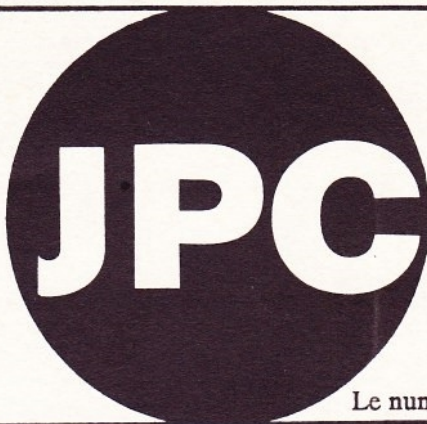




SEPTEMBRE 1989

NUMERO 67

Le numéro 40 F

## A PROPOS DU CLUB

Le Bureau

J. Belin

J. Belin

Editorial

1

Compte rendu de la conference de Chicago

2

Nouveaux produits

2

Courrier du coeur

3

## HP-28

A. Buchmann

J. P. Barre

P. J. Clemenceau

R. Le Bris

K. N. Dang-Trung

Guillaume le Stum

Y. Rousse

Y. Rousse

Nouvelles fonctions

6

Ecran virtuel

7

Résolution d'équations différentielles

12

Programme Orgue

12

Calculs Matriciels

14

Générateur d'histogrammes

17

Cryptage de chaînes de caractères

18

Factorielles (Acte II)

19

Fonction SWITCH (Acte II)

19

## HP-41

B. Wiklund

M. Cracknell

Récupération de registres d'assignations

22

Bataille navale

24

## EDITORIAL

Chers amis,

Tout d'abord, nous espérons que vous avez tous passé d'excellentes vacances.

L'appel que nous avons lancé dans le dernier numéro semble avoir été entendu, du moins en partie car si nous avons reçu un peu plus d'articles que ces derniers mois, vous pouvez noter que ce numéro est encore très fin. Ceci est dû à l'absence totale d'articles pour le HP-71. C'est la première fois depuis la sortie de cette machine et nous espérons que ce n'est qu'exceptionnel.

Pour vous inciter à participer un peu plus à la création du journal, nous avons décidé de récompenser les membres les plus actifs. Ceci en attribuant un certain nombre de points pour chaque article publié. Selon un certain nombre de critères, dont l'originalité et l'humour ne sont pas exclus, nous attribuerons de 1 à 5 points à chaque article. Au bout de 15 points l'heureux membre gagnera un mois d'adhésion supplémentaire. Il n'y a évidemment pas de limitation de temps. Bien entendu, les articles parus dans ce numéro sont pris en compte, et si certains n'y retrouvent pas leurs programmes, c'est que nous n'avons pas fini de les taper ! Vous devez aussi savoir que dans un élan de folie furieuse nous avons décidé que les membres du bureau ne bénéficieront pas de cette mesure (à propos combien de points donnez vous à cet éditorial ?).

Sinon, vous pourrez noter dans ce numéro le grand retour du micro-code pour HP-41. A propos pourquoi ne pas faire un utilitaire MAKELEX pour entrer plus facilement ce programme dans un module HEPAX ?

Vous pourrez aussi noter l'arrivée de nouveaux produits. Malheureusement, les photos que nous avons reçues n'étaient pas de qualité suffisante pour vous les montrer. Ce n'est que partie remise...

Enfin, nous recevons toujours vos questionnaires. Que ceux qui ont oublié de répondre se rassurent, il peuvent encore nous le renvoyer...

Nous vous laissons à votre lecture, et vous donnons rendez-vous le mois prochain.

Le bureau

## LA CONFERENCE DE CHICAGO

Comme vous le savez, une conférence s'est déroulée le 3 juin à Chicago pour fêter les dix ans de la HP-41 et les cinquante ans de Hewlett Packard.

Notre ami Michael Markov nous a envoyé les actes des conférences ainsi que divers documents. Le tout faisant plus de 150 pages, vous comprendrez facilement que ceci n'est qu'un très court résumé.

Les conférences et présentations se sont déroulées entre 9 heures et 18 heures au College Triton de Chicago. Le programme a été préparé par Richard Nelson.

Si notre constructeur préféré n'a pas annoncé de nouveau matériel (hormis le HP-21), d'autres fabricants bien connus de nous en ont profité pour présenter de nouveaux périphériques (voir l'article suivant).

On peut regrouper ces conférences en différents domaines : utilisation pour l'éducation, nouveaux concepts de machines ou variations sur les machines existantes, présentations de programmes ou de réalisation de périphériques...

Dans le premier domaine, Michael Bruggeman, d'EduCalc, a fait une présentation des différentes calculatrices (aussi bien HP que des autres marques) suivant le niveau de complexité de ces machines, et le niveau de ses utilisateurs potentiels ! Après cela, R. Lee Woodruff a raconté les réticences à l'entrée des calculatrices dans les universités américaines. Réticences du côté des enseignants, on le sait, mais aussi du côté des étudiants !

Après avoir montré l'évolution des calculateurs HP (du HP-35 au HP-28), Jake Schwartz a présenté ce qui pour lui serait deux améliorations de la HP-28S : une version avec un clavier unique, comme pour la HP-42, et surtout une version où le clavier gauche serait remplacé par un écran à cristaux liquides servant de clavier configurable à volonté. Souhaitons que les ingénieurs de chez HP ont écouté avec beaucoup d'attention cette conférence !

En ce qui concerne les présentations de programmes on peut citer un programme, développé par Tacit Logic Systems, tournant sur IBM PC, permettant d'émuler une HP-41 (sans toutefois accepter les fonctions XROM et la programmation synthétique) ainsi qu'échanger des données entre les deux machines par l'intermédiaire d'une liaison HPIL.

Seulement sur HP-41 maintenant, Paul Hubbert a écrit un programme permettant d'exécuter de façon fiable des programmes ou sous programmes directement en X-Memory.

Pour les programmeurs en micro-code, il existe maintenant un module, le DAVID-ASSEM, donnant, grâce à 5 fonctions, un environnement de programmation complet, comprenant entre autres un assembleur/désassembleur. Mais sans extension de mémoire comme sur les modules HEPAX.

D'autres programmes ont été présentés pour d'autres machines. Par exemple des programmes mathématiques sur HP-28, ou le Lex TSORTLEX, écrit par Michael Markov, permettant d'effectuer plus facilement des opérations sur les fichiers TEXT, comme le tri par exemple...

Du côté des réalisations électroniques, Gary Friedman a indiqué comment faire soit même un convertisseur HP82166 à moindre frais.

L'autre réalisation, signée Bruce Stevens, est un module infrarouge pour HP-71 (à placer à la place du lecteur de cartes). Pour 50 francs de composants (maximum !) et à l'aide du Lex approprié, il est maintenant possible de sortir les résultats sur l'imprimante de la HP-28 (ou pourquoi pas commander autre chose ?).

Jacques Belin (123)

---

## NOUVEAUX PRODUITS

Le principal défaut de la HP-28 est son manque d'entrées-sorties. La firme Dannoise WM Electronics (bien connue pour son module HEPAX) vient de remédier à ce problème en construisant la HP-28COM, c'est à dire une HP-28S modifiée et dotée d'une interface RS-232. Celle ci se présente sous la forme d'un petit boîtier ayant d'un côté une prise RS-232 9 broches, de l'autre, un câble à 3 broches, à connecter sur la HP-28 à la place de la diode infra-rouge (il est toutefois possible de brancher une diode à la place de l'interface). La machine possède de nouvelles fonctions : multiplication par deux de la vitesse, gestionnaire de mémoire, utilitaires de programmation en assembleur, et bien sûr toutes les fonctions nécessaires pour envoyer des programmes ou des données vers un compatible IBM. Des disquettes contenant les programmes nécessaires à l'IBM sont fournies.



Commercialisé par HP, mais conçu par Firmware Corporation, un système, similaire au DATA-SHOW de Kodak, permet de projeter sur un écran l'affichage d'une HP-28, grâce à la liaison infrarouge. Son prix est de 500 \$. Avis aux enseignants : sachez qu'il est possible de l'avoir gratuitement (ainsi que votre HP-28S personnelle), mais seulement si vous commandez 30 HP-28S avant le 31 Octobre 1989 !

Non présenté à Chicago, la société allemande W&W Software Produkts propose aussi un nouveau lecteur de disquettes HP-IL, dans le même boîtier que le 9114, mais avec DEUX lecteurs superposés ! Pour 1559 DM, vous pourrez enfin faire des copies de vos disquettes préférées !

Jacques Belin (123)

---

## COURRIER DU COEUR

Alain GILLET  
1, rue du marronnier  
BAINVILLIERS BROMEILLES  
45390 PUISEAUX  
TEL: 38 33 56 37 (APRES 18H)

Vend :

imprimante HP-82162A, HPIL, 14 rouleaux de papier : 1200 FF ; Lecteur de code barres, pour HP-41 : 400 FF ; Les deux 1400FF.

André VIALARON  
11, rue Darquié, 11  
31000 TOULOUSE  
Tel : 61 52 20 03  
ou 65 29 71 21

Vend :

Une HP-41C en excellent état, avec modules : QUAD-Mem, X-Fonctions, TIME : le tout 900 FF

---  
Jean REIBEL  
9 Square Victor Fleming  
92350 LE PLESSIS ROBINSON  
Tel : (1) 46 31 46 11

Vend :

HP-42S (sous garantie) 800 FF; Module Infrarouge + Imprimante thermique 82240A (compatible HP-28) : 1000 FF.

---  
Philippe Guez  
56 rue J.J. Rousseau  
75001 PARIS  
Tel : (1) 42 33 49 13

Vend :

HP-41 et matériel compatible HPIL (disquette HP9114, ThinkJet...)

---  
M. du CHAXEL  
11 rue Saint Saens  
92400 COURBEVOIE

Vend :

HP-97 pratiquement neuve, avec manuels en Français et Anglais. Prix à débattre.

---  
Joseph Hectus  
78 Av. d'Alfortville  
94600 Choisy-le-Roi  
Tel : 48 98 13 10  
ou 48 84 84 60

Vend :

HP-41CX, 2 modules X-Memory, module Advantage, module HPIL, notices. Excellent Etat.



## SOS

B. Wiklund  
123, rue M. Detournay  
HAMEL  
59151 ARLEUX  
Tel : 27 89 31 93

Cherche :

VASM de la HP-41CX et des modules d'application,  
ainsi que drive HP9114.

---

L. Weil  
Service SELT  
Alcatel Thomson Faisceaux Hertiens (ATTH)  
55 rue Greffulhe  
92301 Levallois Perret

Cherche :

Manuels du plotter HP7470A et du module 82184 (en  
prêt pour copie).



## HP-28

|                   |                                        |    |
|-------------------|----------------------------------------|----|
| A. Buchmann       | Nouvelles fonctions                    | 6  |
| J. P. Barre       | Ecran virtuel                          | 7  |
| P. J. Clemenceau  | Résolution d'équations différentielles | 12 |
| R. Le Bris        | Programme Orgue                        | 12 |
|                   | Calculs Matriciels                     | 14 |
| K. N. Dang-Trung  | Générateur d'histogrammes              | 17 |
| Guillaume le Stum | Cryptage de chaînes de caractères      | 18 |
| Y. Rousse         | Factorielles (Acte II)                 | 19 |
| Y. Rousse         | Fonction SWITCH (Acte II)              | 19 |

## NOUVELLES FONCTIONS

Vous avez tous encore, dans un coin de mémoire de votre HP-28C, le fameux programme ASS (*JPC 50* p. 8). Dans ce cas, voici quatre nouvelles fonctions à rajouter au menu. Elles se nomment : APL→, →PRG, →ALG, PSIZE.

### Description :

APL→ : Expression, programme ou liste vers pile.

Cette fonction prend une expression, une liste ou un programme de  $n$  objets au niveau 1 de la pile et renvoie les objets constituant l'expression, la liste ou le programme dans les niveaux 2 à  $n+1$  de la pile. Le nombre  $n$  est renvoyé au niveau 1.

→PRG : Pile vers programme.

Cette fonction prend un nombre entier  $n$  dans le niveau 1, plus  $n$  objets supplémentaires dans les niveaux 2 à  $n+1$ , et renvoie dans la pile un programme constitué de  $n$  objets.

→ALG : Pile vers expression.

Cette fonction prend un nombre entier  $n$  dans le niveau 1, plus  $n$  objets supplémentaires dans les niveaux 2 à  $n+1$ , et renvoie dans la pile une expression constituée des  $n$  objets.

### Attention :

L'expression renvoyée au niveau 1 doit obéir aux règles de la syntaxe algébrique définie dans le manuel de référence page 61. En particulier, lorsqu'une expression est évaluée, elle ne prend aucun argument dans la pile, et elle a pour résultat d'y envoyer un argument. Cette expression peut être complètement subdivisée en une hiérarchie de sous-expressions. Il n'y a aucun contrôle, d'où risque de plantage. Pour des expressions compliquées dont on n'est pas sûr de la syntaxe, il est souvent possible d'utiliser →PRG, puis d'évaluer l'expression.

PSIZE : Taille du programme.

Cette fonction prend un programme de  $n$  objets dans le niveau 1 et lui substitue le nombre  $n$ . Ce nombre n'a rien à voir avec la taille en octets du programme.

### Utilisation

- Transformation d'une expression algébrique en programme. Seul l'en-tête est modifié, la séquence d'objets reste la même.

- Programmation synthétique.

- Suppression des "«".

- Utilisation de routines écrites en Rom et non accessibles directement.

- Fusion de plusieurs programmes, etc...

### Listes

Ces codes hexadécimaux correspondent à la HP-28C version 1BB, mais en vous aidant des commentaires, il vous sera aisé de les adapter pour votre machine.

APL→ :

76C20 début de structure  
5B3C1 copié dans LIST→  
307C1 copié dans LIST→  
0BD60 short integer #9  
92B71 algebraic→  
6AD60 short integer #8  
92B71 program→  
88D60 short integer #5  
92B71 list→  
09F20 fin de structure

→PRG :

76C20 début de structure  
881C1 copié dans →LIST  
43350 copié dans →LIST  
54830 analogue à 95830, mais avec short integer  
égal à #26C7 au lieu de #2A96  
C53C1 copié dans →LIST  
09F20 fin de structure

→ALG :

76C20 début de structure  
881C1 copié dans →LIST  
43350 copié dans →LIST  
D6830 analogue à 95830, mais avec short integer  
égal à #2ADA au lieu de #2A96  
C53C1 copié dans →LIST  
09F20 fin de structure



-PRG :

76C20 début de structure  
5B3C1 copié dans SIZE  
307C1 copié dans SIZE  
6AD60 short integer #8  
5DB71 copié dans SIZE  
09F20 fin de structure

Alexandre Buchmann (386)

## ECRAN VIRTUEL

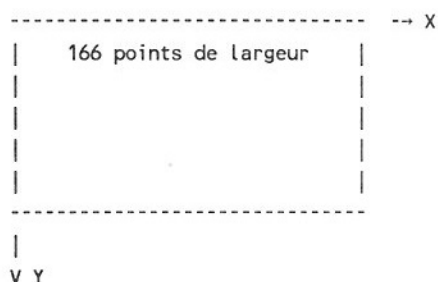
Le programme suivant simule un écran virtuel graphique sur la HP-28S (ou 28C avec une extension mémoire et les fonctions le transfert entre mémoire écran et chaîne).

Plus exactement ce programme est une TOOLBOX qui permet de travailler sur un écran graphique de 166 points de largeur par  $8 \cdot n$  points de hauteur. Il faut remarquer que pour des raisons de temps d'exécution, le tracé ne s'effectue pas en temps réel. (il est cependant possible d'afficher l'écran virtuel à tout instant). La finalité du programme est de pouvoir sortir des courbes sur une largeur de 166 points avec l'imprimante HP82240, et la résolution de celle-ci.

Dans la mesure où les traitements sont assez longs, aucune vérification sur la validité des paramètres passés à la toolbox ne sont effectués. C'est à l'utilisateur de le faire quand cela est nécessaire.

Comme application de cette toolbox, un programme de tracé de fonctions en coordonnées cartésiennes ainsi que paramétriques se trouve plus loin.

Avant de décrire toutes les fonctions, voyons comment se présente cet écran virtuel.



Y est dans le sens du défilement du papier de l'imprimante. On peut remarquer que la disposition des axes X et Y n'est pas habituel. Cela vient de la possibilité d'utiliser les fonctions de tracé de la HP-28 (pour X) et qu'il est naturel que le papier se déroule selon +Y.

Passons aux explications des fonctions de cette toolbox :

- CRSCR (n) : (entrée : un entier sur la pile)

Crée un écran virtuel comportant le nombre de lignes que vous désirez. Pour cela vous devez mettre sur le premier niveau de pile le nombre de points que vous désirez selon Y. Ce nombre doit être un multiple de 8 (il sera de toute façon arrondi au multiple de 8 par défaut). La possibilité de choisir au pixel près aurait entraîné des calculs encore plus longs...

Cet écran est en permanence stocké dans la variable °SCR ce qui permet, par exemple, de travailler sur plusieurs écrans.

L'encombrement des écrans se calcule de la façon suivante :

$$L = IP(n/8) * 166 \text{ octets}$$

- DELSCR :

Détruit l'écran virtuel créé par CRSCR.

- PSET (x,y) : (entrée : deux entiers sur la pile)

Allume un point de l'écran virtuel aux coordonnées (x,y). Les coordonnées doivent être fournies sur les deux premiers niveaux de pile.

**Attention :** x doit être compris entre 1 et 166 et y entre 1 et n.

Il faut absolument respecter cet intervalle sous peine de perturber sérieusement la toolbox...

- LINE (x1,y1,x1,y2) : (entrée : quatre entiers sur la pile)

Trace une ligne entre les points de coordonnées (x1,y1) et (x2,y2).

- PRESET (x,y) : (entrée : deux entiers sur la pile)

Eteint un point de l'écran virtuel aux coordonnées (x,y). Même remarque que pour PSET.

- GETPIX (x,y) : (entrée : deux entiers sur la pile)

Donne l'état d'un point de l'écran. La fonction retourne un entier qui, s'il vaut 0, indique que le point est éteint, sinon qu'il est allumé. Mêmes remarques que pour PSET.

- DISPEN (x,y) : (entrée : deux entiers sur la pile)

Affiche à l'écran de la HP-28 une partie de l'écran virtuel (fenêtre). Le point de coordonnées (x,y) représente le point qui se trouve en haut à gauche de l'écran de la HP-28. Dans la mesure où l'écran virtuel fait 166 points de largeur et que celui de la HP-28 en fait 137, x ne peut prendre n'importe quelle valeur.

En ce qui concerne y, celui-ci doit être un multiple de 8 majoré de 1 pour les mêmes raisons que pour CRSCR.

**Attention** : x doit être compris entre 1 et 29, et y entre 1 et n.

Si y n'est pas un multiple de 8 majoré de 1, y y est ramené par valeur inférieure.

- LCD→SCR (x,y) : (entrée : deux entiers sur la pile)

Transfère le contenu de l'écran de la HP-28 sur l'écran virtuel. Il faut noter que le contenu antérieur de l'écran virtuel est perdu.

Le point de coordonnées (x,y) correspond aux coordonnées sur l'écran virtuel où doit venir se transférer l'écran de la HP-28.

(x,y) est associé au point en haut à gauche de l'écran de la HP-28. Mêmes remarques que pour DISPEN.

- VISU :

Affiche à l'écran de la HP-28 une fenêtre se déplaçant sur l'écran virtuel à l'aide des touches fléchées. Le pas est de 7 points selon x et de 8 points selon y.

- PRTFEN (x,y) : (entrée : deux entiers sur la pile)

Imprime le contenu sur l'imprimante HP82240 d'une fenêtre se trouvant sur l'écran virtuel dont les coordonnées du point en haut à gauche sont (x,y). Mêmes remarques que pour DISPEN.

- PRALL :

Imprime le contenu complet de l'écran virtuel sur l'imprimante HP82240. On peut remarquer que les proportions ne sont pas exactement respectées (valable aussi pour PRTFEN). Cela vient du fait que toute la largeur du papier est utilisée pour

l'impression, soit 166 points ce qui permet des représentations graphiques intéressantes.

Avant de décrire l'utilisation du traceur de courbes donné comme exemple d'application de la toolbox, quelques remarques sur l'utilisation de celle-ci :

- Pour afficher des caractères sur l'écran virtuel il suffit de les écrire sur l'écran de la HP-28, puis de les transférer à l'aide de LCD→SCR sur l'écran virtuel. Un exemple de programme réalisant cette fonction pourrait être :

```
«
  1 1 DISPEN LCD→ "Bonjour" 2 DISP
  LCD→ OR →LCD 1 1 LCD→SCR
»
```

- Dans la mesure où l'écran virtuel est stocké dans une chaîne de caractère nommée \*SCR, il est extrêmement simple d'effectuer des opérations logiques sur les écrans à l'aide des fonctions AND, OR, XOR...

Cela permet par exemple de superposer plusieurs écrans virtuels.

- N'oubliez pas d'armer le flag 52 pour que l'impression ne soit pas trop longue.

- On peut évidemment utiliser cette toolbox sur une HP-28C avec extension mémoire et après avoir entré LCD→ et →LCD (voir les articles de Paul Courbis et Sébastien Lalandre à ce sujet) puisque ceux sont les seules fonctions spécifiques à la HP-28S qui sont utilisées dans la toolbox.

- Il faudrait écrire une telle toolbox (ou même une beaucoup plus complète) en assembleur, surtout la procédure PSET qui fait ralentir les programmes de tracé.

Un exemple d'utilisation de la toolbox : un petit traceur de courbe (un de plus !) en cartésiennes et paramétriques pour l'écran virtuel.

Quelques procédures décrites s'utilisent comme celles intégrées dans la HP-28. C'est le cas de VMIN, VMAX, VCEN. Les paramètres pour le tracé se trouvent dans la variable VSET sous forme d'une liste de la forme :

{ (xmin,ymin) (xmax,ymax) (xcenre,ycentre) N }

(où N représente le nombre de points selon y)

## Description des procédures :

- PREP (n) : (entrée : un entier sur la pile)

Crée un écran virtuel de  $n$  points de hauteur et initialise la variable VSET avec les mêmes valeurs que celles d'initialisation de Pmin et Pmax de la HP-28.

- VMAX (xmax,ymax) : (entrée : un complexe sur la pile)

(xmax,ymax) est un complexe indiquant les coordonnées du point en bas à gauche de l'écran virtuel.

- VMIN (xmin,ymin) : (entrée : un complexe sur la pile)

(xmin,ymin) est un complexe indiquant les coordonnées du point en haut à droite de l'écran virtuel.

- VCENT (xc,yc) : (entrée : un complexe sur la pile)

(xc,yc) est un complexe indiquant que le point de coordonnées (xc,yc) doit être le centre de l'écran virtuel (même fonction que CENTR de la HP-28).

- VSP : sous programme de VPAR et VCAR.

- VPAR (Tmin,Tmax,Nbr) : (entrée : trois réels sur la pile)

Lance la procédure de tracé sur l'écran virtuel de fonctions définies de manière paramétrique. Pour cela, il faut stocker dans les variables EQX et EQY les équations  $x(t)$  et  $y(t)$  de la représentation paramétrique. Dans EQX et EQY, la variable est T. Tmin et Tmax correspondent aux bornes de l'intervalle que doit décrire la variable T. Nbr indique en combien de subdivisions doit être découpé l'intervalle [Tmin,Tmax].

Il faut bien évidemment avoir lancé PREP avant d'exécuter VPAR qui utilise les valeurs contenues dans VSET pour le calcul de l'échelle et du centre de l'écran virtuel. Il faut remarquer que VPAR ne représente pas le tracé à l'écran de la HP-28 et que les conventions sur les sens et positions des axes X et Y sont celles de la toolbox. C'est à l'utilisateur, une fois que les calculs ont été effectués, d'utiliser soit DISPEN(x,y), PRTFEN(x,y), PRTALL, VISU pour la visualisation ou l'impression.

- VCAR (Xmin,Xmax,Nbr) : (entrée : trois réels sur la pile)

Lance la procédure de tracé sur l'écran virtuel de fonctions définies de manière cartésienne. Pour cela, il faut stocker dans la variable EQ l'équation  $f(x)$ . Dans EQ, la variable est X. Xmin et Xmax correspondent aux bornes de l'intervalle que doit décrire la variable X. Nbr indique en combien de subdivisions doit être découpé l'intervalle [Xmin,Xmax]. Mêmes remarques que pour VPAR.

Après ces explications sur la toolbox et les exemples d'utilisation de celle-ci pour le tracé de fonction, voici les programmes :

## - TOOLBOX :

### CRSCR

```
«
  8 / IP 0 CHR → N °0
  « °0 2 166
    START
      °0 +
    NEXT
    DUP '°0' STO 2 N
    START
      °0 +
    NEXT
    '°SCR' STO
  »
»
```

### DELSCR

```
«
  '°SCR' PURGE
»
```

### PSET

```
«
  → X Y
  «
    Y 8.0001 / IP 166 * X + DUP DUP
    °SCR SWAP DUP SUB NUM R→B 2 Y 1
    - 8 MOD ^ R→B OR B→R CHR °SCR
    1 4 ROLL 1 - SUB SWAP + °SCR 3
    ROLL 1 + °SCR SIZE SUB + '°SCR' STO
  »
»
```

### LINE

```
«
  0 RCLF 0 FIX → X1 Y1 X2 Y2 X FL
  «
    IF X1 X2 ≠ THEN
      Y2 Y1 - X2 X1 - / 'X'
      X1 - * Y1 + → EQ
    »
    X1 X2
    FOR X
      EQ →NUM RND X SWAP PSET
    »
  »
```

```

NEXT
»
ELSE
Y1 Y2
FOR Y
X1 Y PSET
NEXT
END
FL STOF
»
»

PRSET
«
→ X Y
«
Y 8.0001 / IP 166 * X + DUP DUP
°SCR SWAP DUP SUB NUM R-B 2 Y 1 -
8 MOD ^ R-B #255d SWAP - AND B-R
CHR °SCR 1 4 ROLL 1 - SUB SWAP +
°SCR 3 ROLL 1 + °SCR SIZE SUB +
'°SCR' STO
»
»

GETPIX
«
→ X Y
«
Y 8.0001 / IP 166 * X + °SCR
SWAP DUP SUB NUM R-B 2 Y 1 -
8 MOD ^ R-B AND B-R
»
»

DISPFEN
«
→ X Y
«
"" Y DUP 31 +
FOR YY
°SCR YY 8 / IP DUP °SCR SIZE
166 / IP
IF < THEN
166 * X + DUP 136 + SUB +
ELSE
DROP DROP
END
8
STEP
DUP →LCD SIZE 548 -
137 / -1
FOR A
"" A 5 + DISP
NEXT
»
»

```

```

LCD→SCR
«
LCD→ → X Y S
«
1 4
FOR YY
S YY 1 - 137 * X + DUP 136 +
SUB °SCR 1 YY 1 - 8 * Y + 8.0001
/ IP 166 * X + 1 - DUP 5 ROLL
SUB SWAP + °SCR SIZE LAST SWAP 4 ROLL
138 + SWAP SUB + '°SCR' STO
NEXT
»
»

VISU
«
°SCR SIZE 166 / 8 * 23 - 1 1 →
N X Y
«
CLLCD X Y DISPFEN
DO
DO
UNTIL KEY
END
DUP
IF "UP" SAME THEN
DROP
IF Y 8 > THEN
Y 8 - 'Y' STO
END
ELSE
DUP
IF "DOWN" SAME THEN
DROP
IF Y N < THEN
Y 8 + 'Y' STO
END
ELSE
DUP
IF "LEFT" SAME THEN
DROP
IF X 7 > THEN
X 7 - 'X' STO
END
ELSE
DUP
IF "RIGHT" SAME THEN
DROP
IF X 29 < THEN
X 7 + 'X' STO
END
END
END
END
END
X Y DISPFEN
UNTIL 0

```

```

END
»
»
PRTFEN
«
→ X Y
«
LCD→ X Y DISPFEN PRLCD
→LCD
»
»
PRTALL
«
27 CHR 166 CHR + → P
«
1 °SCR SIZE
FOR A
°SCR A DUP 165 + SUB P SWAP +
PR1 DROP 166
STEP
»
»
-TRACEUR :
PREP
«
DUP CRSCR
(-6.8,-1.5)
(6.8,1.6)
(0,0)
4 ROLL 8 / IP 8 * 4 →LIST 'VSET' STO
»
VMAX
«
VSET 2 3 ROLL PUT 'VSET' STO
»
VMIN
«
VSET 1 3 ROLL PUT 'VSET' STO
»
VCEN
«
VSET 3 3 ROLL PUT 'VSET' STO
»
VSP
«
→NUM SWAP →NUM 3 ROLL
→NUM → N A2 A1
«
A1 A2 VSET LIST→ DROP DUP DUP
4 ROLL C→R 7 ROLL C→R 8 ROLL C→R

```

```

3 ROLL - 6 ROLL SWAP / SWAP 3 ROLL
- 165 SWAP / DUP 5 ROLL * 83.5
SWAP - 3 ROLL DUP 5 ROLL * 5 ROLL
1 + 2 / SWAP - A2 A1 - N / RCLF
0 FIX
»
»
VPAR
«
VSP → NP X NX Y NY S FL
«
FOR P
P 'T' STO EQX →NUM X * NX +
RND DUP DUP
IF 0 > SWAP 167 < AND THEN
EQY →NUM Y * NY + RND DUP DUP
IF 0 > SWAP NP <= AND THEN
PSET
ELSE
DROP DROP
END
ELSE
DROP
END
S STEP
'T' PURGE FL STOF
»
»
VCAR
«
VSP → NP EX NX EY NY S FL
«
FOR P
P DUP 'X' STO EX * NX +
RND DUP DUP
IF 0 > SWAP 167 < AND THEN
EQ →NUM EY * NY + RND DUP DUP
IF 0 > SWAP NP <= AND THEN
PSET
ELSE
DROP DROP
END
ELSE
DROP
END S
STEP
'X' PURGE FL STOF
»
»

```

Jean-Paul Barre (522)



# RESOLUTION D'EQUATIONS DIFFERENTIELLES D'ORDRE 1

## METHODE DE RUNGE-KUTTA

Le but de ce programme est simple : Résoudre numériquement les équations différentielle d'ordre 1 par la methode de RUNGE-KUTTA. Je n'expliciterais pas ici la methode, le programme sous sa forme donnée (pas la plus compacte possible), étant, je l'espère, assez clair. Je me contenterais donc de donner le listing et le mode d'emploi; mais si quelqu'un connaît la démonstration de la méthode de RUNGE-KUTTA, j'aimerais bien qu'il me la fournisse.

Les équations résolues sont de la forme  $Y' = F(X, Y)$ , et le programme trouve la valeur numérique en  $x=c$  ( $c$  donné) de la solution  $y=g(x)$  qui passe par le point de coordonnées  $(x_0, y_0)$  (solution du problème de Cauchy)

Ce programme est très simple d'emploi. Il vous suffit de définir les variables suivantes:

- dans 'X0' mettre la valeur de  $X_0$  (si, si!!)
- dans 'Y0' celle de  $Y_0 = G(X_0)$  (remarque: ceci définit ainsi la constante de la solution).
- dans 'C' l'abscisse du point où l'on cherche  $y=g(c)$ .
- dans 'N' le nombre de pas, ce qui définit la précision du résultat: dans le cas général, et c'est là que la méthode de RUNGE-KUTTA est très performante, 5 pas suffisent largement, alors que la méthode de Newton par exemple en nécessite en général 500 (Rappel: la méthode de RUNGE-KUTTA est d'ordre 4).
- dans 'FXY' la fonction (au minimum de classe  $C^1$  sinon les résultats peuvent être altérés dû à la non unicité de la solution) qui correspond à  $Y' = F(x, y)$ .
- une fois ceci fait taper [RUNGE].
- Le programme ayant fini de tourner, la solution est dans Y0.

Le programme pourrait être grandement amélioré sur le plan de la convivialité avec plus de mémoire. Malheureusement je n'ai qu'une HP-28C.

```
« C X0 - 2 / N / 'P2' STO
1 N
START
X0 Y0 FXY DUP 6 INV * SWAP R2
P2 'X0' STO+ R1 R2 R1 2 * R2
P2 'X0' STO+ X0 SWAP FXY 6
INV * + + + 2 * P2 * 'Y0' STO+
NEXT
Y0
```

```
»
[RUNGE]
«
X0 SWAP FXY DUP 3 INV * SWAP
»
[R1]
«
P2 * Y0 +
»
[R2]
```

Exemple: calcul de la valeur en 1 de la fonction vérifiant  $Y' = Y$  et  $f(0) = 1$

Rentrer dans X0 la valeur 0 (0'X0 [STO]), 1 dans Y0, 1 dans C, 5 dans N, et définir FXY par « X Y 'Y' » ' ' FXY [STO] ( $FXY(X, Y) = Y$ ). Taper alors RUNGE. Après quelques temps de réflexion la machine vous répond: 2.7182511366 soit une erreur de l'ordre de l'ordre de  $10^{-5}$  par rapport à la valeur exacte (qui est  $e$ ).

Pierre-Jérôme Clémenceau (453)

## ORGUE-ANISEZ VOTRE HP-28

Ce programme permet de faire entendre les notes de la gamme par action d'un groupe de touches.

Il prend dans la pile deux nombres réels correspondant respectivement à l'octave et au temps d'émission souhaités, puis affiche la composition du clavier musical.

Les touches sont choisies pour représenter les notes de la gamme diatonique dans le mode anglo-saxon.

Les touches : C D E F G A B  
désignent les tons : DO RE MI FA SOL LA SI

Les touches : J K M N H  
désignent les demi tons : DO# RE# FA# SOL# LA#

Un chiffre indique l'octave en cours. Entre deux émissions on peut changer l'octave en actionnant les touches "+" ou "-".

On sort du programme par la touche [S] (ou toute autre touche que [ATTN] qui laisserait subsister dans le menu des variables globales d'un intérêt temporaire).

Le programme orgue utilise trois listes (à sauvegarder au préalable par STO) :

- GAMME

Liste des notes (tons et demi-tons).

- CLAVIER

Liste des touches composant le clavier musical.

- LNF

Liste des notes et fréquences.

Dans cette liste les notes sont immédiatement suivies de la fréquence de base (octave 1) qui les caractérise. Certains éléments de cette liste ne sont pas utilisés dans le programme ORGUE. Cela tient à ce que j'utilise LNF dans d'autres programmes de mon répertoire. Ne tenez donc aucun compte de son caractère exhaustif.

## Fonctionnement

Le programme est composé essentiellement de deux boucles DO...UNTIL...END imbriquées. La boucle interne exerce une surveillance des clés par l'instruction KEY. Lorsque une touche est actionnée, le programme passe dans la boucle externe et traite l'un des trois cas suivants :

1) La touche n'appartient pas au clavier musical.

Le programme cherche la position de la touche dans la liste CLAVIER. Ne la trouvant pas, un zéro apparaît dans la pile, ce qui met fin au programme.

2) Il s'agit d'une des touches [+1] ou [-1].

Suivant le cas, l'octave est incrémentée ou décrémentée. Le choix d'une nouvelle octave implique de calculer l'argument "fréquence" de la fonction BEEP. Ce calcul s'effectue à partir de la fréquence de base de la note concernée. (cf liste LNF). Il consiste à multiplier la fréquence de base par  $2^{\text{octave}-1}$ . (En RPN dans le programme).

3) La touche appartient au clavier musical.

Le programme cherche la position de la touche dans la liste CLAVIER. Puis, l'ayant trouvée, il extrait de la liste GAMME l'élément de même position. Enfin il cherche la position de cet élément dans la liste LNF, et prélève la fréquence qui suit. La note est alors émise par l'instruction BEEP.

## Listing du programme

CLAVIER

```
{ "C" "J" "D" "K" "E" "F" "M" "G" "N" "A"
  "H" "B" "+" "-" }
```

GAMME

```
{ "C" "C#" "D" "D#" "E" "F" "F#" "G" "G#" "A"
  "A#" "B" }
```

LNF

```
{ "C" 130.81 "B#" 130.81 "C#" 138.59 "D'" 138.59
  "D" 146.83 "D#" 155.56 "E'" 155.56 "E" 164.81 "F'"
  164.81 "F" 176.61 "E#" 176.61 "F#" 185 "G'" 185
  "G" 196 "G#" 207.65 "A'" 207.65 "A" 220 "A#"
  233.08 "B'" 233.08 "B" 246.94 "C'" 116.54 "S" 1
  "T" 50 }
```

ORGUE

```
«
  1 FIX CLLCD
  " → OCTAVE (0 à 4)"
  1 DISP
  " → DUREE (secondes)"
  2 DISP
  " puis CONT"
  3 DISP HALT → duree
  « 'OCTAVE' STO
  DO CLAVIER
    CLLCD "ORGUE (SORTIE PAR S)" 1 DISP
    "C D E F G A B (CLAVIER)" 2 DISP
    " J K M N H" 3 DISP
    "OCTAVE" "OCTAVE" →STR +
    " ('+' ou '-')" + 4 DISP
    DO KEY
    UNTIL 1 SAME
    END DUP DUP
    IF "+" SAME THEN
      'OCTAVE' 1 STO+ DROP DROP DROP
    ELSE
      IF "-" SAME THEN
        'OCTAVE' 1 STO- DROP DROP
      ELSE
        POS DUP 'SORTIE' STO GAMME SWAP
        DUP "SUB LIST" → DROP LNF SWAP
        POS 1 + LNF SWAP DUP SUB LIST →
        DROP 2 OCTAVE 1 - ^ * duree BEEP
      END
    END
    UNTIL SORTIE 0 ==
  END
  { OCTAVE SORTIE } PURGE
  2 FIX KILL
  »
  »
```

Roger Le Bris (535)

## CALCULS MATRICIELS

SWAP DROP

»

NDLR : Ceci est une compilation d'articles traitant du même sujet, mais provenant d'auteurs différents. Il n'y a donc à priori aucun lien direct (syntaxes, structures de données...) entre eux. C'est donc à vous de les adapter pour en faire un seul programme, si vous le désirez...

Guillaume Le Stum (432)

### Comparaison de matrices

#### Polynôme caractéristique d'une matrice

Ce programme donne le polynôme caractéristique d'une matrice carrée. Si P est le polynôme  $a_0 + a_1x + \dots + a_nx^n$  il est donné sous la forme d'un tableau [a0 a1...an]. si on a  $A(1)=A$ ,  $a(p) = 1/p \times \text{trace}(A(p))$  et  $A(p+1) = (A(p) - a(p) \times Id) \times A$  alors le polynome caractéristique de A est :  $A^n - a_1A^{n-1} - \dots - a_n$   
(Source : Denis Monasse "Un algorithme de calcul du polynôme caractéristique").

Entrée : ARRAY { n n }

Sortie : ARRAY { n+1 }

CARAC

«

DUP 0 CON

DIP SIZE LIST→ DROP2 → A B N

«

1 1 N

FOR K

DUP N IDN \* B + A \* 'B' STO B TRACE NEG K /

NEXT

N 1 + 2

FOR F

F ROLL

-1 STEP

N 1 + →ARRY

»

»

CARAC utilise TRACE qui, comme son nom l'indique, calcule la trace d'une matrice carrée.

TRACE

«

DUP SIZE LIST→ DROP2

0 SWAP 1 SWAP

FOR F

OVER F F 2 →LIST GET +

NEXT

Le calculateur va comparer un élément défini par L\*C (Ligne \* Colonne) de la première matrice avec l'élément correspondant dans la deuxième matrice. Le résultat est présenté sous la forme d'une matrice de même dimension dont les éléments sont des 1 ou des 0. 1 correspondant à un résultat positif du test.

Utilisation :

1) Entrez dans la pile le test à exécuter dans une chaîne de caractères : "==" "<" ">=" etc..

2) Entrez les deux matrices qui doivent absolument être de même dimension M1 et M2 tel que M1 teste M2.

3) [TEST]

Exemple "==" [[1 2] [3 4]] [[1 0] [0 4]] TEST

----- [[1 0] [0 1]]

TEST

«

DUP SIZE → t a1 a2 s

«

s LIST→ DROP → l c

«

1 l

FOR L

1 c

FOR C

a1 L C 2 →LIST DUP 3

ROLL SWAP GET a2 3 ROLL GET

t STR→

NEXT

NEXT

»

»

→ARRY

»

Khôi-Nguyên DANG-TRUNG (538)

## Manipulation de matrices symboliques

MATR est un ensemble de programmes qui permet de manipuler les matrices symboliques. Malheureusement, les routines étant en RPL, les temps d'action des routines sont assez longs. Toute amélioration sera donc la bienvenue.

Les matrices sont entrées sous forme de textes :

Exemple :

```
| A B |  
| C D |  => {{A B}{C D}}
```

Utilisation des programmes :

**MDET** : Déterminant

Entrée : niveau 1: Liste

Sortie : niveau 1 : forme algébrique ou nombre

**MADD** : Addition

Entrée : niveau 1: Liste1 - niveau 2 : Liste2

Sortie : niveau 1 : liste = Liste1 + Liste2

**MSUB** : Soustraction

Entrée : niveau 1: Liste1 - niveau 2 : Liste2

Sortie : niveau 1 : Liste = Liste2 - Liste1

**MSCL** : Multiplication par 1 scalaire

Entrée : niveau 1: Nb - niveau 2 : Liste

Sortie : niveau 1 : Liste = Nb \* Liste

**MMUL** : Multiplication

Entrée : niveau 1: Liste - niveau 2 : Liste

Sortie : niveau 1 : Liste = Liste2 \* Liste1

**MINV** : Inversion

Entrée : niveau 1: Liste1

Sortie : niveau 1 : Liste = inv(Liste1)

**MTRN** : Transposé

Entrée : niveau 1: Liste1

Sortie : niveau 1 : Liste = trn(Liste1)

**MJCB** : Jacobien

Entrée : niveau 1: Liste1 (contenant la liste algébrique des fonctions dont on doit faire le Jacobien.

Niveau 2 : Liste des variables utilisées.

Sortie : valeur algébrique ou nombre

Liste des programmes :

Menu : MATR

**MDET**

```
<<  
→ M  
<<  
RTN M →LST  
IF == THEN  
  DETN SMP →FR UP  
ELSE  
  DROP M 1 ERR  
END  
>>  
>>
```

**MADD**

```
<<  
→ M1 M2  
<<  
RTN M1 M2  
<<  
+  
>>  
ADSUB UP  
>>  
>>
```

**MSUB**

```
<<  
→ M1 M2  
<<  
RTN M1 M2  
<<  
+  
>>  
ADSUB UP  
>>  
>>
```

**MSCL**

```
<<  
RTN →LST → S M L C  
<<  
M DUP 1 L C *  
FOR I  
  I GET S * SMP →FR I SWAP PUT DUP  
NEXT  
DROP L C LST→ UP  
>>  
>>
```

**MMUL**

```
<<  
→ M1 M2  
<<  
RTN M1 →LST M2 →LST A1 L1 C1 A2 C2 L2  
<<  
IF C1 L2 <> THEN  
  M1 M2 1 1 ERR  
ELSE
```

```

O L 1 1 -
FOR I
  1 C2
  FOR K
    O 1 C1
    FOR J
      A1 I C1 * J + GET A2 J 1 -
      C2 * K + GET * +
    NEXT
  SMP →FR
NEXT
NEXT
NEXT
L1 C2 * →LIST L1 C2 LST→ UP
END

```

```

»
»
»

```

MINV

```

«
→ M
«
M MDET M MTRN RTN →LST → DT A L C
«

```

```

IF DT TYPE 0 == THEN
  IF DT == 0 THEN
    M 2 ERR
  END
END
1 L
FOR I
  1 L
  FOR J
    A I →EXT DETN I J + 2 MOD 2 *
    1 - NEG * DT / SMP →FR
  NEXT
NEXT
L SQ →LIST LL LST→ UP

```

```

»
»
»

```

MTRN

```

«
→ M
«
RTN M →LST → A L C
«
1 C
FOR J
  O L 1 -
  FOR I
    A I C * J + GET
  NEXT
NEXT
L C * →LIST C L LST→ UP
»

```

```

»
»

```

MJCB

```

«
→ A B
«
A SIZE B SIZE → L C
«
1 L
FOR I
  1 C
  FOR J
    A I GET B J GET δ
  NEXT
NEXT
RTN L C * →LIST DETN SMP →FR ABS UP
»
»
»

```

```

»
»
»

```

Menu : RTN

```

→LST
«
DEPTH → N
«
LIST→ → L
«
DUP SIZE → C
«
WHILE DEPTH N > REPEAT
  +
END
L C
»
»
»
»
»
»
»

```

LST→

```

«
→ M L C
«
O L 1 -
FOR I
  1 C
  FOR J
    M I C * J + GET
  NEXT
C →LIST
NEXT
L →LIST
»
»
»

```



```

-EXT
«
  3 PICK SIZE √ → M L C N
  «
    1 N
    FOR I
      IF L <> THEN
        1 N
        FOR J
          IF J C <> THEN
            M I 1 - N * J + GET
          END
        NEXT
      END
    NEXT
    N 1 - SQ →LIST
  »
»

DET2
«
  LIST→ DROP 4 ROLL * 3 ROLL * -
»

DETN
«
  DUP SIZE √ → M C
  «
    IF C 2 > THEN
      0 1
      FOR X
        -1 X ^ NEG M X GET * M 1 X →EXT DETN * +
      NEXT
    ELSE
      IF C 2 == THEN
        M DET2
      ELSE
        M LIST→ DROP
      END
    END
  »
»

ADSUB
«
  → M1 M2 OP
  «
    M1 →LST M2 →LST → A1 L1 C1 A2 L2 C2
    «
      IF L1 L2 == C1 C2 == AND THEN
        1 L1 C1 *
        FOR I
          A1 I GET A2 I GET OP EVAL SMP →FR
        NEXT
        L1 C1 * →LIST L1 C1 LST→
      ELSE
        M1 M2 1 ERR
      END
    »
  »
»

```

```

»
»
»
ERR
«
  → E
  «
    { " Invalid Dimension" " Infinite Result" }
    E GET 1 DISP 600 .1 BEEP 1 WAIT UP ABORT
  »
»

```

Menu : HOME

```

UP
«
  PATH DUP SIZE 1 - DUP 0 == + GET EVAL
»

```

```

SMP
«
  DO
    DUP EXPAN DUP 3 ROLL UNTIL SAME
  END
  DO
    DUP COLCT DUP 3 ROLL UNTIL SAME
  END
»

```

Je tiens à dire que je ne suis que l'expéditeur de ces programmes. L'auteur est Thierry Haurie, collègue de maîtrise EEA.

Ronic Chiche (501)

## GENERATEUR D'HISTOGRAMMES

Ce programme permet la visualisation de données statistiques sous forme d'histogrammes : il complète la fonction DRWΣ qui trace un nuage de points. Bien que destiné au HP-28S, pour des raisons de mémoire (une matrice de 50x2 occupe plus de la moitié de la Mem de la HP-28C) et de certaines fonctions spécifiques à ce modèle, il peut être modifié pour tourner sur C. Il est nécessaire de rajouter le contenu des doubles parenthèses et d'enlever DGT12 et LCD->.

Le début du programme calcule les paramètres de traçage ainsi que le PPAR puis cadre le tracé. La deuxième partie effectue le traitement de DAT et prépare les données statistiques au traçage effectué par 3 boucles U (up), H (horizon), D (down).

La fonction DGTIZ permet le traitement du graphique obtenu de la même façon que DRWE : utilisation du curseur graphique.

On trace un histogramme avec des données statistiques en intervalles; or le calculateur ne connaît pas la notion d'intervalles en STAT : à chaque  $X_i$  correspond un  $Y_i$ . Il faut donc ajouter une constante quelconque à la fin des données afin d'obtenir un N pair.

Exemple :

```
Xi   1<----->2<----->3<----->4
Yi       2       .5       6       cte
```

c.à.d. [[1 2] [2 .5] [3 6] [4 cte]] en sachant que 2 est compris entre 1 et 2, ou que 6 est compris entre 3 et 4.

Utilisation :

- 1) Données statistiques dans ΣDAT.
- 2) [HISTΣ]
- 3) Traitement par curseur.

```
«
MAXΣ DUP 2 ((1 ->LIST)) GET DUP
3.1 / *H 31 / SWAP 1 ((1 ->LIST)) GET DUP
13.6 / *W 13.6 / -> k j
«
PPAR 2 GET 0 k R->C - CENTR CLLCD SPEED
DRAX
1 NΣ 1 -
FOR I
  ΣDAT I -1 2 ->LIST GET ΣDAT
  I 1 + 1 2 ->LIST GET -> x x1
«
  ΣDAT I 2 2 ->LIST GET -> y
  «
    0 y
    FOR U
      x U R->C PIXEL
      k
    STEP
    x x1
    FOR H
      H y R->C PIXEL
      j
    STEP
    y 0
    FOR D
```

```
x1 D R->C PIXEL
k NEG
STEP
»
»
NEXT
»
DGTIZ
'PPAR' PURGE
»
'HISTΣ' STO
```

Khôi-Nguyễn DANG-TRUNG (538)

## 007 : PERMIS DE CODER

Ce programme sert à coder/décoder des chaînes de caractères : vous introduisez la chaîne à coder/décoder et le mot clé, vous exécutez 007 et vous obtenez la chaîne codée/décodée (la transformation est involutive). Vous n'êtes pas limités à des données de type string : si par exemple vous avez absolument besoin de votre génial programme qui transforme votre HP-28 en friteuse mais que vous craignez qu'il ne tombe sous des regards indiscrets, transformez tout ce qui le concerne à l'aide de ASTR et codez le. Pour l'utiliser, décodez le et appliquez STRA (et une fois qu'il sera au point, publiez le dans JPC, ça m'intéresse).

Entrée :

niveau 1 : chaîne à coder (ou décoder)  
niveau 2 : chaîne servant de clé

Sortie :

niveau 1 : chaîne codée (ou décodée)

007

```
«
OVER SIZE OVER SIZE /
LN 2 LN / CEIL 1 SWAP
START
DUP +
NEXT
OVER SIZE 1 SWAP SUB XOR
»
```

En principe, plus le mot clé est long, plus le code est dur à décrypter. Toutefois, je n'ai pas fait d'étude approfondie sur ce point. J'espère juste qu'il n'y a pas de méthode ridiculement simple pour décrypter le codages générés par 007.

MDLR : Faites attention le nom du programme est 0-0-7 (deux lettres 0 suivi d'un 7) et non zéro-zéro-7, sinon la HP-28 croit qu'il s'agit d'une constante numérique.

Guillaume Le Stum (432)

## FACTORIELLES ACTE II

Bien que n'étant pas adhérent de PPC Paris, je voudrais ici vous proposer une version optimisée du programme FACP (Factorielle Précise) de Jérôme Devémy paru dans *JPC 65*.

En effet, le calcul d'une factorielle est assez long, ne serait-ce que par la nature de la factorielle elle-même. Seul un programme en langage machine resterait "confortable".

Aussi, après avoir recopié FACP dans ma HP-28, j'ai entrepris une chasse au temps et à l'octet.

Résultat: 20 secondes de moins sur les trois minutes nécessaires pour le calcul de 100! en double vitesse (programme SPEED).

Listing du programme:

```
«
{ 1 } 'FMP' STO 0 'R' STO
2 SWAP
FOR b
1 FMP SIZE
FOR e
'FMP(e)' EVAL b * R +
100000000 / DUP FP
100000000 * 'FMP(e)' STO
IP 'R' STO
NEXT
IF R THEN
FMP T + 'FMP' STO
0 'R' STO
END
NEXT "" FMP SIZE 1
FOR p
FMP p get
DUP ->STR "00000000" IFTE
DUP SIZE 8 - NEG 1 SWAP
"00000000" 3 ROLL SUB
SWAP + +
-1
STEP
```

```
WHILE DUP 1 1 SUB "0" ==
REPEAT DUP SIZE 2 SWAP SUB
END
{ FMP R } PURGE
»
```

NOTE: Je purge la variable FMP qui ne semble pas indispensable. Et merci à Jérôme Devémy pour ce programme!

Yann Rousse

## SWITCH

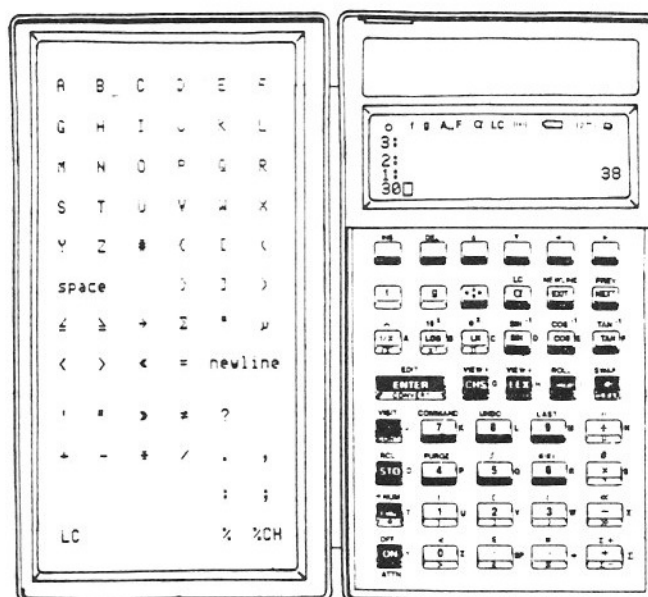
Une remarque sur le programme simple mais maintenant indispensable, SWITCH de Christopher Lishka paru dans *JPC 65*.

On peut l'écrire :

```
«
OVER SWAP POS 1 + GET EVAL
»
```

Bien que le style, la clarté en souffrent, il est plus court et plus rapide (de peu) !

Yann Rousse



FX-45C Casio calculator with the left-hand LCD displaying the HP285 ALPHA keys, and acting as a touchscreen as well. Additional ASCII characters have been added for completeness, as well as the "percent" and "percent change" functions at the bottom. When the "LC" (lower case position) is pressed, all the upper-case characters in the LCD change to lower case (and back to upper case again when LC is pressed again).



## HP-41

B. Wiklund  
M. Cracknell

|                                         |    |
|-----------------------------------------|----|
| Récupération de registres d'assignation | 22 |
| Bataille navale                         | 24 |

## PACKEZ VOS ASSIGNATIONS

Voici un petit programme M-CODE qui, je l'espère comblera une lacune du système d'exploitation HP-41 : packer les registres d'assignation.

En effet, si vous désassignez 2 touches, le HP-41 ne reprendra le registre que si ces 2 assignations sont enregistrées sur un même registre. Ceci est un petit inconvénient qui peut faire perdre des registres...

### Fonctionnement :

- XEQ PKA : 2 possibilités :
  - 1-La machine émet un TONE 89 : appuyez 2 fois sur la touche ON, les registres sont packés.
  - 2-Aucun ne sort de votre HP-41 favorite, les registres sont packés.

### Explications :

Le programme recherche d'abord le registre d'assignation d'adresse absolue la plus élevée. Ensuite, il balaye les registres vers le bas jusqu'à trouver un demi-registre disponible, qu'il sauvegarde. La recherche vers le bas continue pour trouver un second demi-registre disponible, qui sera combiné avec le premier pour gagner un registre. Cette recherche systématique se poursuit jusqu'au bas de la Ram (adresse 192 décimal). S'il n'y a qu'un seul demi-registre disponible, PKA le disposera en fin de Ram (192 décimal). Ce faisant, il n'y aura pas de nouveau registre "ouvert" lors de votre prochaine assignation...

### Bibliographie :

- "Au fond de la HP-41" par J.D. Dodin.
- "PPC Rom's User Manual" (disponible chez EduCalc).
- "HP-41 M-Code for beginners" par K. Emery (disponible chez EduCalc).

### Exemple :

- Supprimez tous les assignements.
- Assignez, dans cet ordre :
  - 1- BEEP à la touche SIN;
  - 2- CHS à la touche COS;
  - 3- ENTER à la touche TAN;
  - 4- LOG à la touche STO;(ou 4 autres fonctions à 4 autres touches).
- GTO ..
- PRGM (00 REG 248 (par exemple)) PRGM
- ASN ALPHA ALPHA SIN

- ASN ALPHA ALPHA TAN
- XEQ PACK
- PRGM (00 REG 248) PRGM

- A ce point vous vous apercevez que, bien que vous ayez libéré l'équivalent d'un registre, le HP-41 ne l'a pas repris en compte dans ses registres libres...

- XEQ PKA
- La HP-41 a beepé, donc : ON ON
- PRGM (00 REG 249)

- Vous constaterez que vous avez récupéré un registre, ce qui peut être utile en cas de longs programmes...

### Remarques :

- Ce programme peut résider n'importe où sur votre MLDL.
  - Ce programme, contrairement à PK de PPC Rom, n'affecte aucun registre Ram et tourne considérablement plus vite.
  - En utilisant les "polling adresses", on doit pouvoir automatiser PKA à chaque mise sous tension.
  - Si vous trouvez des bugs, tenez-moi au courant.
  - PKA est très certainement améliorable...
- J'attends vos suggestions !!

### Listing :

```
B98C 081 "A"
B98D 00B "K"
B98E 010 "P"
B98F 084 CLR F 5
B990 1A0 A=B=C=0
B991 158 M=C ALL
B992 09C PT= 5
B993 0E0 SLCT Q
B994 39C PT= 0
B995 3F8 READ 15(e)
B996 2F2 ?C(MJ) P-Q
B997 027 JC B99B +04
B998 2B8 READ 10({M0})
B999 2F2 ?C(MJ)0 P-Q
B99A 3A0 ?NC RTN
B99B 35C PT= 12
B99C 0A0 SLCT P
B99D 2DC PT= 13
B99E 130 LDI S&X
B99F 0C1 CON: 193
B9A0 106 A=C S&X
B9A1 270 RAM SLCT
B9A2 038 READ DATA
B9A3 222 C=C+1 @PT
B9A4 043 JNC B9AC +08
B9A5 0E0 SLCT Q
B9A6 262 C=C-1 @PT
B9A7 02B JNC B9AC +05
B9A8 166 A=A+1 S&X
```



B9A9 0A6 A<>C S&X  
 B9AA 0A0 SLCT P  
 B9AB 3AB JNC B9A0 -0B  
 B9AC 1A6 A=A-1 S&X  
 B9AD 0A6 A<>C S&X  
 B9AE 270 RAM SLCT  
 B9AF 0A6 A<>C S&X  
 B9B0 038 READ DATA  
 B9B1 284 CLRf 7  
 B9B2 104 CLRf 8  
 B9B3 244 CLRf 9  
 B9B4 0A0 SLCT P  
 B9B5 39C PT= 0  
 B9B6 0E0 SLCT Q  
 B9B7 31C PT= 1  
 B9B8 2F2 ?C{M}0 P-Q  
 B9B9 02F JC B9BE +05  
 B9BA 288 SETf 7  
 B9BB 248 SETf 9  
 B9BC 09C PT= 5  
 B9BD 04A C=0 PT<-  
 B9BE 0A0 SLCT P  
 B9BF 15C PT= 6  
 B9C0 0E0 SLCT Q  
 B9C1 29C PT= 7  
 B9C2 2F2 ?C{M}0 P-Q  
 B9C3 047 JC B9CB +08  
 B9C4 108 SETf 8  
 B9C5 248 SETf 9  
 B9C6 0A0 SLCT P  
 B9C7 15C PT= 6  
 B9C8 0E0 SLCT Q  
 B9C9 2DC PT= 13  
 B9CA 052 C=0 P-Q  
 B9CB 2EE ?C{M}0 ALL  
 B9CC 01F JC B9CF +03  
 B9CD 088 SETf 5  
 B9CE 0AB JNC B9E3 +15  
 B9CF 24C ?FSET 9  
 B9D0 09B JNC B9E3 +13  
 B9D1 244 CLRf 9  
 B9D2 14C ?FSET 6  
 B9D3 0B7 JC B9E9 +16  
 B9D4 148 SETf 6  
 B9D5 3EE LSHFA ALL  
 B9D6 3EE LSHFA ALL  
 B9D7 3EE LSHFA ALL  
 B9D8 08E B=A ALL  
 B9D9 38E RSHFA ALL  
 B9DA 38E RSHFA ALL  
 B9DB 38E RSHFA ALL  
 B9DC 28C ?FSET 7  
 B9DD 02F JC B9E2 +05  
 B9DE 13C RCR 8  
 B9DF 2DC PT= 13  
 B9E0 3D0 LD@PT- F  
 B9E1 010 LD@PT- 0  
 B9E2 158 M=C ALL

B9E3 1A6 A=A-1 S&X  
 B9E4 130 LDI S&X  
 B9E5 0C0 CON: 192  
 B9E6 306 ?A<C S&X  
 B9E7 233 JNC B9AD -3A  
 B9E8 113 JNC BAOA +22  
 B9E9 086 B=A S&X  
 B9EA 144 CLRf 6  
 B9EB 10C ?FSET 8  
 B9EC 04F JC B9F5 +09  
 B9ED 3CE RSHFC ALL  
 B9EE 3CE RSHFC ALL  
 B9EF 3CE RSHFC ALL  
 B9F0 3CE RSHFC ALL  
 B9F1 3CE RSHFC ALL  
 B9F2 3CE RSHFC ALL  
 B9F3 29C PT= 7  
 B9F4 010 LD@PT- 0  
 B9F5 10E A=C ALL  
 B9F6 198 C=M ALL  
 B9F7 370 C=C OR A  
 B9F8 088 SETf 5  
 B9F9 0AE A<>C ALL  
 B9FA 0C6 C=B S&X  
 B9FB 270 RAM SLCT  
 B9FC 0AE A<>C ALL  
 B9FD 2F0 WRIT DATA  
 B9FE 0CE C=B ALL  
 B9FF 03C RCR 3  
 BAO0 270 RAM SLCT  
 BAO1 0EE C<>B ALL  
 BAO2 04E C=0 ALL  
 BAO3 2DC PT= 13  
 BAO4 3D0 LD@PT- F  
 BAO5 2F0 WRIT DATA  
 BAO6 0C6 C=B S&X  
 BAO7 0A6 A<>C S&X  
 BAO8 02E B=0 ALL  
 BAO9 2D3 JNC B9E3 -26  
 BAOA 2DA ?B{M}0 M  
 BAOB 08B JNC BA1C +11  
 BAOC 3AE RSHFB ALL  
 BAOD 3AE RSHFB ALL  
 BAOE 3AE RSHFB ALL  
 BAOF 066 A<>B S&X  
 BA10 130 LDI S&X  
 BA11 0C0 CON: 192  
 BA12 270 RAM SLCT  
 BA13 366 ?A{M}C S&X  
 BA14 033 JNC BA1A +06  
 BA15 038 READ DATA  
 BA16 1D8 C<>M ALL  
 BA17 WRIT DATA  
 BA18 0A6 A<>C S&X  
 BA19 270 RAM SLCT  
 BA1A 198 C=M ALL  
 BA1B 2F0 WRIT DATA  
 BA1C 08C ?FSET 5

BA1D 04B JNC BA26 +09  
 BA1E 130 LDI S&X  
 BA1F 000 CON: 0  
 BA20 270 RAM SLCT  
 BA21 130 LDI S&X  
 BA22 059 CON: 89  
 BA23 358 ST=C XP  
 BA24 341 ?NC XQ  
 BA25 048 ->12D0  
 BA26 084 CLRf 5  
 BA27 144 CLRf 6  
 BA28 284 CLRf 7  
 BA29 104 CLRf 8  
 BA2A 244 CLRf 9  
 BA2B 3E0 RTN

Heureuse Programmation a tous !

Bertil Wiklund (508)

P.S. Mon MLDL est un double module HEPAX. Ce module est tout simplement génial !

## BATAILLE NAVALE

La bataille navale est un vieux loisir joué avec différentes tailles de cartes et autant de règles différentes. Cette version est pour un joueur seul contre la machine.

La configuration requise est : HP-41 + Time + PPC Rom + Imprimante 24 colonnes.

Vous, le joueur, avez un stock limité de munitions, sous la forme de vingt salves de trois coups, et un temps limité de une heure, pour couler une flotte ennemie se composant d'un porte avions, deux croiseurs, trois vedettes lance-torpilles et quatre sous-marins.

Cette version contient un grand nombre de lignes synthétiques. Vous pouvez noter que leur usage économise du temps, particulièrement les Labels Locaux "A" - "E". Si vous avez des problèmes pour substituer des lignes synthétiques par des lignes classiques, n'hésitez pas à me contacter, je ferais de mon mieux pour résoudre votre problème. Si vous n'avez pas de PPC-Rom ou d'imprimante 24 colonnes faites le moi savoir, et je tenterais de modifier le programme pour convenir à votre système. Envoyez un support magnétique (cartes/cassette/disquette 3½") et si je réussis, je vous retournerais la version modifiée.

La carte consiste en une grille de carrés de 10 par 10. La HP-41 observe certaines règles pour positionner sa flotte :

- 1) Deux bateaux ne peuvent pas se toucher par un coin ou par les cotés.
- 2) Un bateau, ou une partie de bateau, ne peut pas résider dans le carré 00.
- 3) Un bateau ne peut pas être placé en diagonale.

Quand vous aurez identifié la position d'un bateau, vous pouvez faire une croix sur tous les carrés l'entourant. Les pas listés sous LBL C (pas 547), placeront une croix dans les carrés indiqués. Si vous devez changer d'idée, Les pas listés sous LBL D (pas 556) effaceront les carrés indiqués. Si vous placez une croix ou un blanc dans un carré occupé, les pas listés sous LBL B (pas 515) remplaceront l'ancienne valeur. Si vous avez oublié la carte, ou l'avez corrigée, ou la réimprimer avant votre prochaine salve, appuyez sur [shift][e], une nouvelle sera réimprimée. mais rappelez vous que l'horloge tourne toujours. Si vous perdez patience, ou êtes limités par le temps, ou voulez savoir où vous vous êtes trompés, appuyez sur [shift][a] et tout sera révélé.

Le comptage des points est simple. Un score de 1000 indique un coup au but sur le porte-avions, 100 indique un coup sur un croiseur, 10 indique un coup sur un vedette, et 1 indique que vous avez coulé un sous marin. Cette version da bataille navale n'autorise pas de toucher un bateau plus d'une fois par salve; la carte enregistrera les deux coups mais seulement un seul sera comptabilisé. Dans l'exemple, un score de 200 à la salve 9 indique que deux croiseurs ont été touchés, et non deux coups sur le même croiseur.

Ce programme est basé sur un programme HP-97, écrit, je crois, par Peter Amlinger. Je remercie aussi les créateurs de PPC-Rom, sans leurs efforts ce programme n'aurait pas été possible.

Et voici les instructions précises, un extrait d'une partie et le listing.

### Instructions d'utilisation :

- 1) XEQ Alpha "SBP" Alpha [USER]
- 2) Entrez la racine du nombre aléatoire puis [A]

Après une petite attente, la carte initiale sera prête.

- 3) Entrez le premier coup puis ENTER^  
 Entrez le deuxième coup puis ENTER^  
 Entrez le troisième coup puis [E]

Après une courte période la nouvelle carte sera imprimée. Avant chaque carte, le numéro de la salve, la position de chaque coup, et le score de chaque salve sont imprimés. Après la carte, la liste des coups précédents est imprimée afin de vous rappeler quels bateaux on étés touchés, et à quelle salve. Finalement, le temps écoulé est affiché et la HP-41 est prête pour la salve suivante. Le processus sera répété tant que vous aurez soit coulé tous les bateaux ennemis, soit épuisé toutes vos munitions ou le temps imparti.

- 4) Pour placer une croix dans un carré : entrez ses coordonnées et appuyez sur [C].
- 5) Pour effacer un carré : entrez ses coordonnées appuyez sur [D].
- 6) Pour placer un nombre dans un carré : entrez ce nombre, appuyez sur ENTER^ puis entrez les coordonnées et appuyez sur [B].
- 7) Pour réimprimer la carte : appuyez sur [shift] [e].
- 8) Pour révéler la position de la flotte ennemie : appuyez sur [shift] [a].

#### Un exemple de partie :

Pour jouer la même partie, entrez .01011983 et appuyez sur [A]. Après un moment, la carte initiale sera imprimée.

Salve 1 : 14 ENTER^ 77 ENTER^ 51 [E] - Un peu de chance, le score de 100 indique un coup sur un croiseur.

Salve 2 : Le croiseur peut être en 14-15-16 alors essayons 15 ENTER^ 95 ENTER^ 20 [E] - Un score de 10 indique montre un coup sur une vedette mais qu'on a manqué le croiseur.

Salve 3 : Le croiseur pourrait être en 67-77-87 alors essayons 67 ENTER^ 38 ENTER^ 83 [E]. Par chance, le score de 210 indique un coup sur un croiseur et une vedette. comme 03 n'apparaît pas après 02, la vedette qui vient d'être touchée n'est pas la même que celle touchée lors de la salve 2.

Salve 4 : Essayons 87,32,59. Un score de 1000 montre qu'on vient de toucher le porte avions et que le croiseur n'est pas en 67-77-87.

Salve 5 : Essayons 57,28,70 pour avoir le porte avions. Le score de 1000 montre qu'on a encore touché le porte avions. Il est en 56-57-58-59 car c'est la seule position où qui corresponde avec les coup 4 et 5. Maintenant nous pouvons placer une croix dans les carrés 45-49,55,65,66,68,69. Pour faire ceci, nous entrons les coordonnées puis [C].

Salve 6 : 58,52,85. Le score de 1000 montre que nous avons mis deux coups dans l'océan et le coup attendu dans le porte avions.

Salve 7 : Essayons de faire un double coup au but en touchant le croiseur que nous avons touché à la salve 1 et en coulant le porte avions, tirons donc 61,56,8. Un score de 1100 montre le succès.

Salve 8 : La vedette que nous avons touché en Salve 2 peut être en 21,22 alors tentons 41,92,21. Le score de 10 montre un coup au but sur un vedette mais il n'est pas sûr que ce soit la même vedette que pour la salve 2. Oublier les croiseurs pendant cette salve n'est pas une perte totale car la seule place où deux coups sur un croiseur sont suffisamment proches pour être le même bateau est en 51 et 61. Si 41 est un échec, cela veut dire qu'un croiseur est en 51-61-71, nous pouvons donc placer une croix en 40,42,50,52,60,72,80,81,82 de la même manière que nous l'avons fait avant de tirer la salve 6.

Salve 9 : Maintenant que nous avons fixé a position d'un des croiseurs, il est plus sûr de sur les deux pendant la même salve. Un coup est attendu, et un score de 200 ou plus montrera que les deux croiseurs seront prêts à aller visiter les domaines du capitaine Némé. nous tirons en 71,37,16. Succès, un score de 200 montre qu'un croiseur a coulé et que le deuxième est sévèrement endommagé.

Salve 10 : Le croiseur restant peut être soit en 36-37-38, soit en 37-38-39 parce que 37 et 38 sont les deux seuls carrés où un but dans les salves 03 et 09 sont suffisamment proches. Essayons 39,23,74. Le score de 110 indique un coup final sur le croiseur et un coup sur une vedette.

Salve 11 : Nous n'avons pas encore coulé de sous marin, alors nous allons tirer un coup dans les espaces vierges - qui sait ce qui peut se terrer derrière les vagues. 64,91,76 semble être un bon coup. Le score de 1 montre qu'on en a eu 1, et qu'il en reste 3.

Je vous laisse finir la partie. Il reste seulement 6 coups à atteindre, et vous avez encore 9 salves à tirer. Vous avez donc une bonne chance.

J'ai deux autres versions de ce jeu, une pour l'interface vidéo 32 colonnes, et une pour l'interface Mountain Computer. Les instructions d'utilisation sont les mêmes, seule la présentation des cartes change. Si vous voulez une de ces versions, envoyez moi une enveloppe auto-adressée pour un listing. Si vous y ajoutez un support magnétique, je vous l'enregistrerais.

J'espère que ce jeu vous donnera autant de plaisir dans le futur qu'il m'en a donné pendant les 9 dernières années.

### Affichages de la partie (extrait)

SALVO 1

14 77 51

S/SCORE=100

MC# 0 1 2 3 4 5 6 7 8 9#

#####

00#XX #

10# 01 #

20# #

30# #

40# #

50# 01 #

60# #

70# 01 #

80# #

90# #

#####

T/SCORE=100

B#

C# 01

T#

S#

0:04:17

SALVO 2

15 95 20

S/SCORE=10

MC# 0 1 2 3 4 5 6 7 8 9#

#####

00#XX #

10# 0102 #

20#02 #

30# #

40# #

50# 01 #

60# #

70# 01 #

80# #

90# 02 #

#####

T/SCORE=110

B#

C# 01

T# 02

S#

0:11:30

SALVO 5

57 28 70

S/SCORE=1000

MC# 0 1 2 3 4 5 6 7 8 9#

#####

00#XX #

10# 0102 #

20#02 05 #

30# 04 03 #

40# #

50# 01 05 04#

60# 03 #

70#05 01 #

80# 03 04 #

90# 02 #

#####

T/SCORE=2210

B# 04 05

C# 01 03

T# 02

S#

0:22:08

SALVO 7

61 56 8

S/SCORE=1100

MC# 0 1 2 3 4 5 6 7 8 9#

#####

00# X 07 #

10# 0102 #

20#02 05 #

30# 04 03 #

40# XXXXXXXXX#

50# 0106 XX07050604#

60# 07 XXXX03XXXX#

70#05 01 #

80# 03 06 04 #

90# 02 #

#####

T/SCORE=4310

B# 04 05 06 07

C# 01 03 07

T# 02

S#

0:27:24

SALVO 11  
64 91 76  
S/SCORE=1

MC# 0 1 2 3 4 5 6 7 8 9#  
#####  
00# X 07 #  
10# 010209 #  
20#0208 10 05 #  
30# 04 090310#  
40#XX08XX XXXXXXXXX#  
50#XX0106 XX07050604#  
60#XX07XX 11XXXX03XXX#  
70#0509XX 10 1101 #  
80#XXXXXX03 06 04 #  
90# 1108 02 #  
#####

T/SCORE=4631

B# 04 05 06 07  
C# 01 03 07 09 10  
T# 02 08 10  
S# 11  
0:41:57

# Listing :

01\*LBL "SBP"  
SF 27 RTN

04\*LBL A  
CF 29 PWRDN 156 PSIZE FIX 0 RDN  
SF 04 CLRG STO 20 55 XROM "IF" CF 00  
10 STO 22 X^2 STO 23 20 STO 25 36  
STO 32 40 STO 33 46 STO 34 52 STO 35  
36,155 " " RCL b ASTO IND Y ISG Y  
STO b

37\*LBL 01  
DSE 25 9 RCL 25 X=Y? GTO 02 0 SETSW  
XEQ C CF 01 55 XROM "IF" BEEP GTO 14

51\*LBL 02  
0 STO 00 19 RCL 25 X=Y? XEQ 03 17  
X<=Y? XEQ 03 RCL 25 14 X<=Y? XEQ 03  
20 XROM "RN" RCL 23 \* INT X=0? GTO 02  
STO 01 XEQ 05 FS?C 02 GTO 02 RCL 01  
STO IND 25 RCL 00 X=0? GTO 01 20  
XROM "RN" ,5 X>Y? SF 02 1 FS?C 02  
RCL 22 STO 03 20 XROM "RN" ,5 X>Y?  
SF 02 RCL 03 FS?C 02 CHS STO 03

99\*LBL 10  
RCL 01 RCL 03 + RCL 22 / FRC RCL 01  
RCL 22 / FRC - ABS ,9 X=Y? GTO 02  
RCL 03 ST+ 01 0 RCL 01 X<=Y? GTO 02  
RCL 23 X<=Y? GTO 02 XEQ 05 FS?C 02 GTO 02

RCL 23 ST/ IND 25 RCL 01 ST+ IND 25 1 E  
ST- 00 RCL 00 X=0? GTO 01 GTO 10

137\*LBL 03  
1 E ST+ 00 RCL 25 RTN

142\*LBL 05  
X<> 25 STO 21 20 STO 25

147\*LBL 00  
DSE 25 9 RCL 25 X=Y? GTO 08 RCL 25  
RCL 21 X=Y? GTO 00 RCL IND 25 X=0?  
GTO 08

160\*LBL 07  
STO 02 INT X=0? GTO 00 RCL 01 FS? 00  
GTO 11 - ABS 2 X>Y? GTO 06 RDN RCL 22  
- ABS 2 X>Y? GTO 06

180\*LBL 12  
RCL 02 FRC RCL 23 \* GTO 07

186\*LBL 06  
SF 02

188\*LBL 08  
RCL 21 X<> 25 RTN

192\*LBL 11  
X=Y? GTO 12 1 E STO 09 13 RCL 25 X<=Y?  
GTO 13 RCL 22 ST\* 09 16 RCL 25 X<=Y?  
GTO 13 RCL 22 ST\* 09 18 RCL 25 X<=Y?  
GTO 13 RCL 22 ST\* 09

215\*LBL 13  
CLA RCL 09 ST+ 07 32 ENTER^ 1 E3 XEQ 09  
33 ENTER^ 1 E2 XEQ 09 34 ENTER^ 1 E1  
XEQ 09 35 ENTER^ 1 E

234\*LBL 09  
RCL 09 X=Y? RTN RDN XEQ 03 RCL T  
RCL IND X ASTO IND X ISG IND Y RTN GTO 03

246\*LBL E  
PWRDN FS?C 04 RUNSW ISG 26 "" STO 06  
STO 31 RDN STO 05 STO 30 RDN STO 04  
STO 29 CF 21 AVIEW SF 21 50 XROM "IF"  
RCLSW 1 E X<Y? GTO a 55 XROM "IF" 56  
ST+ 29 ST+ 30 ST+ 31 CLA XEQ 03  
ASTO IND 29 ASTO IND 30 ASTO IND 31 0  
STO 03 STO 07 STO 25 SF 00 RCL 04 STO 01  
XEQ 05 RCL 05 STO 01 XEQ 05 RCL 06 STO 01  
XEQ 05 RCL 07 ST+ 08 55 XROM "IF"

298\*LBL e  
SF 21 PWRUP SF 12 RCL 26 BEEP "SALVO "  
ARCL X PRA CLA ARCL 04 "- "  
ARCL 05 "- " ARCL 06 PRA "S/SCORE="

ARCL 07 PRA ADV

318\*LBL 14

SF 21 CF 12 PWRUP "MC# 0 1 2 3 "

"-4 5 6 7 8 9#" PRA XEQ 17 9 E-3 STO 27  
56,065

329\*LBL 16

STO 28 INT 56 - CLA X=0? "-0" ARCL X

"-#" ACA RCL 28 CLA

342\*LBL 15

ENTER^ "X" ASTO X RCL IND Y X=Y? GTO 14

CLA ARCL X ACA RCL Z ISG X GTO 15

GTO 13

356\*LBL 14

SF 12 ACA CF 12 RCL Z ISG X GTO 15

363\*LBL 13

"#" ACA PRBUF 1 E-2 + ISG 27 GTO 16

XEQ 17 FS? 04 GTO 13 ADV SF 12 "T/SCORE="

ARCL 08 PRA ADV

380\*LBL 04

36,039 "B" XEQ 14 40,045 "C" XEQ 14

46,051 "T" XEQ 14 52,055 "S" XEQ 14

GTO 13

394\*LBL 14

"-#" SF 12 ACA CF 12 " " ACA CLA

RCL b ARCL IND Y "- " ISG Y STO b

ACA PRBUF RTN

410\*LBL 13

FS?C 06 GTO 13 FS? 04 GTO 13 ADV SF 12

CLA RCLSW INT ARCL X XEQ 14 XEQ 14 ACA

ADV RCLSW 1 E X<Y? GTO a 08 4664 X<=Y?

GTO a RCL 26 20 X>Y? GTO 13

437\*LBL a

SF 21 ADV STOPSW BEEP SF 12 "SHIPS" PRA

ADV CF 12 19 XEQ 25 ASTO 36 XEQ 09

ASTO 37 XEQ 09 ASTO 38 XEQ 09 ASTO 39 18

XEQ 25 ASTO 40 XEQ 09 460 ASTO 41 XEQ 09

ASTO 42 17 XEQ 25 ASTO 43 XEQ 09 ASTO 44

XEQ 09 ASTO 45 16 XEQ 25 ASTO 46 XEQ 09

ASTO 47 15 XEQ 25 ASTO 48 XEQ 09 ASTO 49

14 XEQ 25 ASTO 50 XEQ 09 ASTO 51 13

XEQ 25 ASTO 52 12 XEQ 25 ASTO 53 11

XEQ 25 ASTO 54 10 XEQ 25 ASTO 55 SF 06

GTO 04

499\*LBL 14

"-:" 1 E1 LASTX FRC 1 E2 \* INT

GTO 14

508\*LBL 09

LASTX FRC 1 E2 \* INT GTO 09

515\*LBL B

SF 01 56 + X<>Y XEQ 09 ASTO IND Z

GTO 13

523\*LBL 03

1 E1 RCL 26 GTO 14

527\*LBL 25

RCL IND X INT

530\*LBL 09

CLA 1 E1 X<>Y

534\*LBL 14

X<Y? "-0" ARCL X RTN

539\*LBL 17

"#####" ASTO X ARCL X ARCL X ARCL X

PRA RTN

547\*LBL C

SF 01 56 + "X" ASTO IND X FS? 04 RTN

GTO 13

556\*LBL D

SF 01 56 + " " ASTO IND X

562\*LBL 13

FC? 01 XROM "PO" "SALVO " RCL 26 1 E +

ARCL X "- \*\*\*\*" XROM "T1" CF 21 AVIEW

FC?C 01 PWRDN

END

Mark Cracknell (HPCC)  
6 Stratton News  
Off South st./ Leighton Buzzard  
Bedfordshire LU7 8NT  
GRANDE BRETAGNE



Le Journal JPC est le bulletin de liaison entre les membres de l'Association "PPC Paris", régie par la loi de 1901. Le Club est éditeur de JPC, et son siège social est au 56, rue Jean-Jacques Rousseau, 75001 Paris.

La maquette de ce numéro a été préparée et réalisée par Jacques Belin, Laurent Chouraki et Jean Reibel grâce à un système comprenant un HP71B, un lecteur de disquettes HP9114A, un HP9807A, deux HP9154 et une imprimante LaserJet Séries II.

Les dessins sont de Jean-Jacques Dhénin et Paul Courbis.

Directeur de la publication : Jean Reibel  
Numéro ISSN: 0762 - 381X

Veuillez adresser toute correspondance à :  
PPC Paris, BP 604, 75028 Paris Cedex 01.

Imprimé par Copy-Express, 42 86 91 94.