Glen B. HAYDON & Robert E. KUNTZE publié par Mountain View Press. Le premier volume contient une description complète de tous les mots disponibles. Son titre : 'All about FORTH' de Glen B. HAYDON, toujours publié par Mountain View Press. Ce premier volume est pratiquement nécessaire pour utiliser cette version de FORTH.

Est également inclus sur la disquette le listing assembleur (FORTH.ASM) en plus du programme directement exécutable (FORTH.EXE).

NOUVELLES BREVES

La seconde disquette contient un ensemble de mots utiles (enregistrements FORTH), tels :

- VLIST : liste le vocabulaire disponible.
Ex: VLIST <rc>
- XY : positionne le curseur aux ligne Y et colonne X
similaire au Locate x,y du Basic
Ex: 10 10 XY <rc>
- THRU : Charge les enregistrements.
Ex: 20 30 THRU <rc>
charge les enregistrements 20 à 30.

Pour charger ces mots, vous tapez sous FORTH :
20 LOAD <rc>

Avant toute chose, ne travaillez jamais avec vos deux disquettes Originales contenant FORTH : Faites des copies de Sauve-garde. Placez la disquette contenant FORTH.EXE dans le Drive B: et celle contenant les enregistrements FORTH dans le Drive A: Lancez FORTH en frappant au clavier A:b:forth <rc>. Tapez ensuite 118 LOAD <rc>.

Enfin, placez une disquette formatée dans le drive B:

DISK_COPY <rc> copiera la disquette contenant les enregistrements FORTH placée dans le drive A: sur votre disquette du drive B: Copiez la disquette l par des moyens classiques (Diskcopy sous Ms-Dos).

NB: FORTH n'est utilisable qu'en travaillant en MAJUSCULES.

Ce FORTH comprends également bien entendu un éditeur d'enregistrement. Celui-ci est compris dans les enregistrements 55 à 117. Pour le charger, tapez la ligne de commandes suivante :

55 117 THRU <rc>

Pour le lancer, tapez EDITOR <rc>

Vous serez alors en mesure d'éditer et de sauver des enregistrements FORTH : Tapez E ID..... <-- Entrez la date <rc>

Quand vous aurez fini d'utiliser FORTH, taper BYE vous ramènera sous système d'exploitation.

NOUVELLES BREVES

Les adeptes du FORTH sur Hector HEX ou équivalent seraient sûrement intéressés par "l'Hectorien" revue éditée par SORACOM, Le grand Logis, 10 Av du Gal de Gaulle 35170 Bruz tel 99 52 98 11. Je leur ai proposé un échange de revue, sans réponse à cette date.

La grande réunion du club CCD va se tenir à Franckfort le 2 Février 86, mais en allemand, ce qui réduit la participation des étrangers (elle était prévue initialement internationale et en anglais)

Je vous remercie de m'avoir envoyé un exemplaire du livre de Mr Horn, qui est parfois très intéressant.

Sachez qu'ici, à Paris, nous préparons des mots-clé de plus en plus performants. Nous allons sans doute bientôt créer un logiciel qui permette de gérer des ports indépendants de 8, 16, compliqué d'empêcher qu'ils fussent détruits lors de chaque reconfiguration du 71.

En attendant, j'ai profité des vacances de Noël pour faire quelques petites fonctions amusantes :

TR(A) Renvoie la trace de A, où A est une matrice pour l'instant non complexe.

EXECUTE L\$ Considère L\$ comme une ligne, la compile, la place dans un buifer puis l'exécute. On peut ainsi programmer

des ordres non programmables :

EDIT PRG ABC

AUTO

10 DISP 'OK'

20 EXECUTE "FREE PORT (0) à RUN PRG ABC,30"

30 DISP 'DONE'

RUN

ENTRY\$ TOKEN TYPE Renvoient respectivement le point d'entrée, le token (sous la forme requise pour MSG\$), ou le type d'un mot-clé.

FIND S\$ Recherche la chaîne S\$ dans le programme BASIC courant. Affiche ou bien un message d'erreur ou bien la première ligne qui la contienne :

EDIT EDTEXT

FIND "ND SUB"

donne après 5 à 6 secondes : 4000 END SUB

Les syntaxes de SECURE, UNSECURE, DISABLE et ENABLE ont été étendues pour que l'on puisse écrire :

SECURE ALL

UNSECURE :PORT

DISABLE :MAIN

ENABLE :PORT (0.01)

On a prévu d'y ajouter des fonctions comme HELP...

ARGUS

Enfin, CONFIGX appelle un programme BASIC après chaque MEMORY LOST, ce qui permet de reconfigurer facilement son HP-71

Jean-Jacques Moreau 64 Av de la Paix 93150 Le Blanc Mesnil

SERVEUR MINITEL - BOITE AUX LETTRES POUR PPC-T

Nous avons retenu deux boîtes à lettres sur un serveur nommé KOALA. La première est réservée au bureau de PPC-T. La deuxième est pour vous. Ces boîtes aux lettres sont essentiellement destinées à la communication entre les animateurs et les adhérents ou des adhérents entre eux. Ce service est gratuit pour le club. Mais il est payant pour l'utilisateur (nous y compris quand nous installons les messages). Pour que le prix (1F la minute) reste raisonnable, il faut, dès que le message voulu est affiché sur votre minitel que vous vous déconnectiez (deux pression sur la touche "connexion/fin", un F dans le coin en haut à droite de l'écran). Vous conservez le message sur votre écran, mais la ligne est libérée. Pour cette même raison de coût, nous ne pourrions pas lire vos messages (du moins tant que nous ne pourrions pas nous connecter avec un ordinateur), nous essayons d'inscrire un message à chaque réunion de bureau, environ tous les 15 jours, sans effacer l'ancien. Donnez-nous (par écrit) vos réactions.

La procédure est la suivante : allumez votre Minitel, décrochez le téléphone, composez 36 15 91 77, à la tonalité pressez la touche de connexion et raccrochez le combiné. L'indicatif de connexion est JB, puis 3 (Koala Telestrad), puis 3 boîte aux lettres. Le mot de passe est "000PPC-T", lisez ensuite les messages (l'option 2, relecture est la plus utile, vous serez rarement le premier lecteur). Si vous voulez "envoyer un message", il faut savoir que le nom du destinataire qui vous est demandé est le numéro de sa boîte aux lettres. Ici c'est "IK0009" (lettre I, lettre K, puis des chiffres). Ecrivez aussi pour me dire ce que vous souhaitez voir dans ces "lettres". La lettré est un simple message d'essai, enregistré alors que personne ne connaissait ce service.

ARGUS

Cet ARGUS est très indicatif, peu de transactions ayant été enregistrées il tient compte cependant des tendances de prix vues dans les petites annonces. Il vaut pour des appareils en parfait état de marche et d'aspect, dans l'emballage d'origine et avec tous les manuels. La première colonne contient les prix neufs au tarif Janvier 86 de Hewlett-Packard. La deuxième colonne s'entend pour des matériels ayant encore au moins deux mois de garantie à courir à la date du dépôt. Les prix tiennent compte d'une décote de 30% dès l'achat, augmentée de 15% par an, le prix après cette période dépendant surtout de l'état de l'appareil. Le HP-75 se vendant mal est coté très bas, et représente une très bonne affaire à ce prix.

Argus PPC-T. Valeurs indicatives pour les échanges de matériel d'occasion entre particuliers.

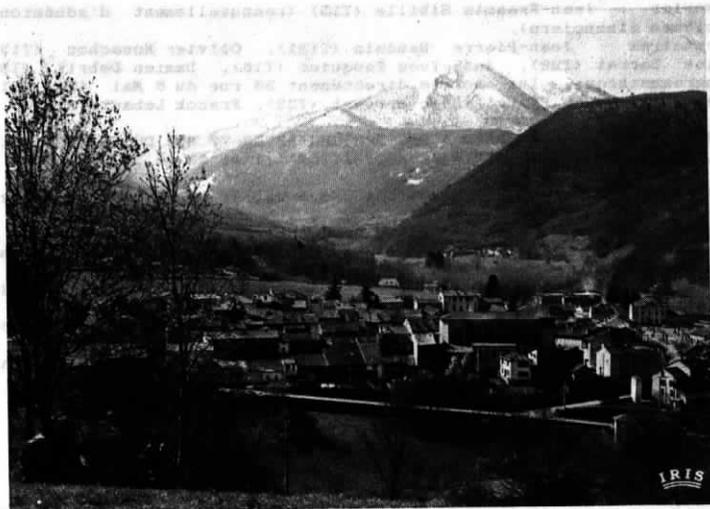
MATERIEL	ARGUS PPC-T			
	PI NEUF 100%	UN AN 70%	DEUX ANS 55%	TROIS ANS 40%
Taux				
HP-11	846	592	465	338
HP-15	1593	1115	876	637
HP-16	1593	1115	876	637
HP-12	1593	1115	876	637
HP-41	-	1500	1200	1000
HP-41CV	2834	1984	1559	1134
HP-41CX	4184	2929	2301	1674
HP-71BF	6691	4684	3680	2676
HP-75C	-	6000	5000	4000
HP-75D	15304	10713	8417	6122
Modules HP-41				
Auto Dup	469	328	258	188
IL-DEV	1005	704	553	402
MEM simple	-	0	0	0
QUAD	-	500	400	300
XF	1012	708	557	405
XN	1012	708	557	405
TIME	1012	708	557	405
IL-E/S 82183	1012	708	557	405
PLOTTER	1005	704	553	402
HP-IL	1687	1181	928	675
Périph HP-41				
Lect Cartes	2631	1842	1447	1052
Lect Cod. Barre	1686	1180	927	674
Imp 82143	5195	3636	2857	2078
Accessoires HP-71				
Lect Cartes	2200	1540	1210	880
RAM 4K	965	676	531	386
HP-IL	1622	1135	892	649
Trans Pack	1677	1174	922	671
FORTH/assembleu	2012	1408	1107	805
Math	1273	891	700	509
Analyse circui	1005	704	553	402
Finance	1005	704	553	402
Lissage	1273	891	700	509
Edit texte	1005	704	553	402

Accessoires HP-75				
Mod C. Barres 7	1005	704	553	402
MEM 8K	2615	1830	1438	1046
I/O ROM	1273	891	700	509
Text Formateur	1273	891	700	509
Math	1945	1362	1070	778
Visicalc	2615	1830	1438	1046
Pod 04K	11374	7962	6256	4550

Accessoires HP-1L Universels				
Thinkjet 2225	5869	4108	3228	2348
Disquette 9114	10424	7297	5733	4170
Lecteur 82161	7382	5167	4060	2953

Argus PPC-T. Valeurs indicatives pour les échanges de matériel d'occasion entre particuliers.

Imp 82162	6073	4251	3340	2429
Traceur 7470	14374	10062	7906	5750
IL/RS232	3954	2768	2175	1582
GPIO/IL	3981	2787	2190	1592
IL/HP-1B	5263	3684	2895	2105
Cassettes	127	89	70	51



Qu'est-ce que c'est ?

ORGANISATION DE PPC-T

Cette page sera publiée dans chaque numéro de MICRO-REVUE. Seules les indications répertoriées dans le dernier numéro du journal font foi. PPC-T étant une association sans but lucratif, elle ne fait pas de bénéfice. Elle est administrée par des bénévoles et ses produits et services sont réservés à ses membres. La marge prise sur les produits vendus est minimale et destinée à couvrir les frais de fonctionnement.

ASSOCIATION PPC-T

Association à buts non lucratifs (loi de 1901), PPC-T accepte des adhérents dans toute la France et à l'étranger. La cotisation est de 250F y compris l'abonnement à MICRO-REVUE. DOM-TOM et reste du monde par avion, ajouter 50F. Règlements par chèques et mandats postaux à l'ordre de PPC-T (Eurochèques acceptés, mandats postaux ou chèques sur une banque métropolitaine pour les DOM-TOM). L'inscription prends effet au premier jour du bimestre en cours lors de la réception au club du bulletin d'adhésion (1er Janvier, 1er Mars...). Les adhérents s'engagent à apporter leur contribution au journal et au club dans la mesure de leurs moyens, même modestement.

RESPONSABLES DE PPC-T

Président : Jean-Daniel Dodin (T1) (nouvelles adhésions, édition du journal, réception des articles)
Trésorier : Jean-François Sibille (T15) (renouvellement d'adhésion, problèmes financiers).
Secrétaires : Jean-Pierre Baudoin (T131), Olivier Monachon (T13), Gilles Barret (T22), Jean-Yves Pasquier (T18), Damien Debril (T33) (Programmathèque, lui écrire directement 38 rue du 8 Mai 45, 59190 Hazebrouk), Bruno Piguet, Eric Sergent (T33), Franck Lebastard.

ADRESSER TOUT LE COURRIER A PPC-T/MICRO-REVUE 77 RUE DU CAGIRE 31100 TOULOUSE. TEL (A UTILISER AVEC MODERATION) 61 44 03 06

COORDINATEURS REGIONAUX : N'hésitez pas à entrer en contact avec eux !

Bretagne : Olivier Pugeon 6 Bd Chanard, 56170 Quiberon.
Paris : Serge Vaudenay (T270), 62 av Ardouin, B3, 94420 Le Plessis Tréville ; Olivier Arbey (T164) 34 av de la République 94100 St Maur.
Picardie : Bruno Tredez (T120), 16 Chemin St Jean Allonville 80260 Villers Bocage.
Savoie : Georges-Noël Nicolas (T438), 5 rue de la Jonchère, 74600 Seynod.
Nice : Eric Angelini (T120), Le Vallon des fleurs, B9, 103 Av Henri Dunand, 06100 Nice.



RENSEIGNEMENTS FACULTATIFS

Sujets d'intérêt, domaine préférentiel d'utilisation de votre ordinateur de poche :

Je dispose des matériels informatiques suivants (type et modèle, date d'achat) :

J'ai eu connaissance de PPC-T par :

N°PPCUS : N°CHHU

Observations :

Signature : Date :

Votre attention sera prise en compte à réception par PPC-T du présent bulletin convenablement rempli. Remplir le verso SVP, merci.



Club d'utilisateurs d'ordinateurs de poche
Hewlett-Packard

Club d'utilisateurs du langage Forth
77 rue du Cagire 31100 Toulouse France

- Nouvel adhérent
- Renouvellement
- Changement d'adresse
- (Rayer la mention inutile)
- PPC-T N°.....

BULLETIN D'ADHESION

NOM :

PRENOM :

ADRESSE :

PAYS : Téléphone :

PROFESSION :

Ecrivez clairement en majuscules, merci.

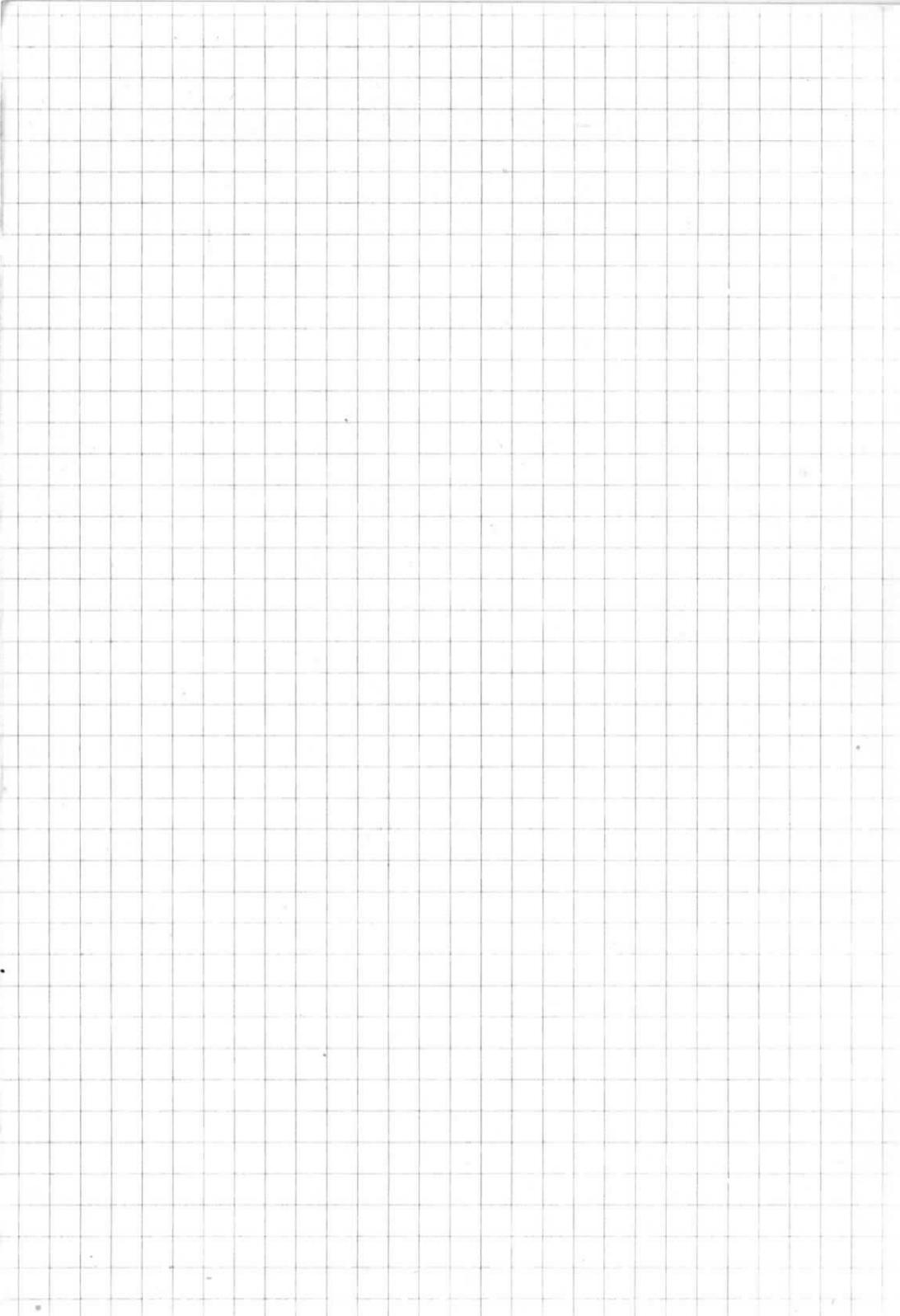
La fonction de PPC-T est de favoriser les connaissances sur les ordinateurs et leurs langages, grâce aux échanges entre les adhérents, dont les adresses sont régulièrement publiées.

Je demande à adhérer à PPC-T. Je joins mon règlement de 250F (300F par avion) par : Mandat Chèque CCP Espèces.

Date : _____ Signature : _____

Votre adhésion sera prise en compte à réception par PPC-T du présent bulletin convenablement rempli.

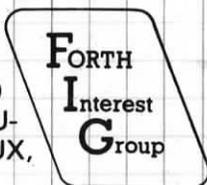
Remplir le verso SVP, merci



PPC

PROGRAMMATION SYNTHETIQUE, LCD, IMPRIMANTE, REGISTRES, MODULE, TOUCHES, OVER LAYS, MATRICES, MEMOIRE CONSTANTE, LECTEUR

DE CODES A BARRES, LECTEUR DE CARTES, BASIC, RPN, FORTH, LANGAGE SPECIALISE, MICROCODE, ASSEMBLEUR, MATHEMATIQUES, SOUS PROGRAMME, NOMBRE ALEATOIRE, DRAPEAUX, LUCASIEWICZ, PILE, ALPHANUMERIQUE, PARASITES, RI-DEAU, SWAP, NOTATION POLONAISE INVERSE, DUP, SHIFT, ECRAN, MOT, COMPILER, EXECUTION, TESTS, BOUCLES, DO LOOP, BEGIN UNTIL, IF ELSE THEN, : ;, BOUCLES CONDITIONNELLES, LABELS, TABLEAUX, TRACE, SST, BST, R/S



Edité par PPC-T - TITRE: MICRO-REVUE

77, rue du Cagire 31100 TOULOUSE

ISSN 0754-6335

Directeur de Publication : J.-D. Dodin

Imprimé par la Société Pyrénéenne d'Impression S.A.

20 Av. du Lauragais 31400 TOULOUSE