



# EDITORIAL

Chers PPC-PCistes,

Voici le JPC nouveau, avec son air de vacances. Quatre-vingt pages bien remplies !

Il va falloir vous habituer. Dorénavant, notre contrat avec notre imprimeur prévoit 75 pages par mois. Ceci sans augmentation de la cotisation, signalons-le.

Mais maintenant, il va falloir remplir ces 75 pages ! Nous vous faisons confiance. De plus en plus d'articles nous parviennent, témoignant ainsi de la vitalité de notre Club.

En ce moment de vacances, pensez aussi à ceux qui passent leurs week-ends à rédiger ce journal. Aidez-nous, en nous faisant parvenir vos articles sur support magnétique. Cela nous évitera de les retaper.

Saluons la naissance d'une nouvelle rubrique, pour les 71istes: "Le coin des Lhex". Désormais, chaque mois, tous les lecteurs pourront rentrer dans leur ordinateur favori les programmes écrits en assembleur. Plus besoin de module Forth/Assembleur pour goûter les joies du Lex !

Vous trouverez dans le compte-rendu de la réunion PPC de Paris quelques détails quant à nos relations avec HP. Celles-ci sont au beau fixe depuis le Sicob. Cela se traduit par notre présence à l'espace des "4-temps" au Sicob d'automne.

Mais HP pourrait faire encore plus: au Danemark, par exemple, PPC est cité dans les documents publicitaires HP. Enfin ! N'avançons pas plus vite que la musique. Les résultats obtenus sont déjà très prometteurs...

Peut-être vous souvenez-vous d'une nouvelle qui avait entâchée notre présence au Sicob de l'année passée (1984): nous avions eû un vol de matériel. Nous pouvons maintenant vous annoncer que HP et Jean-Claude Becker ont été remboursés intégralement. PPC-PARIS tient ses engagements.

Enfin, pour l'anecdote, je tiens à vous signaler que ce numéro occupe plus de 300Ko de textes sur une disquette.

J'espère avoir à vous réserver de la place sur la disquette !

A bientôt  
Pierre DAVID

## PPC PARIS SE REUNIT UNE FOIS PAR MOIS

Comme vous le savez peut être déjà, PPC PARIS se réunit une fois par mois, en plein cœur de Paris. Amenez votre matériel, votre bonne volonté et vos idées ! Plus vous en apporterez, et plus vous en trouverez chez vos collègues de PPC. Ces réunions se déroulent de manière très libre, aucun ordre du jour, discussion ou autre n'étant imposé. Un membre du Bureau est toujours présent. Ainsi, si vous désirez remettre votre article tout frais au journal, si vous avez des suggestions à faire, si vous voulez vous procurer des anciens numéros de JPC, ce sera en principe toujours possible.

Si donc cela vous intéresse, n'hésitez plus un seul instant venez nous rejoindre tous les premiers samedi de chaque mois, au CENTRE DE JEUNESSE ET DE LOISIRS JEAN VERDIER  
11, rue de Lancry  
75010 PARIS

Et en montant au deuxième étage, vous entendrez des éclats de rire et des discussions passionnées vers la salle 215. Attention, toutefois, de venir entre 16 et 19 h.

Pour l'accès en métro, trois possibilités s'offrent à vous à savoir:

Métro Strasbourg St Denis: sortie Porte St Martin / Bld St Denis, coté pairs.

Métro République: sortie Bld St Martin, coté pairs.

Métro J. Bonsergent: Sortie Bld Magenta, coté impairs.

Ah, j'oubliais: JPC est distribué en avant-première lors de ces réunions ... A bon entendeur Salut !

Les dates des prochaines réunions sont: (16h-19h)

- Le Samedi 7 Septembre 1985

- Le Samedi 5 Octobre 1985

A Bientôt...  
Pierre DAVID

## REUNION PPC A PARIS

Les 22 et 23 juin derniers s'est tenue une réunion des Clubs PPC à Paris, au CISP, 6 avenue Maurice Ravel, dans le 12ème arrondissement (Si si, rappelez-vous, c'est là que s'est tenue notre dernière Assemblée Générale).

L'organisation, très efficace, était de Georges-Noël NICOLAS. Durant deux jours, des orateurs se sont succédés pour parler de sujets aussi divers que "Les Clubs PPC", "La politique d'HP", "Le Basic du HP71".

Pour la première fois, HP était avec nous, en les personnes de MM. Eric CLEMENT, Robert BAYLE, et Philippe CHAILLOT. "La Règle à Calcul" était représentée, ainsi que "Maubert Electronique".

Tout à commencé le samedi matin aux alentours de 9 heures, par le déballage du matériel. Après cette petite période d'installation, J.D. DODIN, Président de PPC-T, et Philippe GUEZ, par la voix de Laurent ISTRIA nous ont parlé des Clubs PPC, de leur passé, de leur présent, et de leurs projets.

M. Eric CLEMENT a ensuite présenté la politique d'HP vis-à-vis des Clubs, une ouverture sensible vers nous, qui se traduit immédiatement par un don de module "Translator Pac". A ensuite été évoquée notre participation au Sicob d'Automne, sur le stand des "4 temps", au même titre que les autres partenaires d'HP. Enfin, la mise en place de dépliant PPC dans les boîtes de matériels HP a été annoncée.

Comme nous pouvons le voir, HP avance dans la bonne direction...

Ce fut ensuite le tour de J.J. DHENIN de parler du Basic. Rétroprojecteur à l'appui, Jean-Jacques nous a exposé le Basic tel qu'il existait avant, puis la norme ANSI, et enfin son application sur le HP71B. Très apprécié, cette intervention nous a appris qu'il existait, dans le "Basic complet", une structure DO..LOOP.. qui est en fait la structure REPEAT..UNTIL et WHILE.. du Pascal.

Il est à noter la compatibilité des Basics HP: le Basic Complet existe à l'heure actuelle sur les machines de la série 200, et on y retrouve le Basic du HP71 avec module Maths.

Ensuite, nous avons ouvert le carton qui contenait les tee-shirts HP. La distribution a conduit à un petit quart d'heure de pause.

Nous nous sommes ensuite détendus en écoutant Jean-François GARNIER parler de ses réalisations: Une interface vocale donnant la voix à tous les contrôleurs HPIL: HP41, HP71 et HP75...

Une interface HPIL-Centronics, qui donne la possibilité de programmer une petite table traçante du type X710, par exemple, en HPGL,

Un modem à couplage direct, et composition automatique, qui sera commercialisé par la firme grenobloise KRISTAL, où Jean-François travaille en tant qu'ingénieur.

Il était l'heure de manger, tout était (bien) prévu. Les repas étaient pris au self-service de centre dans une ambiance très décontractée.

Puis vers 13h30 après un bon café, tout le monde est revenu s'asseoir pour écouter la suite des événements.

C'était le tour de Jean-Claude BORNES. Il est en train de réaliser une interface HPIL (==) Minitel. C'est fantastique, bientôt nous aurons une véritable interface vidéo, un modem et pour les heureux possesseurs de Titan: un clavier en plus. Tout cela pour la somme de 2000 F environ.

Ensuite, après le petit quart d'heure de pause, Pierre DAVID s'est installé près du rétro-projecteur pour nous faire un exposé sur FORTH.

Tout à commencé vers la fin des années 60, quand Charles H. MOORE décida de créer son propre langage. Depuis, de standard en standard, Forth s'est peu à peu affiné pour arriver à la forme actuelle qu'il revêt dans le HP71. Pierre est quelqu'un qui connaît très bien le Forth, pour en avoir écrit un sur un gros ordinateur (DPS8 pour les connaisseurs...). Il a d'ailleurs obtenu une mention pour cela (entre-autre).

Durant le quart d'heure nécessaire pour assimiler tout ceci, Jean-Pierre BONDU nous a

montré un système original de présentation du HP71: dans une luxueuse valise prennent place une caméra vidéo, et le HP71. Et l'Assemblée peut alors suivre les manipulations de l'opérateur sur un écran de télévision (couleur...).

En fin d'après-midi, Jean-Jacques MOREAU nous a révélé les particularités du HP71, et de sa structure. Jean-Jacques connaît admirablement le HP71, et cela se traduit par des articles de qualité... (REPEAT).

Il était 17h environ, et j'ai préféré ne pas faire l'exposé que j'avais préparé sur le fonctionnement de l'HPIL. La salle commençait à ressentir la fatigue d'une journée d'interventions (les bidouilleurs ont aussi envie de s'échanger leurs bidouilles) !

L'après-midi s'est donc achevée en discussions particulières, en échanges de points de vue et de programmes.

Le dîner s'est pris au même endroit que le déjeuner.

La soirée s'est achevée très tard pour les plus mordus. A 2h du matin, nous n'étions plus que quatre: Jean-Claude BORNES, Jean-François GARNIER, Pierre et moi-même, penchés depuis plusieurs heures sur le logiciel de l'interface minitel. A 1h30, le HP75 était compatible. A 2h, le HP71 l'était aussi...

La journée de Dimanche se rapprochait plus des réunions mensuelles, ou des réunions de Lyon ou de Bretagne: les petites discussions personnelles ont agréablement rempli la matinée.

Après le repas, Eric GENGOUX nous a expliqué comment écrire aux Etats-Unis, pour commander du matériel (nous avons tous pris note). Puis, à la demande générale, il a exposé la structure et le fonctionnement du "POD" du HP75, ainsi que sa déception devant la politique d'HP envers le HP75. Eric, comme à l'accoutumé, avait remarquablement préparé son intervention.

Vers 17h30, après quelques parties de flipper

sur Compatible "IBM PC" avec Franck LEBASTARD et Alain FARGE, il était l'heure de tout ranger.

C'est là que nous avons constaté la disparition d'un lecteur d'EPROM et d'une HP41 bien remplie. C'est lamentable !

Remercions encore une fois Georges-Noël NICOLAS pour l'organisation impeccable, ainsi que Stefano TENDON (futur PPC-Italie) et Steen PETERSEN (PPC-Danemark) pour s'être déplacés.

Merci à tous et à bientôt

Michel MARTINET (SIG#2, P#12 et CHHU#617)

HEWLETT-PACKARD FRANCE : S. A. au capital de 250.000.000 de Francs régie par les articles 118 A 150 de la loi sur les sociétés commerciales - R.C.S. Corbeil-Essonnes : B 709605030.

SIÈGE SOCIAL : Parc d'activité du Bois Briard - 2, Avenue du Lac - 91040 EVRY CEDEX - Tél. (6) 077.83.83 - Telex 692315 F

Monsieur Philippe GUEZ  
CLUB PPC

56 rue JJ Rousseau

75001 PARIS

Evry, le 25 juin 1985

Monsieur,

Nous tenons à vous féliciter de l'organisation et du bon contenu technique du congrès PPC du 22 juin. C'est effectivement sur des bases professionnelles que HEWLETT-PACKARD peut vous accorder une aide efficace.

Nous avons le plaisir de vous confirmer nos premières actions concrètes :

1) insertion d'information sur la promotion HP 41 et information produits dans "Micro Revue" et "JPC Journal". Nous avons fait paraître dans looping les adresses des 2 clubs. Attention de ne pas diffuser des informations uniquement valables aux USA.

2) Insertion carte postale dans les boîtes (10,40,70)

- nous attendons, très vite, vos textes de présentation des clubs,

- nous vous confirmons que le format agréé par HP est : *14x9 cm*

- prévoir un tirage de 15 000 exemplaires à faire parvenir au Service Marketing PC à EVRY (après acceptation par HP du bon à tirer)

En contrepartie du coût d'insertion pris en charge par

l'usine, HEWLETT-PACKARD se réserve le droit d'avoir accès au fichier et de l'utiliser pour ses besoins commerciaux.

3) SICOB : STAND DES 4 TEMPS

Nous avons bien noté votre intérêt pour avoir un stand commun PPC aux 4 temps. A titre exceptionnel, nous vous confirmons notre accord pour vous réserver un emplacement et prendre en charge 75 % des frais. Il y a donc lieu de nous renvoyer au plus vite (Philippe GUEZ) votre dossier d'inscription avec un chèque de 5 000 FHT + 18,6 % TVA.

Conditions: Une présence continue du 1er au dernier jour est indispensable.

4) HEWLETT-PACKARD offrira gracieusement un module Translator Pac à chacun des 2 clubs, dans le but d'inciter les membres à développer sur le HP 71 B.

5) La négociation des remises chez les distributeurs agréés HP sur présentation de la carte de membre PPC sera menée par les PPC, indépendamment d'HP.

6) Il serait bon que vous preniez contact avec le club Micro de la SNCF : Microfer

Mr Jacques LEDOUX  
285.96.46

à qui nous avons déjà prêté un HP 71B.

Nous souhaitons renforcer nos liens avec vos clubs d'une façon amicale mais toujours professionnelle.

Nous vous prions d'agréer, Monsieur, l'expression de nos sentiments les meilleurs.

Eric CLEMENT  
CHEF DE PRODUITS-INFORMATIQUE PERSONNELLE

## NOUS EN AVONS

Sont en vente au Club:

- Cartes magnétiques pour HP41 au prix de 150FF + 3,10FF de port les 50 cartes;
- Cartes magnétiques pour HP71B au prix de 91FF + 3,10FF de port les 10 cartes;
- Cassettes digitales pour HP82161A: 100 FF pièce, plus 3,10 FF de port par cassette.
- Eproms (2716 & 2732) vierges au prix de 60FF + 3,10FF de port pièce. Pour les Eproms programmées, nous consulter;
- Module TIME (82182A) au prix de 400FF + 6,50FF de port. Livré avec manuel en français;
- Module QUADRAM (82170A) au prix de 400FF + 6,50FF de port;
- Module HPIL (82160A) au prix de 970FF + 6,50FF de port. Livré avec manuel en français;
- Pour les commandes des ZENROM, CCDROM et ROM PANAME nous envoyer 500FF minimum d'arrhes;
- VASM (listing en anglais des 3 premières pages de la 41) au prix de 170FF + 13,50FF de port;
- Le manuel de service de la 41 (en anglais) au prix de 50FF + 6,50FF de port;
- Le manuel technique du convertisseur (en anglais) au prix de 20FF + 3,20FF de port;
- HPIL Interface Specifications (en anglais) au prix de 60 FF + 9,50 FF de port
- The HPIL Integrated Circuit User's manual (en anglais): 40 FF + 6,50 FF de port
- Interface Kit Technical Guide (en anglais): 15 FF + 3,50 FF de port
- Manuel technique du Minitel R (Ref. 250 Alcatel): 40 FF + 6,50 FF de port
- Les anciens numéros au prix de:
  - No 1 à 3 au prix de 15FF + 3,20FF de port par numéro,
  - No 4 à 10 au prix de 20FF + 3,20FF de port par numéro,
  - No 11 à 17 au prix de 25FF + 6,50FF de port par numéro,
  - à partir du No 18 au prix de 35FF + 6,50FF de port par numéro.

Les ports sont les tarifs PTT actuels en non urgent.

## ACHATS GROUPES

Nous assurons des commandes groupées pour ceux qui désirent profiter de notre réduction "spécial Club", soit 25% par rapport au prix HP. Pour ceux qui le désirent, nous assurons l'envoi postal en urgent ou en non urgent. Ces achats concernent aussi bien des modules que des calculatrices et autres.

NOTA: ces deux rubriques "nous en avons" et "achats groupés", sont des services "Club" que nous rendons exclusivement aux membres effectivement inscrits au club et possédant leur carte de membre PPC-PARIS.

Chers amis,

Comme d'habitude, nous recherchons tous les membres désireux de nous aider à tenir le stand. Et que tous ceux qui ont du matériels à nous prêter lors de cette manifestation, seront les bienvenus.

Merci et à bientôt pour le prochain SICOB.

Philippe.

Evry, le 10 Juin 1985

Au cœur du centre commercial des 4 Temps, un nouvel espace est né :  
L'ESPACE HEWLETT-PACKARD 4 TEMPS.

Il ouvrira ses portes pendant toute la durée du SICOB (18-27 Septembre)  
pour accueillir plus de 70 sociétés Partenaires Hewlett-Packard.

Très attendue par la plupart d'entre vous, cette manifestation est un  
moyen de promouvoir vos produits, unique en son genre et très efficace  
à en juger par le succès de l'opération 4 TEMPS 84.

En effet, l'année dernière, 40 participants et 55 000 visiteurs ont largement  
contribué à la pleine réussite de cet événement.

Aujourd'hui, PERFORMANCES A L'APPUI, Hewlett-Packard a la volonté  
d'amplifier ce phénomène et de faire de l'Espace HEWLETT-PACKARD  
4 TEMPS, un lieu privilégié de l'Informatique où seront regroupées des  
sociétés dynamiques et compétentes par les solutions qu'elles proposent.

Nos moyens : publicité dans la Presse spécialisée, spots radio, affi-  
chage, fléchage sur le parvis, signalisation percutante dans le centre  
commercial, animation originale, catalogue...

**L'ESPACE HEWLETT-PACKARD 4 TEMPS VOUS ATTEND!**

Vous êtes déjà très nombreux à nous avoir sollicités pour cette manifes-  
tation. Le nombre de places étant limité, nous vous demandons de nous  
retourner très rapidement la demande de participation ci-jointe accom-  
pagnée d'un chèque d'acompte.

Compte tenu de la très forte demande, nous ne prendrons en compte  
que les dossiers dûment complétés (demande de participation et  
chèque).

**ET CECI AVANT LE 21 JUIN 1985.**

**IL NOUS SERA IMPOSSIBLE D'ACCEPTER LES DOSSIERS NOUS PAR-  
VENANT APRÈS CETTE DATE.**

Dans l'espoir de vous compter parmi les participants, nous vous prions  
d'agréer, Madame, Monsieur, l'expression de nos sincères salutations.



Véronique Tripard  
Relations avec les Partenaires  
Direction de la Communication



**HEWLETT  
PACKARD**

**PERFORMANCES A L'APPUI.**

1216185

## Sup. de Co.-Bordeaux : ordinateur de rigueur

Grande première européenne à l'école supérieure de commerce et d'administration des entreprises de Bordeaux : dès la prochaine rentrée, chaque étudiant de première année devra obligatoirement être muni d'un micro-ordinateur. L'appareil sélectionné est un Hewlett-Packard 110. Il ne pèse qu'un kilo et les élèves l'apporteront à l'école, le ramèneront chez eux, ni plus ni moins qu'un vulgaire stylo-bille ou un polycopié.

Cette banalisation de l'outil informatique n'est pas destinée à former des super-spécialistes capables de concevoir des machines intelligentes ou des programmes. Pour le directeur de Sup. de Co.-Bordeaux, Henri de Pitray, un homme qui a longtemps exercé des responsabilités dans le commerce international, il s'agit tout simplement d'ôter tout complexe étudiants vis-à-vis de l'informatique, de leur apprendre à utiliser l'appareil aussi naturellement que les cadres français, en 1985, ont recours au téléphone. L'objectif principal de l'école, créée en 1874, reste inchangé : former des cadres « généralistes, flexibles et adaptables ». Mais le fait d'avoir vécu en permanence avec un matériel informatique, d'avoir appris à s'en servir, devrait « libérer les étudiants vis-à-vis du phénomène informatique ». L'intérêt du micro-ordinateur sélectionné est évident : sur le H.P. 110, l'étudiant pourra préparer son rapport de stage sur le dispositif de traitement de texte intégré, puis brancher son appareil sur une imprimante ; en matière d'études de marché, il établira facilement ses fichiers et ses systèmes de tri ; les cours de mathématiques financières vont être totalement réformés : les calculs d'amortissement ou les tableaux de financement étant réalisés par la machine, l'enseignement pourra

être centré essentiellement sur le « pourquoi » et non plus sur le « comment »...

### Charge supplémentaire

Difficulté pratique de cette opération « informatique » : les étudiants doivent acheter leur propre machine, ce qui constitue une charge supplémentaire à côté des frais de scolarité classiques. La direction a réglé le problème en passant deux accords : l'un avec la société Hewlett-Packard qui a consenti une réduction de 60 % sur chaque appareil par rapport à son prix public, l'autre avec le Crédit du Nord qui proposera des prêts à des conditions très avantageuses aux étudiants. Ainsi le remboursement s'effectuera, de façon échelonnée, après entrée des intéressés dans la vie active.

Cette initiative de Sup. de Co.-Bordeaux pourrait bien faire école rapidement : les grands établissements français de commerce et de gestion ont toujours manifesté leur intérêt pour l'ordinateur, et la plupart sont dotées de centres informatiques importants. Jusqu'à présent, aucune n'avait osé sauter le pas en imposant à ses étudiants le port du micro-ordinateur personnel. Mais, dans ce secteur d'enseignement supérieur particulièrement dynamique, l'émulation n'est pas un vain mot...

Sup. de Co.-Bordeaux, en jouant les précurseurs, va encore améliorer son image de marque. L'école était déjà située dans les premiers rangs du hit-parade national : les 120 étudiants de chaque promotion sont le fruit d'une sélection très sévère, à partir de quelque 3 000 candidats.

Jacques MALHERBE.

-----  
URGENT-URGENT-URGENT-URGENT-URGENT-URGENT-URGENT-U

Grégoire JOUVE (T400)  
7, rue A. Leray  
35100 RENNES  
Tel: (99) 50 22 23

Messieurs les Présidents des Clubs  
PPC-T et PPC-PC,

A l'issue de la réunion PPC au CISP, avenue Maurice Ravel à Paris, le 23 juin 85, je n'ai pas pu retrouver mon HP41 et ses modules qui ont disparu entre 17h et 17h30.

Je vous serais reconnaissant de demander aux adhérents des Clubs de m'aider à les retrouver.

Cette machine est une HP41CV de numéro de série:  
2217 9 42949

mais avec une carte de CX et contenait un module Paname, un module Stat I et un module triple regroupant dans un boîtier X-Fonction un Zenrow et deux XMemory.

Avec mes remerciements,  
Grégoire JOUVE

ENT-URGENT-URGENT-URGENT-URGENT-URGENT-URGENT-URGE  
-----

R. PENNETIER  
Société I C E  
ZI de la Vigne aux Loups  
6 rue Bossuet  
91160 LONGJUMEAU

Monsieur Pierre DAVID

Monsieur,

Vous avez été très aimable d'essayer de me renseigner sur le problème rencontré sur la boucle HPIL entre HP75C et imprimante HP82162A, entrée graphique et codes-barres.

Ayant fini par découvrir ce qui "ne collait pas", je vous en fais part, à toutes fins utiles.

Il s'agit en fait d'une toute petite erreur d'impression du manuel de l'imprimante (p.13). Il suffit de faire précéder dans la séquence le "b" d'un signe "multiplier" (\*). Finalement (et ce serait tellement plus clair de l'exprimer ainsi) la séquence complète est la suivante:

```
PRINT CHR$(27)&"*bnng"&CHR$()....
```

Suivent les codes définis à la page 9 du manuel (0 à 127).  
nnn définit le nombre de codes qui vont suivre.

Dans la pratique, on peut remplacer l'ensemble des codes par une variable (A\$ par exemple), et la séquence devient:  
PRINT CHR\$(27)&"\*bnng"&A\$  
Au préalable, on écrira:  
A\$=CHR\$( )&CHR\$( )...  
A\$ peut être dimensionné (1 octet par code).

A noter que le livre de Janick TAILLANDIER "Autour de la boucle" n'est pas plus explicite à ce sujet et recopie fidèlement la faute.

Pour les codes-barres, on peut effectivement les obtenir en 8 bits en écrivant par exemple:  
PRINT CHR\$(27)&CHR\$(124)&CHR\$(210)&CHR\$(N)&"ABC..."  
N aura pour valeur (ce peut être une variable):  
127 + nb de caractères (16 au maximum).

Toutefois, l'édition de ces codes-barres semble assez peu intéressante (grosse consommation de papier: une seule ligne de code sur la largeur de la bande).

Merci encore de votre obligeance, et recevez mes sincères salutations.

R. PENNETIER

CHOC EN RETOUR

Je voudrais revenir sur l'article qui a été publié en mon nom dans le JPC No 22 en page 10: "La rapidité des Basic's".

Cet article avait été écrit en catastrophe à l'occasion du premier Sicob, où HP a présenté le HP71B. Vous aurez remarqué qu'il participe de l'enthousiasme qui régnait à l'époque dans notre Club. Mécontent des mauvaises conditions dans lesquelles j'avais réalisé ce travail, c'est avec une secrète satisfaction que j'avais appris un peu plus tard qu'on avait égaré le manuscrit de cet article.

Bien entendu, j'avais râlé comme un diable contre celui qui avait osé perdre mon travail... Aussi ai-je été pris au dépourvu lorsque j'ai vu mon article dans notre Journal.

Je ne renie pas ce qu'a été publié, cependant, dans un souci d'information, je tiens à contester certaines de mes affirmations. D'abord, outre le chauvinisme pro-HP qui suinte désagréablement d'entre les lignes, le lecteur attentif aura noté un vice de fond dans cet article: il était clairement destiné au néophyte ignorant. Dans ce cadre là, il eu été judicieux que l'auteur (moi) prenne la peine d'adopter un point de vue plus élevé que celui du néophyte, pour permettre à ce dernier d'accéder à des informations qu'il ne peut se procurer lui-même.

Si je ne mets pas en cause la partie technique du travail, la réalisation, l'analyse et le commentaire des tests -et de leurs limites- la partie qui tente, plus ou moins honnêtement, d'exposer les principaux défauts des machines est truffée d'erreurs. En particulier, les commentaires que j'ai pu faire sur la facilité d'accès des différents matériels: il est évident, à la lecture de l'article, que je n'ai pas pris le temps de m'attarder sur les manuels d'utilisation, alors que tout utilisateur raisonnablement modeste commence par là. De plus, la qualité de la documentation joue un rôle essentiel dans l'efficacité d'un matériel. HP n'est pas toujours ce que l'on fait de mieux dans ce domaine. Ainsi, faute d'avoir lu le manuel, le fonctionnement du M10 m'a presque entièrement échappé. L'affirmation selon laquelle il faudrait faire un LIST complet pour remonter d'une ligne de programme sur les machines Casio est fautive: comme bien souvent, il suffit de faire LIST suivi d'un numéro de ligne... De plus, il paraît que le PB700 a une touche équivalente à la flèche ascendante du HP71B. Un utilisateur de X07 courroucé m'a signalé que celui-ci calcule sur beaucoup plus de six chiffres.

Enfin, si le HP71B m'avait paru si facile d'accès, c'est aussi grâce à l'assistance de personnes qui le connaissent déjà bien.

Pierre LANGLOIS (PPCPC # 87)

#### UN NOUVEAU MODULE POUR LA HP41

Le "Bulletin Board" de PPC (dernières nouvelles, diffusées par répondeur téléphonique aux USA), fait état de l'arrivée d'une nouvelle ROM de 12Ko pour HP41, qui serait un émulateur des machines de la "Série 10", soit l'ensemble des trois machines HP12 (financière), 15 (scientifique-mathématique) et 16 (multibases).

Ce module serait écrit pour partie en langage utilisateur, et le reste en microcode, et serait offert dans un premier temps à tout nouvel acquéreur d'une 41CV ou CX, à l'occasion de la campagne de promotion qui débiterait aux Etats-Unis le 1er Août prochain.

Cette information, que je tiens de Janick TAILLANDIER (qui a pu appeler le "Bulletin Board" dernièrement), semble étonnante à plusieurs égards, puisque l'apparition d'une telle ROM ferait concurrence à la fois à la Série 10, à certaines ROM de la 41 (Maths, Finance) et partiellement au 71 (fonctions multibases et module Maths, pour un prix très inférieur). Elle est donc à préciser et confirmer, bien que PPC ne lance généralement pas de telles informations à la légère.

Eric GENGOUX (P#108)

#### COURRIER DU COEUR

Michel MARTINET (P#12, SIG#2, CHHU#617)  
83 rue de Gergovie  
75012 PARIS  
Tél: (1) 545 92 80  
Vend:  
HP75C + 8Ko + Visicalc + Text formateur + nbreux logiciels: 6000 FF.

Jean-Jacques DHENIN  
35 rue Boileau  
92120 MONTRouGE  
Vend:  
2 (deux) modules "Translator-Pac" pour HP71B.  
Prix neuf:1830 FF, soldés à 1400 FF pièce.

Eric GENGOUX (P108)  
rue de Furstenberg,  
75006 PARIS  
Tél. D: (1) 633 65 79, B: (1) 285 68 23  
Vends 2 unités de cassette à 2600FF (pièce) date

d'achat fin 83.

Etienne POUPEE  
8, rue Maître Jacques  
92100 BOULOGNE

Vends HPIL (pour 71) 1200FF à débattre. 1 module mémoire (pour 71) 600FF. 2 modules HPIL (pour 41) 500FF pièce. 1 Tandy modèle 100 2500FF. D'autre part, étudie tout soft et interface à la demande.

Philippe DAVASE  
7, rue Ho Chi Minh  
78500 SARTROUVILLE

Tél. D: (3) 913 51 67, B: 757 31 40 (poste 351) .  
Vends du matériel pour HP41. Imprimante 24 colonnes HPIL (prix neuf 5628FF), vendu à 2814FF. Imprimante GP100-HPIL (prix neuve 6000FF) vendue 3000FF. 1 module d'extension fonctions + 2 modules d'extension mémoire groupés en un seul module (prix neuf 2790FF) vendu 1395FF. Module PPCROM (prix neuf 1400FF) vendu 700FF. Module mathématique (prix neuf 435FF) vendu 218FF. Module statistique (prix neuf 435FF) vendu 218FF. Module Structural Analysis (prix neuf 560FF) vendu 280FF. Module Surveying -topographie- (prix neuf 435FF) vendu 218FF. Module Home Management (prix neuf 435FF) vendu 218FF. Boite de 6 rouleaux papier (prix neuve 145FF) vendue 73FF. Lecteur optique pour HP41 (prix neuf 1560FF) vendu 780FF. Port extender 8 ports (prix neuf 980FF) vendu 490FF. Soit totaux prix neuf 20803FF, et totaux prix d'occasion 10402FF. Soit un taux de vente de 50% .

René JACQUET  
2, villa Boissière  
75116 PARIS  
Tél: (1) 727 50 29 après 20 Heures  
Vend HP41CV + HPIL + HP82162A + HP82161A + Module time + Module XF  
Prix neuf: 23000 Frs. Vendu 8500 Frs.

M. MAUGUERET  
Tél: (1) 594 63 08  
HP41CV prix neuf 2150 FF vendu 1100 FF  
XFonctions: prix neuf 950 FF, vendu 400 FF  
Mod. Time: prix neuf 850 FF, vendu 400 FF  
Jeu de batteries suppl.: prix neuf 395 FF, vendues 200 FF.

HP75C + 8 Ko: prix neuf 8250 FF, vendu 4500 FF  
Mod. Visicalc: prix neuf 2300 FF, vendu 1150 FF  
Mod. Text Format.: prix neuf 1200 FF, vendu 600 FF  
Imp. HP82143A: prix neuf 3825 FF, vendue 1500 FF  
Imp. HP82162A: prix neuf 4825 FF, vendue 2000 FF  
Vidéo HP82163A: prix neuf 2700 FF, vendue 1250 FF  
Imp. HP82905B: prix neuf 7200 FF, vendue 3500 FF  
Moniteur Prince: prix neuf 1200 FF, vendu 500 FF

Frédéric BOCCARA  
8 rue Rabelais  
63100 CLERMONT-FERRAND  
Tél: (73) 25 68 08  
Recherche du matériel d'occasion:  
Accus, chargeur et XFonctions.

José Ignacio DE LA PENA  
66 Bis rue Sébastien Mercier  
75015 PARIS  
Tél: (1) 579 41 94  
Vend: lecteur de cartes pour HP71B: 1500 F



Chers amis,

Nous sommes heureux de souhaiter le bonjour à tous les adhérents de PPC-Paris. A l'invitation de votre Président, nous serons amenés à vous présenter régulièrement dans ces colonnes les activités de PPC-Toulouse.

Comme vous le savez peut-être, PPC-T est "l'autre" club français consacré aux portables HP. Au fil des années les deux clubs ont affirmé leur personnalité, au point que leurs publications -JPC pour PPC-Paris et MICRO-REVUE pour PPC-Toulouse- sont maintenant beaucoup plus complémentaires que concurrentes, ce qui est une très bonne chose.

Pour cette première présentation, disons en simplifiant que PPC-T s'intéresse aux calculatrices Hewlett-Packard série 10 et 40, au HP-75, qui dispose d'une rubrique régulière dans nos colonnes, et au HP-71. Il s'intéresse également au langage FORTH, et cela quel que soit l'ordinateur Hôte. Ainsi dans le No 6 (Mai-Juin, parution tous les deux mois) de MICRO-REVUE, on trouve 20 pages de programmes HP-41, 35 pages d'applications HP-41, dont une description d'un module MATH en préparation, 10 pages consacrées au HP-71, 18 pages consacrées au HP-75, un programme pour HP-16, 18 pages consacrées au FORTH dont 12 pages pour débutants, et 16 pages de magazine (les journaux, petites annonces, sur le marché,...).

Nous avons également un service librairie à prix réduits, une petite coopérative, et un service photocopie qui diffuse à peu près tout ce qui est disponible (et de copie autorisée) pour HP ou FORTH. Nous venons d'inscrire à notre catalogue les "IDS" du module HP-IL du 71. Ce document -522 pages, qui ne sera pas diffusé par Hewlett-Packard- est la description abondamment commentée du programme en assembleur du module HP-IL. Ce document est déconseillé aux débutants et tout acheteur s'engage à ne pas interroger HP sur son contenu. Il est disponible pour nos adhérents au prix de 250F franco.

Nous avons également tous les listings assembleur qui permettent d'installer le FORTH sur votre micro-ordinateur (assembleur et petite connaissance du langage machine nécessaire) prix 50F.

Par contre nous ne diffusons plus le module Zengrange, à la suite de difficultés innombrables avec le fournisseur, ni le module CCD.

Nous invitons tous ceux d'entre vous qui sont intéressés à nous demander notre catalogue/descriptif du club qui vous sera envoyé avec la liste de nos quelques 450 adhérents actifs en France et à l'étranger.

Salut à tous

Le bureau de PPC-T

## 7<sup>e</sup> Championnat International de programmes d'Othello-Reversi

### 7th Othello-Reversi Programs World Championship

Organisé par

**L'ORDINATEUR  
L'INDIVIDUEL**

Paris, le 20 juin 1985

Chers amis,

Vous savez, sans doute, que le prochain championnat de programmes d'OTHELLO-REVERSI se déroulera au SICOB, le 21 septembre prochain pour la catégorie ordinateurs de table, et les 21 et 22 septembre pour la catégorie ordinateurs de poche.

Nous serions très heureux de compter certains de vos adhérents parmi les participants et souhaitons recevoir leur inscription le plus rapidement possible afin de pouvoir organiser ce tournoi encore mieux que les années précédentes.

A cet effet, vous trouverez ci-joint le texte des règlements pour les deux catégories, et un bulletin d'inscription que vous pouvez évidemment photocopier.

Dès réception des bulletins d'inscription, nous enverrons aux participants les informations pratiques concernant le tournoi.

Nous vous attendons nombreux,

Bien cordialement,

*Brigitte Millé*

Brigitte MILLÉ

Promotion

## HP41 R P N

Pierre COLIGNON	Transformations Etoile-Triangle
Pierre ANTOINE	Taille du catalogue
Pierre LANGLOIS	Recherche de dates
Pierre ANTOINE	Conversions Minuscule / Majuscule
Philippe TENAND	Des nouveautés pour HP41
Christian MORLOT	Conversion de fichier texte en minuscule
Christian MORLOT	Balistique
Christian MORLOT	Réglage du module TIME

## HP41 ASSEMBLEUR

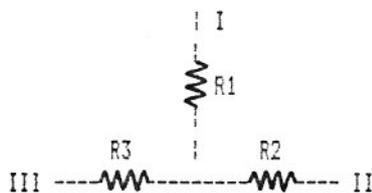
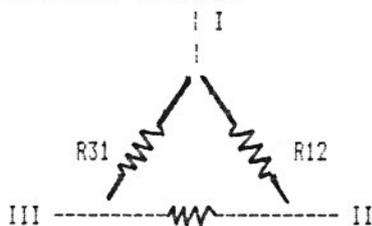
Pierre ANTOINE	Inversion du registre "C"
----------------	---------------------------

## TRANSFORMATIONS ETOILE-TRIANGLE

Dans le cadre de différents calculs d'électrocinétique, il est parfois fort utile de simplifier le réseau étudié. On transforme alors en une résistance équivalente l'ensemble des résistances qui constituaient le réseau. Cette opération peut se révéler difficile si on ne dispose que des formules d'association de résistances en série et en parallèle, et qu'on ne connaît pas le théorème de Kennelly...

### METHODE UTILISEE

Le théorème de Kennelly et les formules qui en découlent, permettent de transformer une maille triangulaire de 3 résistances d'un réseau, en une étoile de trois résistances sans que les caractéristiques du réseau soient modifiées (en soient affectées). Rassurez-vous, je n'ai pas l'intention de vous faire un cours (j'en serais d'ailleurs bien incapable), néanmoins et pour plus de clarté, il me semble nécessaire de brièvement vous rappeler les formules de transformations utilisées.



Explicitons tout d'abord les notations. Les résistances désignées par un seul indice appartiennent à l'étoile et celles désignées par un indice double au triangle. Plus précisément, on notera R1 (resp. R2, R3) la résistance placée entre les noeuds 0 et 1 (resp. 0 et 2, 0 et 3).

Dans le cas du triangle, on notera R12 (resp. R23, R31) la résistance placée entre les noeuds 1 et 2 (resp. 2 et 3, 3 et 1). L'erreur d'identification est donc impossible.

Donnons maintenant les formules de passage du triangle à l'étoile:

$$T = R12 + R23 + R31$$

$$R1 = R12 * R31 / T$$

$$R2 = R12 * R23 / E$$

$$R3 = R23 * R31 / E$$

Donnons enfin les formules de passages de l'étoile au triangle:

$$E = R1 * R2 + R2 * R3 + R3 * R1$$

$$R12 = E / R3$$

$$R23 = E / R1$$

$$R31 = E / R2$$

### PROGRAMME

Le programme que je vous propose est donc très simple.

La première partie du programme qui regroupe les routines de calcul fractionnaire, est très largement inspirée du livre de Jean-Jacques Dhémin paru aux éditions du PSI: "Programmer HP41". Tous les calculs s'effectuent dans la pile sans qu'aucune mémoire soit utilisée.

On trouve ensuite les routines de transformation Triangle-Etoile (T-E) et Etoile-Triangle (E-T).

Ces deux routines font appel à des sous-programmes communs d'initialisation et d'affichage.

Viennent enfin deux routines qui permettent de déterminer la résistance équivalente à deux résistances disposées en série pour "RS" ou en parallèle pour "RP".

### MODE D'EMPLOI

Supposons que vous vouliez faire une transformation Triangle-Etoile...

Vous lancez alors le programme en tapant: [XEQ] "T-E".

La machine vous demande alors d'entrer la valeur de la résistance R12 du triangle; elle affiche donc:

"TR12 = "

En supposant toujours que la valeur de cette

résistance vaille 0,5 ohms, vous devez répondre: 1 [ENTER] 2 ou encore 0,5 [ENTER] 1, puis appuyer sur [R/S].

Et ainsi de suite, jusqu'à ce que les valeurs des trois résistances aient été introduites.

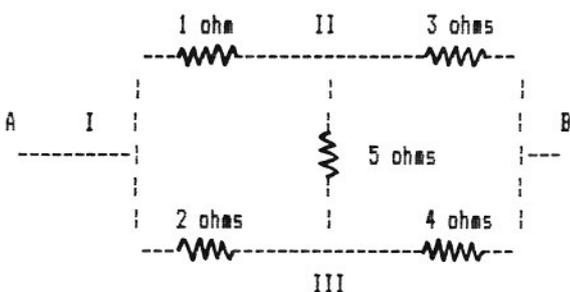
Le programme affiche alors les résultats sous la forme: "ER1=xxx/xxx" et ainsi de suite...

Je vous laisse découvrir le fonctionnement des autres programmes (E-T, RS, RP) fonctionnent d'ailleurs sur le même modèle.

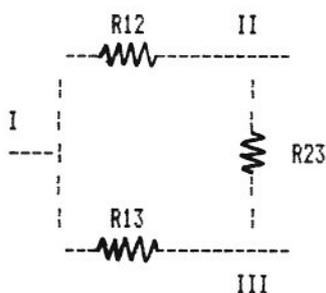
Bien entendu, il est possible d'effectuer les 4 opérations sur les fractions (F+,F-,F\*,F/) directement au clavier. Il convient alors d'entrer les deux fractions avant d'appeler ces fonctions.

### EXEMPLE

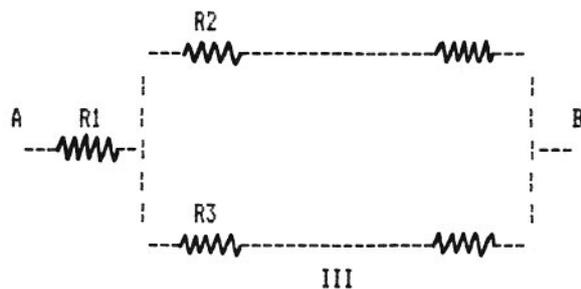
Soit le réseau suivant:



On a donc:



Ceci se simplifie pour donner:



Pour simplifier un tel réseau, on utilisera plutôt une transformation T-E. (Il est également possible d'utiliser une transformation E-T, mais cela conduit à des calculs plus complexes (je vous laisse vous amuser...). On tapera donc : [XEQ] "T-E"

"TR12 = " -> 1 [ENTER] 1 [R/S]

"TR23 = " -> 5 [ENTER] 1 [R/S]

"TR31 = " -> 2 [ENTER] 1 [R/S]

Les résultats donnés par le programme seront

"ER1 = 1/4"

"ER2 = 5/8"

"ER3 = 5/4"

A vous maintenant de trouver la résistance équivalente entre A et B.

Bon courage et bonne vacances...

Pierre COLIGNON

```
01#LBL "RF"
02 STO Z
03 X<>Y
04 STO T
05#LBL 01
06 MOD
07 LASTX
08 X<>Y
09 X#0?
10 GTO 01
11 +
12 ST/ Z
13 ST/ Y
14 RDN
15 RTN
16#LBL "F/"
17 X<>Y
```

18#LBL "F#"	:	68 RCL 03
19 ST# Z	:	69 RCL 06
20 RDN	:	70 RCL 05
21 ST# Z	:	71 SF 00
22 RDN	:	72#LBL 51
23 GTO "RF"	:	73 XEQ "F#"
24#LBL "F-"	:	74 RCL 07
25 CHS	:	75 RCL 08
26#LBL "F+"	:	76 XEQ "F/"
27 ST# T	:	77 FC?C 00
28 X<> Z	:	78 RTN
29 ST# Z	:	79 3
30 #	:	80 XEQ 54
31 RCL Z	:	81 RTN
32 +	:	82 GTO "T-E"
33 X<>Y	:	83#LBL "E-T"
34 GTO "RF"	:	84 "ER"
35#LBL "T-E"	:	85 ASTD 08
36 "TR"	:	86 1
37 ASTD 08	:	87 XEQ 50
38 1,1	:	88 "TR"
39 XEQ 50	:	89 ASTD 00
40 RCL 06	:	90 RCL 04
41 "ER"	:	91 RCL 03
42 ASTD 00	:	92 RCL 02
43 RCL 05	:	93 RCL 01
44 RCL 04	:	94 XEQ "F+"
45 RCL 03	:	95 RCL 06
46 XEQ "F+"	:	96 RCL 05
47 RCL 02	:	97 XEQ "F#"
48 RCL 01	:	98 STO 07
49 XEQ "F+"	:	99 X<>Y
50 STO 08	:	100 STO 08
51 X<>Y	:	101 RCL 04
52 STO 07	:	102 RCL 03
53 RCL 02	:	103 RCL 02
54 RCL 01	:	104 RCL 01
55 RCL 06	:	105 XEQ "F#"
56 RCL 05	:	106 RCL 08
57 XEQ 51	:	107 RCL 07
58 1	:	108 XEQ "F+"
59 XEQ 54	:	109 STO 07
60 RCL 02	:	110 X<>Y
61 RCL 01	:	111 STO 08
62 RCL 04	:	112 X<>Y
63 RCL 03	:	113 RCL 06
64 XEQ 51	:	114 RCL 05
65 2	:	115 XEQ "F/"
66 XEQ 54	:	116 12
67 RCL 04	:	117 XEQ 54

118 RCL 08		168 4
119 RCL 07		169 MOD
120 RCL 02		170 X=0?
121 RCL 01		171 1
122 XEQ "F/"		172 ARCL X
123 23		173 RTN
124 XEQ 54		174*LBL "RP"
125 RCL 08		175 " R1 ="
126 RCL 04		176 PROMPT
127 RCL 03		177 1
128 RCL 05		178 1
129 XEQ "F/"		179 RDN
130 31		180 RDN
131*LBL 54		181 XEQ "F/"
132 FIX 0		182 STO 01
133 " "		183 X<>Y
134 ARCL 00		184 STO 02
135 ARCL X		185 1
136 "┌ ="		186 1
137 ARCL Z		187 " R2 ="
138 "┌-/"		188 PROMPT
139 ARCL Y		189 XEQ "F/"
140 FIX 4		190 RCL 02
141 PROMPT		191 RCL 01
142 RTN		192 XEQ "F+"
143 GTO "E-T"		193 1
144*LBL 50		194 1
145 STO 07		195 RDN
146 1,006		196 RDN
147 STO 00		197 XEQ "F/"
148 FIX 0		198 " R = "
149*LBL 53		199 FIX 0
150 " "		200 ARCL Y
151 ARCL 08		201 "┌-/"
152 ARCL 07		202 ARCL X
153 ISG 07		203 FIX 4
154 XEQ 70		204 PROMPT
155 "┌ ="		205 GTO "RP"
156 PROMPT		206*LBL "RS"
157 STO IND 00		207 " R1 ="
158 ISG 00		208 PROMPT
159 X<>Y		209 " R2 ="
160 STO IND 00		210 PROMPT
161 ISG 00		211 XEQ "F+"
162 GTO 53		212 FIX 0
163 FIX 4		213 " R = "
164 RTN		214 ARCL Y
165*LBL 70		215 "┌-/"
166 RCL 07		216 ARCL X
167 INT		217 FIX 4

218 PROMPT  
219 GTO "RS"  
220 END

Pierre ANTOINE (P61)

TAILLE DU CATALOGUE

Le programme "MEDSIZ?" donne la taille du catalogue de la cassette.  
La valeur est donnée par l'octet 19 (voir un ancien numéro de JPC...)

01#LBL "MEDSIZ?"  
02 CF 15  
03 CF 16  
04 16  
05 FINDAID  
06 SELECT  
07 4  
08 DEVL  
09 .  
10 OUTXB  
11 OUTXB  
12 2  
13 DEVT  
14 3  
15 DEVL  
16 19  
17 OUTXB  
18 .  
19 DEVT  
20 INXB  
21 8  
22 \*  
23 E  
24 -  
25 FIX 0  
26 CF 29  
27 PRBUF  
28 ACX  
29 " FICHIERS"  
30 "← POSSIBLES."  
31 PRA  
32 END

RECHERCHE DE DATES

Même si, comme vous, j'estime que le module Time de la HP41 n'a pas encore trouvé de concurrent sur les plans de la puissance et de la facilité d'utilisation, je trouve qu'il aurait gagné à avoir quelques fonctions supplémentaires.

Entre autres, la fonction que je vous présente aujourd'hui qui permet de répondre aux questions du type: "Quelle est la date du mardi précédent le 1er janvier 1986 ?"  
Avec SEEKD, il suffit d'introduire 1,011986 [ENTER^], puis -2, et de lancer la routine. Vous avez compris que le premier chiffre correspond à la date de départ, et le deuxième au jour de la semaine recherché, le signe de ce dernier indiquant au programme si la date est à chercher avant ou après la date de départ.

On m'a déjà objecté qu'il est plus simple de faire 1,011986 [DOW], et d'utiliser ensuite une gymnastique mentale simple. Et effectivement, si vous êtes rompus à ce genre d'exercice, vous n'avez pas besoin de cette routine, mais je pense que votre machine avait besoin de l'apprendre. A moins que vous ne préféreriez voir votre programme d'agenda s'arrêter toutes les 5 minutes pour vous demander par exemple: "Quel est le premier lundi du mois de décembre 1985 svp ?"...

Remarquez que la routine détruit le registre T et l'état du drapeau 0, le registre Z descend intact en Y, les registres X et Y étant détruits.

Maintenant, sauriez-vous me dire la date du deuxième samedi du mois de septembre 4320 ?

Pierre LANGLOIS (PPC#87)

01#LBL "SEEKD"  
02 CF 00  
03 X<0?  
04 SF 00  
05 RCL Y  
06 DOW  
07 FC? 00  
08 CHS  
09 +  
10 7  
11 MOD  
12 FS?C 00  
13 CHS  
14 DATE+  
15 END

14 GTD 00  
15 CLX  
16 91  
17 X<Y?  
18 GTD 00  
19 32  
20 ST+ Z  
21 RDN  
22#LBL 00  
23 RDN  
24 XTOA  
25 X<>Y  
26 DSE X  
27 GTD 01  
28 END

Pierre ANTOINE (#61)

-----  
Programmes "MIN" et "MAJMIN"

MIN(us) transforme l'ALPHA en minuscules. Ceci ne concerne bien sûr que les lettres, mais les autres signes sont également traités (% reste %, mais A devient a, et a reste a).

MAJMIN, fait la même chose, mais il ne touche pas à la première lettre de l'ALPHA. (PPC devient Ppc), et (%\*A devient %\*A).

01#LBL "MAJMIN"  
02 E  
03 AROT  
04 ALENG  
05 -  
06 CHS  
07 GTD 01  
08#LBL "MIN"  
09 ALENG  
10#LBL 01  
11 ATOX  
12 65  
13 X>Y?

-----  
DES NOUVEAUTES POUR HP41

Les amateurs de HP41 peuvent trouver au sein de certains clubs et distributeurs spécialisés un certain nombre de périphériques nouveaux venant de petits fabricants européens nombreux et variés. Voici une sélection des productions les plus intéressantes. Chaque modèle existant sous des formes diverses et à des prix variant du simple au double suivant ses possibilités. Ce sont:

Des tables traçantes et des imprimantes interfacées HPIL (seikosha, canon, etc... à environ 4600FF).

Des modules ROM commandés par les clubs à Hewlett-Packard (PPCROM, CCDROM, ZENROM, PANAME... à environ 1000FF). Des lecteurs d'eprom (de 700 à 2500FF ou plus), dont le très beau HHP de 32k (à 5000FF) sous la forme d'un faux lecteur de cartes magnétique.

Du matériel pour programmer en microcode (ce qu'on appelle un MLDL, de 2500 à 5000FF), des logiciels en eproms programmés en langage normal

ou microcode (ce qui multiplie la vitesse d'exécution par 349.5), de 300 à 450FF, et qui nécessitent un lecteur d'eproms. Ces logiciels touchent des domaines très variés (math, gestion, routines, programmation synthétique, etc...).

On trouve aussi des ports d'extension (de 6 à 8 sorties suivant le modèle, dont le français SICAPE à 1250FF).

Un synthétiseur de voix (HHP à 5000FF).

Un clavier de type machine à écrire (PBG à plus de 5000FF).

Un module spécial avec un interrupteur et une tige flexible se terminant par une petite lampe alimentée par pile, vous permettant d'utiliser votre HP41 la nuit (ANE 300FF).

Des étuis de taille diverse pour la HP41 et une extension, des catalogues réunissant sur chaque page plastifiée de quoi réunir une feuille de programme et une ou plusieurs cartes magnétiques dans leurs compartiments (60FF).

Tous ces matériels sont distribués en France seulement par des clubs ou quelques distributeurs spécialisés, et un seul ou deux modèles (extensions de ports et imprimantes) se vendent dans certains magasins.

Au niveau des Clubs, moyennant forte finance, on vous modifie HP41: accélération de lecture, modification de la batterie, led de vérification de charge, insertion de modules (jusqu'à 7), couplage de modules, interrupteur de toute commande (ALL RESET), insertion sur les côtés gauche et droit de la boucle HPIL.

Beaucoup de livres sur la HP41, en langue étrangère à des prix élevés (le dollar de papier imprimé importé revenant parfois à 15FF).

Ouverts à tous, les Clubs font toutefois des remises à leurs membres. Les prix et les modèles étant très variés, se renseigner aux adresses ci-dessous:

- SICAPE, 122 avenue de la république, 94120 FONTENAY SOUS BOIS, Tél: (1) 887 38 16.
- PPC-FRANCE, 56 rue Jean-Jacques Rousseau, 75001 Tél: (1) 233 87 73.
- EDITIONS DU CAGIRE, 77 rue Cagire, 31100 TOULOUSE, Tél.: 16 (61) 44 30 06.

Philippe TENAND

-----  
Programme "MAJ-MIN"

Ce programme vous permet de convertir un fichier texte tapé en majuscules dans l'éditeur en minuscules.

RESTRICTIONS: Pas de majuscules en début de ligne. Cependant, il tient compte des points et place une majuscule derrière. Ce programme est assez long d'exécution et demande à être optimisé tant au point de vue vitesse qu'au niveau possibilités. toutes les suggestions seront les bienvenues.

Le mode d'emploi est assez simple. Lancer le programme et répondre aux questions. Si vous désirez partir d'un caractère donné, placez le pointeur à l'endroit désiré (SEEKPT), puis XEQ 00.

- 01#LBL "MAJ-MIN"
- 02 "FICHIER"
- 03 AON
- 04 PROMPT
- 05 AOFF
- 06 ,
- 07 SEEKPTA
- 08#LBL 00
- 09 RCLPT
- 10 SF 25
- 11 GETREC
- 12 FC?C 25
- 13 GTD 03
- 14 SF 25
- 15 SEEKPT
- 16 FC?C 25
- 17 XEQ 01
- 18 ATOX
- 19 CLA
- 20 XTOA
- 21 65
- 22 X>Y?
- 23 GTD 02

24 CLX	:	
25 91	:	Ce programme calcule les différents paramètres
26 X<Y?	:	dans un problème de balistique.
27 GTD 02	:	
28 RDN	:	MODE D'EMPLOI: Lancer le programme, introduire
29 32	:	les données, passer l'inconnue par R/S puis le
30 +	:	programme fait le reste.
31 CLA	:	
32 XTOA	:	01#LBL "BLST".
33 1	:	02 DEG
34 DELCHR	:	03 CF 28
35 INSCR	:	04 SF 29
36 GTD 00	:	05 CF 00
37#LBL 01	:	06 CF 01
38 RCLPT	:	07 CF 02
39 INT	:	08 CF 03
40 SEEKPT	:	09 CF 04
41 GETREC	:	10 CF 22
42 SEEKPT	:	11 "Y=?"
43 RTN	:	12 PROMPT
44#LBL 02	:	13 FS?C 22
45 X<>Y	:	14 SF 00
46 46	:	15 STD 01
47 X=Y?	:	16 "X=?"
48 ,002	:	17 PROMPT
49 ,002	:	18 FS?C 22
50 X#Y?	:	19 SF 01
51 ,001	:	20 STD 02
52 RCLPT	:	21 "G=?"
53 +	:	22 PROMPT
54 SF 25	:	23 FS?C 22
55 SEEKPT	:	24 SF 02
56 FS?C 25	:	25 STD 03
57 GTD 00	:	26 "V0=?"
58 RCLPT	:	27 PROMPT
59 INT	:	28 FS?C 22
60 1	:	29 SF 03
61 +	:	30 STD 04
62 SF 25	:	31 "L =?"
63 SEEKPT	:	32 PROMPT
64 FS?C 25	:	33 FS?C 22
65 GTD 00	:	34 SF 04
66#LBL 03	:	35 STD 05
67 BEEP	:	36 FC? 01
68 END	:	37 GTD 02
	:	38 FC? 02
	:	39 GTD 03
Programme "BLST"	:	40 FC? 03
	:	41 GTD 04
	:	42 FC? 04

43 GTO 05	:	93 /
44#LBL 01	:	94 XEQ 10
45 RCL 03	:	95 *
46 XEQ 08	:	96 "V0="
47 /	:	97 ARCL X
48 RCL 02	:	98 PROMPT
49 X^2	:	99#LBL 05
50 *	:	100 RCL 02
51 XEQ 09	:	101 X^2
52 +	:	102 RCL 03
53 "Y="	:	103 *
54 ARCL X	:	104 RCL 04
55 PROMPT	:	105 X^2
56#LBL 02	:	106 -2
57 RCL 03	:	107 *
58 XEQ 08	:	108 /
59 /	:	109 RCL 02
60 RCL 05	:	110 RCL Y
61 TAN	:	111 RCL 01
62 RCL 01	:	112 -
63 CHS	:	113 XEQ 06
64 XEQ 06	:	114 ATAN
65 "X1="	:	115 X<>Y
66 ARCL X	:	116 ATAN
67 PROMPT	:	117 X<0?
68 X=Y?	:	118 X<>Y
69 RTN	:	119 "% ="
70 "X2="	:	120 ARCL X
71 ARCL Y	:	121 PROMPT
72 PROMPT	:	122#LBL 10
73#LBL 03	:	123 RCL 01
74 XEQ 10	:	124 XEQ 09
75 XEQ 08	:	125 -
76 RCL 02	:	126 RTN
77 X^2	:	127#LBL 08
78 /	:	128 RCL 04
79 *	:	129 X^2
80 "G="	:	130 RCL 05
81 ARCL X	:	131 COS
82 PROMPT	:	132 X^2
83#LBL 04	:	133 *
84 RCL 05	:	134 -2
85 COS	:	135 *
86 X^2	:	136 RTN
87 -2	:	137#LBL 09
88 *	:	138 RCL 05
89 RCL 03	:	139 TAN
90 RCL 02	:	140 RCL 02
91 X^2	:	141 *
92 *	:	142 RTN

```

143#LBL06      : du mode chrono. Introduire la modification comme
144 X<>Y      : pour "T+X", etc...
145 X^2        :
146 -4         : Le réglage effectué, appuyer sur "C", et voilà.
147 R^         :
148 ST# Y      : 01#LBL "TI-COR"
149 R^         : 02 SF 27
150 ST# Z      : 03 RUNSW
151 R^         : 04 SETSW
152 ST+ T      : 05 SW
153 X<> L      : 06 STOP
154 CHS        : 07#LBL00
155 STO Y      : 08 ENTER^
156 R^         : 09 ENTER^
157 X<0?       : 10 RCLSW
158 GTO 07     : 11 HMS+
159 SQRT       : 12 4 E-5
160 +          : 13 HMS+
161 X<>Y       : 14 SETSW
162 LASTX      : 15 SW
163 -          : 16 RDN
164 LASTX      : 17 STOP
165 X<> T      : 18 GTO 00
166 ST+ X      : 19#LBL C
167 ST/ Z      : 20 RCLAF
168 ST/ Y      : 21 RCLSW
169 RDN        : 22 CORRECT
170 RTN        : 23 X<>Y
171#LBL 07     : 24 RCLAF
172 "PAS DE SOL" : 25 END
173 PROMPT    :
174 END        :

```

Programme "TI-COR"

Bonnes Vacances à tous,  
Christian MORLOT

Ne voulant pas entrer dans le débat de l'utilité de l'heure d'été, je vous propose simplement un programme vous permettant de régler avec précision l'horloge de votre machine. Il permet en outre un délai plus court que la normale pour déterminer le facteur d'exactitude avec une même précision puisque l'heure juste est préalablement ajustée dans le chronomètre.

Introduire l'heure, lancer le programme. Si cette heure n'est pas tout à fait juste, sortir

-----

#### INVERSION DU REGISTRE C

But: inverser le registre "C" du microprocesseur de notre chère HP41, pour réaliser une inversion vidéo, par exemple.

Utilise:  
C et A(2)

P ou Q (le pointeur actif)

Note: les octets de valeur supérieure à 128 ne peuvent être inversés.

AC EX W  
PT = 2  
LC 6  
AC EX W  
RCR 2 (Début de la boucle d'inversion)  
CST EX  
S7 = 1  
CST EX  
A=A-1 S  
GONC\* -5 (fin)  
RTN

Heureuse Programmation  
Pierre ANTOINE

TOI, ENVOYER CE  
MESSAGE AU CLUB  
"PPC - PARIS..."



## HP71 ASSEMBLEUR / FORTH

PPC-PARIS	Re-numérotation des Tokens
Jean-Jacques MOREAU	Une répétition à votre mesure
Michel MARTINET	MARGIN
Jean-Jacques DHENIN	MENU
Olivier ARBEY	Nombres premiers en assembleur
Pierre DAVID	Histoire d'impressionner
Jean-Jacques MOREAU	Gestion de touches en assembleur
Jean-Jacques DHENIN	INDATE
DAVID/MARTINET	Formattons, formattons !! (Suite)
Christian BACHELET	VLIST

## HP71 BASIC

Jean-Marie SIMON	Navigation
Philippe GUEZ	Utilisation de la table traçante X710
Jean-Pierre BONDU	A propos de DISP\$
Laurent ISTRIA	SUPREM
Jean-Claude FOURES	Tri d'une table
Laurent SERANO	Compte bancaire / Compte téléphonique
Xavier BILLE	Caractères Ethiopiens

## LE COIN DES LHEX

PPC-Paris	Récapitulatif optimisé et les Lhex du mois
-----------	--

LA RENUMEROTATION DES TOKENS

Dans la salle de rédaction de JPC, au milieu des téléscribes, des kilos de giga-octets sur disque-dur interfacé HPIL, un cri jaillit soudain: "Trop, c'est trop !"

En effet. La situation ne pouvait plus durer. L'anarchie régnait parmi les bits.

Depuis le numéro de Février 85 de JPC, chacun faisait ses Lex dans son coin, sans se préoccuper des autres, et surtout de leurs Lex. Il s'ensuivit donc une numérotation des tokens quelque peu erratique. Le procédé le plus fiable pour avoir le moins de chance d'entrer en conflit avec quelqu'un d'autre était de calculer: Token=IP(RND#255)+1

Si nous avons laissé faire, la situation serait devenue invivable.

Il fallait donc réagir vite. La solution que nous avons adoptée vous causera peut-être quelques tracas, mais songez aux désagréments que cela vous aurait causés si la décision avait été prise en décembre 1985, ou plus tard encore !

Désormais, tous les Lex qui paraîtront dans le journal seront numérotés d'office par nos soins. Tous les programmes BASIC qui nous parviendront devront, s'ils utilisent des Lex, être transmis avec les bons numéros de tokens.

Ceci entrainera peut-être quelques soucis, mais le jeu en vaut la chandelle. Désormais, il n'y aura plus de conflits, PPC-PARIS gèrera tout cela pour vous...

Assemblez bien !  
PPC-PARIS

TOKENS:

NOUVELLE  
APPELLATION:

ANCIENNE  
APPELLATION:

XFN	113001	ADBUF\$	LEX DESAL	XFN	092001
XFN	113002	ASC\$	.	XFN	092002
XFN	113003	ATH\$	.		
XFN	113004	HTA\$	.	XFN	092003
XFN	113005	RED\$	.	XFN	092005
XFN	113006	REPLACE\$	LEX REPLEX	XFN	092008
XFN	113007	FILE?	LEX FILELEX	XFN	092240
XWORD	113008	ATTN	LEX ATTNLEX	XWORD	092033
XWORD	113009	DISABLE	LEX DRIVELEX	XWORD	092040
XWORD	113010	ENABLE	.	XWORD	092041
XWORD	113011	FKEY	LEX FKEYLEX	XWORD	092034
XFN	113012	CONTRAST	LEX DESLEX	XFN	092020
XWORD	113013	INVERSE	.	XWORD	092021
XFN	113014	INV\$	.	XFN	092022
XFN	113015	PAINT	.	XFN	092023
XFN	113016	ENDUP\$	LEX ENDUPLEX	XFN	092230
XWORD	113017	ENDUP	.	XWORD	092231
XFN	113018	STARTUP\$	.	XFN	092232
XWORD	113019	CMD16	LEX CMD16LEX	XWORD	093130
XFN	113020	ARR	LEX COMBARR	XFN	093018
XFN	113021	COMB	.	XFN	093019
XFN	113022	HMS+	LEX HMSLEX	XFN	092060
XFN	113023	HMS-	.	XFN	092061
XFN	113024	HMS	.	XFN	092062
XFN	113025	HR	.	XFN	092063
XFN	113026	XERR	LEX XERRLEX	XFN	093128
XWORD	113027	STACK	LEX STKLEX	XWORD	092050
XWORD	113028	REPEAT	LEX REPEAT		
XWORD	113029	MARGIN	LEX MARGELEX		
XFN	113030	PRIM	LEX PRIMLEX		
XFN	113031	CURSDOWN	LEX DNKEYLEX		
XFN	113032	CURSKEYS	.		
XFN	113033	MENU	LEX MENULEX		



Appuyez maintenant sur une touche (Par ex.: A)  
et maintenez la enfoncée. Essayez aussi:

```
REPEAT 15
REPEAT -1
REPEAT 17
```

Le paramètre de la fonction REPEAT redéfinit le temps de répétition d'une touche. Le paramètre de la fonction REPEAT doit être réel et compris entre 0 et 15. Tout autre type d'argument provoque une erreur (ERR xxx: ...). L'échelle adoptée va donc de 0 à 15 (1 quartet).

Jean-Jacques MOREAU

#### NOTE DE LA REDACTION

IMPORTANT: Le Lex REPEAT prend le POLL pWTKY et doit donc toujours se trouver en dernière position dans votre système.

```
LEX 'REPEAT'
ID #71
MSG 0
POLL POLHND

ENTRY REPEAT
CHAR #D
KEY 'REPEAT'
TOKEN 28
```

```
EXPEXC EQU #0F186
SLEEP EQU #006C2
RNDAX EQU #136CB
IVAERR EQU #0E920
RESERV EQU #2F986
MPY EQU #0ECBB
CKSREQ EQU #00721
FIXDC EQU #05493
FIXP EQU #02A6E
POPBUF EQU #010EE
NXTSTM EQU #08A48
SFLAGS EQU #135FA
SFLAGC EQU #13601
SFLAG? EQU #1364C
```

```
: USRSTA EQU #01880
: BLDDSP EQU #01898
: SETTMO EQU #13158
: ALMSRV EQU #1257D
: RKRSTK EQU #014DD
: RSTK<R EQU #014A8
: DEBNCE EQU #00CF7
: KEYBUF EQU #2F444
: IDIVA EQU #0EC6E
: KCOLD EQU #2F46F
: TIMER1 EQU #2E3F8
: KEYPTR EQU #2F443
: FORSTK EQU #2F59E
: DELAYp EQU #02AC6
: fTMOUT EQU #3
: f1RPTD EQU #C5
: k#OFF EQU #63
: pWTKY EQU #1C
```

ENDTXT

```
*****
* Routine d'interception de poll
*****
POLHND
```

```
LC(2) =pWTKY
?B=C B S'agit-il du poll envoyé
* lorsque le 71 est en mode
* d'édition ?
GOYES KEYR Oui; il faut scruter le
* clavier
RTNSXM Non; le poll envoyé ne
* nous concerne pas
```

```
*****
* Routine KEYR
* Cette routine scrute le clavier: elle attend que
* l'on appuie sur une touche, puis place le code
* de la touche sur laquelle on a appuyé dans R0(B)
* KEYR boucle indéfiniment sur elle-même si l'on
* n'appuie sur aucune touche.
```

```
*****
KEYR
P= 4 Sauvegarde en RAM les 5
* derniers adresses placées
* dans la pile de
* GOSBVL =R<RSTK Retour (RSTK)
* GOSUB RPTKY L'action d'une touche
* doit-elle se répéter ?
* GOC DEFKY Oui; on retourne le code
* de cette touche a la
* routine qui a émis le pol
```



```

C=R0          Enfin on place le      |          GOYES RPTKCC
*            résultat dans C(5-0) sous |          GOSUB USRREQ
*            la forme:                 |          GOTO RPTK20
C=A   B       70000T                  |
RPTK10 GOSUB WRTTM1 La routine ci-dessous est | WRTTM1
*            commentée dans les IDS 3   |          DO=(5) =TIMER1
LC(2) =f1RPTD                          |          P= 5
GOSBVL =SFLAGS                          |          DATO=C 6
RPTK20 P= 8                              |          A=DATO 6
GOSBVL =DEBNCE                          |          ?A=C WP
D1=(5) (=KEYBUF)+2*14                  |          GOYES WRTTMX
A=0   A                                      |          DATO=C 6
A=DAT1 B                                |          WRTTMX P= 0
A=A-1 A                                  |          RTNCC
C=0   A                                      |
LC(2) 14                                  |          *****
GOSBVL =IDIVA                          |          * REPEAT stocke le paramètre P défini par
P= 0                                      |          * l'utilisateur sous la forme de son complément à
LC(5) =KCOLD                          |          * 15 ( complément(P)=15-P ). Ceci est dû au fait
C=C-B A                                  |          * que lorsque la mémoire est réinitialisée, la
D1=C                                      |          * valeur par défaut de P est zéro (0). Or il est
C=DAT1 XS                              |          * difficile d'utiliser le clavier lorsque l'option
LCHEX 3                                  |          * REPEAT 0 est active. Si par contre on stocke le
A=A&C P                                  |          * stocke le complément à 15 de P, la valeur par
RPTK30 A=A-1 P                          |          * défaut lors d'une réinitialisation ne sera plus
GOC RPTK40                              |          * zéro (0) mais quinze (15-0).
C=C+C XS                                  |          *****
GOTO RPTK30                              |          REL(5) fixdc
RPTK40 C=C+C XS                          |          REL(5) fixp REPEAT (Exp. num.)
GOC RPTK45                              |          REPEAT GOSBVL =EXPEXC Evalue l'expression
RPTKCC LC(2) =f1RPTD                    |          *          suivant l'ordre REPEAT
GOSBVL =SFLAGC                          |          GOSBVL =RNDAXX convers. en hex dans A(A)
TMRST C=0 W                               |          GONC ivaerr
C=C+1 XS                                  |          C=0 A erreur si >15
GOTO WRTTM1                              |          C=C-1 P
RPTK45 DO=(5) (=TIMER1)+5                |          ?C)=A A
D1=(5) =KEYPTR                          |          GOYES REP10
A=DATO P                                  |          ivaerr GOVLNG =IVAERR
LCHEX 7                                  |          REP10 A=-A-1 P Stocke dans la "Reserved
?A=C P                                    |          * Ram" (Mémoire réservée par
GOYES RPTK47                              |          * HP pour une future
GOSUB TMRST                              |          *          utilisation) le
DO=(5) (=KEYBUF)+2*14                  |          *          complément à 15 de P
A=DAT1 XS                                  |          DATO=A P
?A#0 XS                                  |          GOVLNG =NXTSTM
GOYES RPTKCC                              |          *****
A=DATO B                                  |          * NOTE:
B=0   A                                      |          * Toutes les routines utilisées par REPEAT sont
B=A   B                                      |          * supportées par HP. Néanmoins, cet ordre utilise
RTNSC                                      |          * aussi une partie de la "Reserved Ram" qui ne lui
RPTK47 A=DAT1 XS                          |          * est pas allouée en propre. En conséquence, des
?A#0 XS                                  |

```

\* conflits peuvent éventuellement avoir lieu entre  
 \* le Lex REPEAT et d'autres Lex. C'est pourquoi  
 \* nous vous communiquerons en temps utile  
 \* l'adresse de la partie de la "Reserved Ram" qui  
 \* nous sera allouée.

```
fixdc GOVLNG =FIXDC
fixp  GOVLNG =FIXP

      END
```

MARGIN

Le HP75 et le HP150 possèdent l'ordre MARGIN.  
 Aujourd'hui c'est chose faite, le HP71 aussi  
 peut "Beeper" lorsque le curseur se trouve à une  
 certaine position.

Syntaxe: MARGIN ( <exp. num.> )

Le HP71B ajuste la position de la marge en  
 fonction du paramètre numérique.

Michel MARTINET (SIG#2 P#12 CHHU#617)

```
LEX  'MARGELEX'
ID   #71
MSG  0
POLL POLHND
ENTRY MARGE
CHAR #D
KEY  'MARGIN' Syntaxe: MARGIN N
TOKEN 29
ENDTXT
```

```
pWTKY EQU #1C Poll intercepté pour la
      *   *   *   *   *   *   *   *   *   *   *   *   *
RESERV EQU #2F986 Le Lex utilise un octet de
      *   *   *   *   *   *   *   *   *   *   *   *   *
      *   *   *   *   *   *   *   *   *   *   *   *   *
      *   *   *   *   *   *   *   *   *   *   *   *   *
      *   *   *   *   *   *   *   *   *   *   *   *   *
```

```
CURSOR EQU #2F47E Octet donnant la position
      *   *   *   *   *   *   *   *   *   *   *   *   *
CHIRP  EQU #0EC5A Emet le BEEP d'erreur du
      *   *   *   *   *   *   *   *   *   *   *   *   *
EXPEX- EQU #0F178 Cette routine permet de
      *   *   *   *   *   *   *   *   *   *   *   *   *
      *   *   *   *   *   *   *   *   *   *   *   *   *
      *   *   *   *   *   *   *   *   *   *   *   *   *
RNDAXH EQU #136CB "POP", teste et transforme
      *   *   *   *   *   *   *   *   *   *   *   *   *
      *   *   *   *   *   *   *   *   *   *   *   *   *
      *   *   *   *   *   *   *   *   *   *   *   *   *
NXTSTM EQU #0BA4B Fin d'exécution de l'ordre
      *   *   *   *   *   *   *   *   *   *   *   *   *
      *   *   *   *   *   *   *   *   *   *   *   *   *
FIXP    EQU #02A6E Routine de compilation de
      *   *   *   *   *   *   *   *   *   *   *   *   *
      *   *   *   *   *   *   *   *   *   *   *   *   *
FIXDC   EQU #05493 Routine de décompilation
      *   *   *   *   *   *   *   *   *   *   *   *   *
```

```
*****
* POLHND
* But: intercepter le poll pWTKY, tester la
* position du curseur et faire un BEEP si elle
* concorde avec la valeur de la marge.
*****
POLHND LC(2) pWTKY On charge dans C(B) la
      *   *   *   *   *   *   *   *   *   *   *   *   *
      *   *   *   *   *   *   *   *   *   *   *   *   *
      *   *   *   *   *   *   *   *   *   *   *   *   *
      *   *   *   *   *   *   *   *   *   *   *   *   *
      *   *   *   *   *   *   *   *   *   *   *   *   *
      *   *   *   *   *   *   *   *   *   *   *   *   *
      *   *   *   *   *   *   *   *   *   *   *   *   *
```

```
* Il existe deux sortes de POLL: les "slow polls"
* et les "fast polls" ou FPOLL.
*
* Lors de l'émission d'un FPOLL, le système ne
* prend aucune précaution pour préserver son
* environnement. Il est donc impératif de sauver
* le contenu de D(A)
```

```
POLL  C=D  A
      R3=C  Sauvegarde.
      D1=(5) 1+RESERV D1 ^ l'endroit où est
      *   *   *   *   *   *   *   *   *   *   *   *   *
      *   *   *   *   *   *   *   *   *   *   *   *   *
      *   *   *   *   *   *   *   *   *   *   *   *   *
      *   *   *   *   *   *   *   *   *   *   *   *   *
      *   *   *   *   *   *   *   *   *   *   *   *   *
      *   *   *   *   *   *   *   *   *   *   *   *   *
      *   *   *   *   *   *   *   *   *   *   *   *   *
      *   *   *   *   *   *   *   *   *   *   *   *   *
```

```

* conservé plus haut dans
* R3(A)
D=C A Ok
RTNSXM On n'intercepte pas le
* POLL.

```

```

*****
* MARGIN
*****
REL(5) HARPEd
REL(5) MARGEp
MARGE 605BVL EXPEX- "Collapse" la Math Stack
* et y dépose le paramètre.
605BVL RNDAXX "POP", teste et convertit
* l'expression en un nombre
* HEXA dans A(A).
60NC SetA20 Si l'expression est < 0,
* on met 0 par défaut.
?A#0 A Si l'expression <> 0
60YES Maxmrg On passe au test suivant.
SetA20 A=0 A
Maxmrg LC(2) 96 On charge dans C(B): 96
?C>=A B L'expression <= 96 ?
60YES Setmrg Ok: Setmrg.
A=C A Autrement: on place 96
* par défaut.
Setmrg D1=(5) 1+RESERV D1 ^ (RESERV)+1
A=A-1 A On décrémente pour
* compatibilité avec CURSOR
DAT1=A B Sauvegarde de la marge
* dans (RESERV)+1
60VLNG NXTSTM Fin d'exécution ....

MARGEp 60VLNG FIXP
MARGEd 60VLNG FIXDC

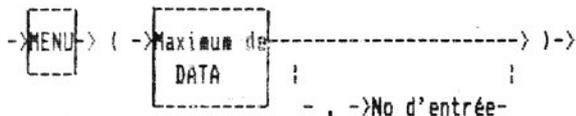
END

```

MENU

MENU affiche les éléments d'une liste DATA, un à un suivant l'utilisation des flèches ^ v ^ v. La pression de la touche Endline renvoie le numéro d'ordre de la DATA affichée.

- o ORDRE
- o Fonction
- o Opérateur
- o Exécution au clavier
- o Mode CALC
- o IF...THEN...ELSE



EXEMPLES

```

MENU (A,B) X = MENU (A,B)
MENU (A) DISP 'ESSAI: '; @ X = MENU (A)
MENU (A,B) ON MENU (A,B) GOTO 100,200,300

```

Eléments	Description	Restriction
Maximum	Expression numérique indiquant le nombre d'élément du menu.	< 255
Entrée	Expression numérique indiquant le numéro d'ordre du premier élément visualisé.	

COMMENTAIRES

Le premier élément visualisé par MENU est spécifié par le 2ème paramètre. Les éléments sont comptés à partir du début de la ligne contenant les DATA. La pression des flèches ^ v ^ v facilite le déplacement à l'intérieur du menu. Si l'on essaie d'accéder à plus d'éléments que n'en contient les DATA, il en résulte une erreur NO DATA. L'ordre RESTORE est utilisable pour spécifier le début de la liste des éléments. L'expression contenant MENU respecte l'affichage d'une expression précédente qui s'est terminée par un délimiteur (,) ou (;). Aucun message n'est affiché par défaut.

STITRE PPCPC

```

*****
*****
**
** Nom: MENU - Affiche les éléments de DATA et

```

```

**          renvoie le numéro de l'élément
**          retenu par Endline.
**
** Catégorie: Simplification de programme.
**
** Propos:   Menus arborescents
**
** Entrée:
**
** Sortie:
**
** Utilise: Voir le début du source.
**
** Stk lvls: 7
**
** Algorithmes:
**   Lit le premier parametre R3<-- 1er param
**   Si 1 seul parametre
**   | alors le deuxième parametre vaut 1
**   | par défaut. Goto A
**   | Sinon lit le 2ème parametre
**   A) Construit R3: Max[5,4] Ele[3,2] Cpt[1,0]
**   Sauve le masque d'écran dans FUNCRO
**   Lit les pointeurs de programme BASIC
**   L) Construit l'affichage: XEQ R
**   Actualise le Cpt R3[1,0]
**   F) Attend la pression d'une touche:
**   ATTN ---> quit
**   DOWN ---> D : Elément suivant. Goto L
**   UP ---> U : Elément précédent. Goto L
**   TOP ---> T : Premier élément. Goto L
**   BOT ---> B : Dernier élément. Goto L
**   EOL ---> E
**   Autre --> F
**   E) Goto évaluation d'expression NFRTN1
**   quit Ignore le calcul et passe au Statement
**   suivant.
**   R) Si le pointeur de DATA = 0
**   | alors prendre le pointeur de début
**   | de programme BASIC
**   Recherche les DATA
**   Si pas de DATA
**   | alors erreur NO DATA
**   Tant que le compteur d'éléments #0
**   | Lit une DATA compteur-1
**   Reconstitue le masque d'écran
**   Remplace le curseur à gauche
**   Affiche la DATA
**   Efface la suite de l'écran.
**
** Historique:
**   Date   Programmeurs   Modifications

```

```

** -----
** 05/07/85 JJ.D.      Création du mot
** 12/07/85 JJ.D & JJ.M  Optimisation
**

```

```

*****
*****

```

Jean-Jacques DHENIN (SIG#5 P#177)

```

LEX 'MENUMEX'
* TITRE : MENUMEX Jean-Jacques DHENIN SIG#5
* 336 12 05

```

```

ID      #71
MSG     0
POLL    0
=LINE  EQU #15275
=ASRW5 EQU #0E0A
=ASLW5 EQU #0ED1B
=BF2DSP EQU #01C0E
=CSLC5 EQU #1B435
=CSRC5 EQU #1B41B
=CRLFND EQU #0229E
=CURSFL EQU #151DF
=DSPMSK EQU #2F540
=DATPTR EQU #2F692
=DSPBUF EQU #09723
=HDFLT EQU #1B31B
=FUNCRO EQU #2F89B
=FINDA EQU #023E3
=FNRTN2 EQU #0F219
=IVAERR EQU #0E920
=MFERR EQU #09393
=NXTSTM EQU #08A48
=POPBUF EQU #010EE
=PRGMEN EQU #2F567
=PRGMST EQU #2F562
=PRSCKB EQU #07B88
=RNDAXH EQU #136CB
=RPTKY EQU #152BA
=SCRLLR EQU #0212E
=TKSCN+ EQU #08A6B
=WINDST EQU #2F471
=k#ATTN EQU #2B
=k#-LIN EQU #6B
=k#BOT EQU #A3
=k#DOWN EQU #33
=k#EOL EQU #26
=k#TOP EQU #A2

```

```

=k#UP EQU #32
*****
ENTRY menu
CHAR #F
KEY 'MENU'
TOKEN 33
ENDTXT

PARM GOSBVL =RNDAXX Convertir Reel sur MATSTK
      GONC INVPAR en Hex A(A)
      C=0 A
      C=C-1 B C(B)= FF
      ?C>=A A Restriction à 255
      RTNYES

INVPAR GOVLNG =IVAERR

Read DO=(5) =PRGMEN Limite de recherche
      C=DATO A pour les DATA
      D=C A Copie PRGMEN dans D(A)
      DO=(4) =DATPTR
      C=DATO A Lit DATPTR
      ?C#0 A DATPTR=0?
      GOYES R10 Non, alors OK. DATA
      DO=(4) =PRGMST Oui, alors utilise PRGMST
      C=DATO A

R10 DO=C
R11 A=DATO B Lit le token
      LCASC ', '
      ?A=C B Est-ce une virgule?
      GOYES R20 Oui, alors les data suivent
      DO=DO- 4
      ?A#0 P
      GOYES R15 Non, alors pas de ligne#
      DO=DO+ 4

R15 LCHEX C6
      GOSBVL =TKSCN+ trouve le data statement
      GOC R20
      GOSUB popbuf
      LCHEX 20 =eNODAT Error: No data

MFERRj GOVLNG =MFERR
R20 DO=DO+ 2
      CDOEX
      DO=C
      R2=C Sauve l'ad. du début de DATA
      LCASC ', '
      B=C B
      LC(2) 13

R60 A=DATO B
      ?A=B B

GOYES R80 Oui. Fin de la DATA
DO=DO+ 2
?A#C B Est-ce une fin de ligne?
GOYES R60 Non. Cherche encore

R80 C=R0 Rapel des pointeurs
      C=C-1 B Décrémente le pointeur.
      R0=C Sauve le pointeur.
      ?C#0 B Est-ce la DATA cherchée?
      GOYES R11 Non. Alors DATA suivante.

R83 R1=A Sauve le terminator.
      D1=(5) =FUNCRO D1 pointe la sauvegarde.
      DO=(5) =DSPMSK
      GOSUB dsp
      GOSBVL =CURSFL Curseur à gauche.
      A=R1 Rappel le terminator.
      C=R2 Rappel l'adresse de la DATA
      D1=C D1 pointe la DATA
      GOSBVL =DSPBUF Affiche la DATA
      GOVLNG =LINE efface le fin de ligne
      dsp C=DAT1 W
      DATO=C W
      D1=D1+ 16
      DO=DO+ 16
      P= 7
      C=DAT1 WP
      DATO=C WP
      P= 0
      RTN
      popbuf GOSBVL =CRLFND
      popb GOVLNG =POPBUF

* Début de l'exécution de MENU

NIBHEX 8812 2 chaînes num. 1 option.

menu GOSUB PARM Lit et convertit le 1er parm.
      D1=D1+ 16
      R3=A Sauve le paramètre
      A=0 A
      A=A+1 A
      C=C-1 S
      C=C-1 S
      GOC #10
      GOSUB PARM Lire le 2ème paramètre.
      D1=D1+ 16
      AR3EX

#10 C=R3
      CSL A
      CSL A

```

```

C=A B
ADOEX
GOSBVL =ASLW5
AD1EX
GOSBVL =ASLW5
R4=A
R0=C
R3=C
GOSUB pop
NIBHEX 02
NIBHEX FF
pop C=RSTK
D1=C
GOSBVL =BF2DSP
D0=(5) =FUNCRO D0 pointe la sauvegarde
D1=(5) =DSPMSK
GOSUB dsp
mAFF GOSBVL PRSCKB Lecture des pointeurs BASIC
GOSUB Read

menuF GOSUB popb Attend une touche.
GOSBVL =SCRLLR

menuR GOSBVL FINDA CASE OF de la touche
CON(2) k#ATTN
REL(3) quit
CON(2) k#DOWN
REL(3) menuD
CON(2) k#UP
REL(3) menuU
CON(2) k#TOP
REL(3) menuT
CON(2) k#BOT
REL(3) menuB
CON(2) k#EOL
REL(3) menuE
CON(2) 0
GONC menuF

menuU C=R3
C=C-1 B
?C#0 B
GOYES m+

menuT C=R3
C=0 B
C=C+1 B
m+ R0=C
R3=C
GOTO mAFF

menuD C=R3

```

```

A=C A
ASR A
ASR A
C=C+1 B
?C<A B
GOYES m+

menuB C=R3
A=C A
ASR A
ASR A
C=A B
GONC m+

menuE
GOSUB popbuf
GOSBVL =CRLFND
A=0 A
C=R3
A=C B
GOSBVL =HDFLT conversion flottant
SETHEX
C=A W
A=R4
GOSBVL =ASRW5
D1=A
GOSBVL =ASRW5

GOVLNG =FNRTN2
quit GOVLNG =NXTSTM
END

```

---

#### NOMBRES PREMIERS EN ASSEMBLEUR

Vous n'en pouvez plus de savoir où Dame Nature a caché ses mythiques Nombres Premiers ? Le PRIMLEX est alors fait pour vous. Son but est de déterminer si un nombre N donné est premier.

Si oui, PRIM(N) retourne N ; dans le cas contraire, le programme affiche le premier diviseur qu'il a rencontré. L'algorithme utilisé est bien sûr celui, classique, de la décomposition en facteurs premiers, adapté à cette tâche précise. Comme vous l'imaginez,

quelques améliorations y ont été apportées. Ainsi, on évite les multiples de 2, 3 et 5 parmi les diviseurs successivement essayés et on ne teste le caractère premier de N qu'à chaque entrée dans la boucle des diviseurs ( BCL ) et non pas à chaque utilisation de la sous-routine TEST, testant si N est divisible par le diviseur essayé.

Au chapitre des performances, les résultats obtenus m'ont paru un peu décevants car le gain en rapidité n'est que d'un rapport d'environ 3 à 1 par rapport au même programme écrit en BASIC. Ainsi, 82646281 est "décomposé" en 9091 \* 9091 en à peu près 29" 21/100 contre 87" 8/100 pour la version BASIC, et le caractère premier de 9999999967 est identifié en environ 5' 45" contre 16' en BASIC: rien de bien transcendant, comme vous pouvez le constater.

Pour conclure, des dix jours d'Enfer vécus pour l'écriture de ce LEX, j'ai pu dégager deux Règles fondamentales concernant l'écriture d'une routine assembleur, Règles que je vous livre pour votre édification personnelle :

- 2) Au Plus Inattendu Le Programme Plantera.
  - 3) Le Plus Simple En Premier Délirera.
- (la Règle 1 est réservée à mon usage personnel...)

A bientôt sur la même chaîne...

Olivier Arbey ( SIG#7-P118-T164 )

LEX 'PRIMLX'  
 ID #71  
 MSG 0  
 POLL 0

POPIN EQU #0BD1C Pop N à tester dans A  
 TST12A EQU #0D476 Teste 2 nombres à 12 chiffres virgule flottante  
 \*  
 DV2-12 EQU #0C4AB Divise ...  
 CLRFRC EQU #0C6F4 (<=> INT et carry levé si FP=0  
 \*  
 FLTDH EQU #1B223 12 chiffres décimal flottant -> HEXA dans A(A)  
 \*  
 HDFLT EQU #1B31B Le contraire, de A(A) dans A  
 \*  
 MPY EQU #0ECBB A\*C en HEXA

```

HXDCW EQU #0ECB4 C(W) HEXA -> décimal
FLOAT EQU #1B322 Entier décimal cadré à droite -> 12 chiffres flottant dans A(W)
FNRTN1 EQU #0F216 Retour

ENTRY PRM
CHAR #F
KEY 'PRIM'
TOKEN 30
ENDTXT

PRM NIBHEX 811
GOSBVL POPIN
R0=A N sauvé en R0
C=0 W Prépare D, diviseur à tester
*
R1=C D -> R1
P= 2 Divisible par 2 ?
GOSUB TEST
P= 1 ... par 3 ?
GOSUB TEST
P= 2 ... par 5 ?
GOSUB TEST
P= 2 Etc ...
GOSUB TEST
BCL A=R1 Début test primalité
GOSBVL FLTDH
C=0 W
C=A A D en HEXA dans C(A) ...
A=C W ... et A(A)
SETHEX
GOSBVL MPY D^2
GOSBVL HXDCW
GOSBVL FLOAT D^2 en décimal dans A
* pour le test
C=R0 N -> C
?A=C W D^2=N ?
GOYES F3 ...oui : N carré parfait
P= 4 Test >
GOSBVL TST12A D^2>N ?
GOC FIN2 ...oui : N premier
P= 4 Boucle évitant les multiples de 2, 3 et 5 parmi les diviseurs
*
GOSUB TEST
P= 2
GOSUB TEST
P= 4
GOSUB TEST
P= 2
GOSUB TEST

```

	P=	4	:	
	GOSUB	TEST	:	
	P=	6	:	Lorsque je vis le Lex INKLEX, tout droit venu
	GOSUB	TEST	:	des Etats-Unis, je me dis qu'il devait être
	P=	2	:	possible de faire mieux. Aussitôt dit, aussitôt
	GOSUB	TEST	:	fait. Un week-end a été nécessaire pour que les
	P=	6	:	onze fonctions voient le jour.
	GOSUB	TEST	:	
		Fin boucle	:	
	GOTO	BCL	:	Le but est d'offrir à tous les utilisateurs
F3	GOTO	FIN3	:	d'imprimante (HP82905B ou HP2225B) un lot de
	*	Saut intermédiaire ( GOC	:	mot-clefs tous plus utiles les uns que les autres.
	*	limité à 128 quartets dans	:	Vous trouverez donc, dans PRINTLEX, les
	*	chaque sens )	:	fonctions et ordres suivants:
FIN2	C=R0	Retourne le premier ( gag	:	
	*	interne... ) diviseur	:	BELL
	*	trouvé	:	
	GOVLNG	FNRTN1	:	Syntaxe: BELL
			:	
TEST	C=0	W	:	Test de divisibilité
	C=P	0	:	
	R2=C	P -> C(0)	:	Pour possesseurs de HP82905B. Il n'y a pas de
	A=R1		:	raisons qu'il n'y en ait que pour la HP2225B
	GOSBVL	FLTDH	:	(ThinkJet pour les intimes). BELL active la
	C=A	A	:	sonnerie.
	A=0	W	:	
	A=C	A	:	BOLD
	C=R2		:	
	A=A+C	A	:	Syntaxe: BOLD ON:OFF
	GOSBVL	HDFLT	:	
	*	D en décimal flottant	:	Pour possesseurs de HP2225B, cela signifie
	*	pour sauvegarde	:	"passage en mode gras".
	R1=A		:	Pour les possesseurs de HP82905B (comme moi),
	A=R0	N -> A	:	cela veut dire: "passage en mode caractères
	C=R1	D -> C	:	alternés".
	GOSBVL	DV2-12	:	
	*	A/C et résultat en 15	:	
	*	chiffres sur A-B	:	
	GOSBVL	CLRFRC	:	Lève le carry s'il n'y a
	*	pas de FP	:	CR
	RTNNC	Retour si FP#0 ( N non	:	
	*	divisible par D )	:	Syntaxe: CR
			:	
FIN1	C=RSTK	Pop le GOSUB en attente	:	Bien utile, cela ne vaut vraiment pas la peine
FIN3	C=R1	Retourne N, nombre premier	:	de s'en passer. Remplace bien souvent un ENDLINE
	GOVLNG	FNRTN1	:	CHR\$(13) embêtant, ou une belle acrobatie pour
			:	obtenir le double passage.
	END		:	
			:	ESC\$
			:	
			:	Syntaxe: ESC\$
			:	
			:	La seule fonction de PRINTLEX. Fonction, car
			:	utile aussi bien sur vidéo que sur imprimante (ou
			:	encore dans un fichier texte). Voilà.

HISTOIRE D'IMPRESSIONNER

FF

Syntaxe: FF

Je vais vous expliquer quelque chose: mon imprimante est une HP82905B. Elle est située loin de mon bureau (au moins 1m). Lorsqu'il s'agit de faire avancer le papier jusqu'à la page suivante, il faut que je me lève, que j'appuie sur OFF-LINE, et c'est fatigant. Alors, j'ai fait FF. Et, utilisant aussi à certains moments des ThinkJet, je peux vous assurer que c'est plus reposant de faire FF [ENDLINE] que de lever la main, l'amener jusqu'à l'imprimante, atteindre le bouton blanc, et appuyer...

LF

Syntaxe: LF

Voir paragraphe précédent.

MODE

Syntaxe: MODE <expression numérique>

Changement de caractère: le paramètre (de 0 à 9) indique le mode choisi: 0 pour normal, 1 pour 40 caractères/ligne, 2 pour 132 (142) caractères/ligne...

PERF

Syntaxe: PERF ON:OFF

Active ou désactive le saut de perforations. Il y aura bien quelqu'un pour utiliser un jour PERF OFF.

PL

Syntaxe: PL [ <exp. num.> [ , <exp. num.> ] ]

Règle la longueur de la page. Ce mot-clef est, à mon avis, très intéressant.

Le premier paramètre définit la longueur de la

page, le deuxième définit la longueur du texte dans la page. Attention toutefois: le deuxième paramètre, lorsqu'il est présent, est envoyé en premier.

Lorsque le deuxième paramètre est omis, il n'y a aucune valeur par défaut (c'est à dire que l'imprimante garde sa valeur antérieure).

Lorsque les deux paramètres sont omis, PL engendre une séquence "standard": papier 12", texte de 66 lignes, saut de perforations. Pour une fois que ceux qui utilisent du papier 12" (format # A4) ne sont pas ignorés, cela mérite d'être signalé...

UNDERLINE

Syntaxe: UNDERLINE ON:OFF

Pour possesseurs de HP2225B seulement. UNDERLINE ON place l'imprimante en mode "caractères soulignés". UNDERLINE OFF retourne en mode normal.

WRAP

Syntaxe: WRAP ON:OFF

Pour possesseurs de HP2225B seulement. Par défaut, ThinkJet "oublie" les caractères dépassant la ligne (pour un défaut, c'est un défaut...). WRAP ON permet d'imprimer le reste du texte sur la ligne suivante. WRAP OFF remet les choses dans l'état initial. WRAP a été écrit à la suite d'une suggestion de Michel MARTINET.

#### NOTES SUR LE PROGRAMME

Le Lex fait 565 octets au total. Il me semble, à moi, bien optimisé. Je serai heureux d'ouvrir toute amélioration éventuelle.

Tous les mot-clefs définis dans PRINTLEX sont transparents vis-à-vis de TAB.

En effet, si vous faites:

```
PRINT "AREUH"
```

```
PRINT CHR$(27)&"&l72p66f1L";TAB(12);"TAGADA"
```

vous pouvez constater que "TAGADA" n'est pas placé en colonne 12. Cela provient du fait que



```

TOKEN 6+t           ; * - PR-RTN: retour à la routine appelante.
KEY 'PERF' PERF ON:OFF ; *
TOKEN 7+t           ; * Principe:
KEY 'PL' PL [<num.> [ , <num.> ] ] ; * Le poll pPRTIS est envoyé, pour être intercepté
TOKEN 8+t           ; * par l'HPIL. Celui-ci renvoie l'adresse de la
KEY 'UNDERLIN' UNDERLINE ON:OFF ; * routine PRASC qui se chargera de l'impression.
TOKEN 9+t           ; *
KEY 'WRAP' WRAP ON:OFF ; * Schéma de l'appel:
TOKEN 10+t          ; * GOSUB PR-BAS (ou PR-RTN)
                        ; *   CON(1) n
                        ; *   CON(2) C1
                        ; *   :
                        ; *   CON(2) Cn
ENDTXT
POLL EQU #12337
pPRTIS EQU #0F Le poll pour imprimer.
MFERR EQU #09393 Main-Frame Error
NXTSTM EQU #08A48 Retour à Basic
PPDS EQU #2F956 Printer-Position (<=> TAB)
tON EQU #E0 Token de DN
EXPEX- EQU #0F178 Evaluation de l'expression
      * pointée par D0
POP1R EQU #0E8FD Obtention du résultat.
RJUST EQU #12AE2 conversion en entier déc.
DO=AVS EQU #09B2C
ADHEAD EQU #181B7 Ajoute une en-tête à la
      * chaîne sur la math-stack.
OUTELA EQU #05303 Décompilation simple.
TRACDC EQU #052FC Décompile de TRACE
WRDSCN EQU #02C2A Recherche du Token
REST# EQU #03035 Réinitialisation de
      * l'analyse (parse)
EOLCK EQU #02A7E Fin de ligne ?
DELAYp EQU #02AC6 Analyse de DELAY
RESPTR EQU #03172 Utilitaire d'analyse
DROPDC EQU #05470 Décompilation de DROP
FIXP EQU #02A6E Analyse de FIX
ARGERR EQU #0BF19 "Argument non valide"
FORSTK EQU #2F59E Début de la math-stack
ASLW3 EQU #0ED21 Utilitaire
ASLW4 EQU #0ED1E " "
OUT2TC EQU #02CFD " "
CNVMUC EQU #03FB8 Conversion en majuscules
SYNTXe EQU #02E2B "Erreur de syntaxe"
D=AVMS EQU #1A460 Utilitaire
EOLXCK EQU #05405 Fin de ligne ?

*****
* PRINT
* But: Cette routine imprime la chaîne pointée par
* la première adresse de la pile de retours.
* Le retour peut s'effectuer de deux manières:
* - FR-BAS: retour à Basic
      ;
      ; C=A A
      ; RSTK=C
      ; A=0 A
      ; P= 0
      ; A=DAT1 WP A(A)= nb de caractères.
      ; RTN <=> GOTO A(A)
      ; retour ?ST=1 4

```

RTNYES            Retour à l'appelant  
nxtstm GOVLNG NXTSTM    sinon retour à Basic

\*\*\*\*\*  
\$ onoff?  
\$ But: tester le token suivant:  
\$ Sortie: Cy=1 si tON, Cy=0 sinon  
\*\*\*\*\*  
onoff? A=DAT0 B  
          LC(2) tON  
          ?A=C B  
RTNYES            Retour avec Carry=1

\$ Ici, après le test, nous savons que Carry=0.  
\$ Puisqu'une routine d'analyse (parse) simple doit  
\$ être RTNCC, il était tentant de profiter de la  
\$ situation actuelle pour faire un RTN avec mise à  
\$ zéro de la Carry.  
\$ Bilan: 2 quartets économisés. Eh oui, un octet,  
\$ c'est toujours 8 bits !

parse RTNCC            Avec Cy=0

\*\*\*\*\*  
\*\*  
\*\* Routines d'analyse et de décompilation  
\*\* (ou encore: décomparse)  
\*\*  
\*\*\*\*\*

decomp GOVLNG OUTELA

\*\*\*\*\*  
\$ UNDERd  
\$ But: décompiler UNDERLINE ON:OFF  
\*\*\*\*\*  
UNDERd LCASC ' E'  
          D0=D0- 2        Retour sur l'espace  
          GOSBVL OUT2TC    Affichage de "E"  
          A=DAT1 B        Token suivant (ON/OFF)

\*\*\*\*\*  
\$ ONOFFd  
\$ But: décompiler une alternative ON:OFF  
\*\*\*\*\*

ONOFFd GOVLNG TRACDC    Décompilation du token

\*\*\*\*\*  
\$ UNDERp  
\$ But: analyser UNDERLINE ON:OFF  
\*\*\*\*\*  
UNDERp GOSBVL CNVWUC    Conversion en majuscule  
          LCASC 'E'  
          ?A#C B        Est-ce "E" ?  
          GOYES syntax    Non: "Erreur de syntaxe"  
          D1=D1+ 2        Oui: caractère suivant

\*\*\*\*\*  
\$ ONOFFp  
\$ But: analyser ON:OFF  
\*\*\*\*\*  
ONOFFp GOSBVL WRDSCN    Token suivant  
          CON(2) tON     Est-ce ON ?  
          REL(3) parse    Oui: retour  
          CON(2) 1+tON    (= tOFF )  
          REL(3) parse  
          NIBHEX 00        Fin de la table  
\$ Si nous sommes arrivés là, c'est que le token  
\$ n'était ni tON, ni tOFF. Alors, on ne reconnaît  
\$ pas l'ordre, et on repasse la main au système  
\$ d'exploitation.

rest# GOVLNG REST#    Reprendre l'analyse

syntax GOVLNG SYNTXe    "Erreur de syntaxe"

\*\*\*\*\*  
\$ PLp  
\$ But: analyser PL [num [, num ]]  
\$ Principe: on teste le premier paramètre, et s'il  
\$ existe, on continue sur DELAYp  
\*\*\*\*\*  
PLp GOSBVL EDLCK    Premier param. existe ?  
          GOC resptr    Non: on revient à Basic  
          GOSUB resptr    Oui: on branche sur  
          GOVLNG DELAYp    DELAY num [, num]

\*\*\*\*\*  
\$ NUMp  
\$ But: analyser un ordre de type ORDRE <num>  
\*\*\*\*\*  
NUMp GOVLNG FIXP

\*\*\*\*\*  
\$ PLd, NUMd  
\$ But: décompiler un ordre suivi d'une suite  
\$ d'expressions numériques.

```

*****
PLd
NUMd GOVLNG DROPDC Aucun param.: autorisé

```

```

resprr GOVLNG RESPTR

```

```

*****
$ BELL $
*****

```

```

REL(5) decomp
REL(5) parse
BELLe GOSUB PR-BAS
CON(1) 1 1 seul caractère
CON(2) 7 code 7

```

```

*****
$ BOLD ON:OFF $
*****

```

```

REL(5) ONOFFd
REL(5) ONOFFp
BOLDDe GOSUB onoff? ON ou OFF ?
GOC boldon -> ON
GOSUB PR-BAS -> OFF
CON(1) 1 1 seul caractère
CON(2) 15 SI (Shift In)
boldon GOSUB PR-BAS
CON(1) 1
CON(2) 14 SO (Shift Out)

```

```

*****
$ CR $
*****

```

```

REL(5) decomp
REL(5) parse
CRe GOSUB cr CR est aussi utilisé dans
GOTO nextstm LF et FF

```

```

*****
$ cr

```

```

$ But: envoyer un retour-chariot, et remettre la
$ position de la tête d'écriture à 0.
$ Note: la position de la tête d'écriture est
$ gardée en mémoire par le HP71. Elle est utilisée
$ par TAB.

```

```

*****
cr D1=(5) PPOS Printer Position
C=0 B
DAT1=C B PPOS=0 (TAB=0)
GOSUB PR-RTN Impression avec retour...
CON(1) 1

```

```

CON(2) 13
rtn RTN ...ici

```

```

*****
$ ESC$ $
*****

```

```

NIBHEX 00 Fonction sans paramètre
ESCe CD1EX D1=pointeur de pile math.
R1=C dans R1 pour ADHEA
D1=C
D1=D1- 2
P= 0
LC(2) 27 ESC
DAT1=C B
GOSBVL D=AVMS requis par Basic
ST=0 0 Pas de retour
GOVLNG ADHEAD Ajout de l'en-tête

```

```

*****
$ FF $
*****

```

```

REL(5) decomp
REL(5) parse
FFe GOSUB cr On remet d'abord PPOS à 0
GOSUB PR-BAS et on envoie...
CON(1) 1 ...un caractère...
CON(2) 12 ...de code 12

```

```

*****
$ LF $
*****

```

```

REL(5) decomp
REL(5) parse
LFe GOSUB cr Même principe que FF
GOSUB PR-BAS
CON(1) 1
CON(2) 10

```

```

d0=avs GOVLNG D0=AVS

```

```

*****
$ MODE <num> $
*****

```

```

REL(5) NUMd
REL(5) NUMp Un param. num. obligatoire
MODEEe GOSUB evalex Evaluation de ce paramètre
C=0 W
C=A WP Ici, on a P=0
?A#C W nb > 9 ?
GOYES argerr

```

```

SETHEX
LCASC '0'      Conversion en Ascii
A=A+C B
GOSBVL ASLW4  Décalage à gauche de 7
GOSBVL ASLW3  quartets
LCHEX 53006B261B5 ESC '&k_S'

```

```

* A ce niveau, il est bien important de voir ce
* que l'on a dans les registres:
*
* C(W)= x x x x x 5 3 0 0 6 B 2 6 1 B 5
* A(W)= 0 0 0 0 0 0 3 n 0 0 0 0 0 0 0 (MODE n)
*
* L'addition de A et C nous donne bien évidemment
* la bonne séquence d'échappement.
*
* C va alors être transféré en mémoire (AVMEMS),
* afin d'être envoyé à PRASC

```

```

C=C+A W
GOSUB d0=avs DO reçoit AVMEMS
DAT0=C 11 La table est créée à cet
CDOEX endroit, pointé par DO.
RSTK=C On simule un GOSUB.
GOTO PR-BAS

```

```

*****
* PERF ON/OFF *
*****
REL(5) ONOFFd
REL(5) ONOFFp
PERFe GOSUB onoff?
A=0 A
LCASC 'L' Fin de séquence
GONC perfoff
A=A+1 A l=perf on
perffoff GOSUB escseq
GOTO nxtstm retour à Basic

```

```

*****
* evalex, trad
* But:
* - evalex: évaluer la chaîne de tokens pointée
* par DO, et...
* - trad: dépiler le sommet de la pile-maths,
* convertir le nombre en décimal, et vérifier
* qu'il est inférieur à 128.
*****
evalex SETHEX requis par EXPEX-
GOSBVL EXPEX- évaluation...
trad GOSBVL RJUST Traduction en entier déc.
SETHEX après RJUST

```

```

GOC argerr Inf
C=0 W
LCHEX 128 C=00...00128
?AKC W
RTNYES Retour autorisé si < 128
argerr GOVLNB ARGERR sinon: "invalid argument"

```

```

*****
* escseq
* But: Envoyer une séquence du type: ESC & l ### $
* ou ### est dans A(A), et $ dans C(B)
*
* Note:
* Beaucoup de séquences d'échappement sont de la
* forme ESC & l # $, où:
* # est un nombre entier
* $ est un caractère (/F, p/P, l/L ou d/D)
* Il est à noter que les zéros de tête dans # ne
* sont pas pris en compte. Cette caractéristique
* est très intéressante...

```

```

* Principe:
* - La table est construite à partir de AVMEMS:
* Les caractères ESC, & et l sont d'abord
* placés, puis on calcule # sur 3 chiffres
* (octets), et enfin, on place le caractère $
* à la fin. Il est à noter que cette séquence
* d'échappement est de longueur fixe.
* - La table est suivie en mémoire d'un code RTN
* (#01). Le retour de PR-RTN se fera là, d'où
* retour final à la routine appelante...
* Simple, non ?

```

```

*****
escseq D=C B Sauvegarde du caractère
B=A A et du nombre
GOSUB d0=avs
LCHEX 6C261B7 7 caractères, dont: ESC & l
DAT0=C 7
DO=DO+ 7
BSRC
BSRC B(0-0)=premier nombre
LCASC '0' C(B)=#30
P= 2 3 boucles au total
boucle A=B A A(A)=0000x
B=0 A B=xx0000...0000
A=A+C B A(B)=3x
DAT0=A B
DO=DO+ 2
BSLC B=x0000...0000x
P=P-1
GONC boucle Boucle si P)=0
C=D B caractère final

```

DAT0=C B	:	GOSBVL POP1R	
D0=D0+ 2	:	GOSUB trad	
LCHEX 01 code de RTN	:	AD1EX Contorsions abominables !	
DAT0=C B	:	unique AD1EX	
D0=D0- 15 D0=début de la table	:	LCASC 'P' Longueur de la page	
GOSUB 1a	:	GOSUB escseq	
1a C=RSTK C(A)=pointeur programme	:	nxtstm GOTO nxtstm Retour à Basic	
A=C A	:		
LC(5) (rtn)-(1a) adresse d'un RTN	:	*****	
C=C+A A	:	* UNDERLINE ON:OFF *	
RSTK=C empilée...	:	*****	
CDOEX	:	REL(5) UNDERd	
RSTK=C puis l'adresse de la table	:	REL(5) UNDERp	
GOTO PR-RTN Envoi...	:	UNDERe GOSUB onoff?	
	:	GOC UNDon	
argerR GOTO argerr	:	GOSUB PR-BAS Séquence d'échappement	
	:	CON(1) 4 difficile à paramétrer	
*****	:	CON(2) 27	
* PL [ <num> [ , <num> ] ] *	:	CON(2) '&'	
*****	:	CON(2) 'd'	
REL(5) PLd	:	CON(2) 'e'	
REL(5) PLp	:	UNDon GOSUB PR-BAS	
1a A=DAT0 B A(B)=token suivant	:	CON(1) 4	
GOSBVL EOLXCK Fin de ligne ?	:	CON(2) 27	
GONC parametres	:	CON(2) '&'	
GOSUB PR-BAS Oui: chaîne par défaut	:	CON(2) 'd'	
CON(1) 11 11 caractères	:	CON(2) 'D'	
CON(2) 27	:		
CON(2) '&'	:	*****	
CON(2) '1'	:	* WRAP ON:OFF *	
CON(2) '7'	:	*****	
CON(2) '2'	:	REL(5) ONOFFd	
CON(2) 'p'	:	REL(5) ONOFFp	
CON(2) '6'	:	WRAPe GOSUB onoff?	
CON(2) '6'	:	GOC wrapon	
CON(2) 'f'	:	GOSUB PR-BAS	
CON(2) '1'	:	CON(1) 5	
CON(2) 'L'	:	CON(2) 27	
parametres	:	CON(2) '&'	
GOSUB evalex Evaluation des paramètres	:	CON(2) 's'	
D1=D1+ 16 Dépilement	:	CON(2) '1'	
AD1EX	:	CON(2) 'C'	
R1=A Sauvegarde de D1	:	wrapon GOSUB PR-BAS	
D0=(5) FORSTK	:	CON(1) 5	
C=DAT0 A	:	CON(2) 27	
?A>=C A Un seul paramètre ?	:	CON(2) '&'	
GOYES unique Oui: longueur de la page	:	CON(2) 's'	
AD1EX	:	CON(2) '0'	
LCASC 'F' Longueur du texte	:	CON(2) 'C'	
GOSUB escseq	:		
A=R1	:	END	
D1=A	:		

GESTION DE TOUCHES EN ASSEMBLEUR

Lorsque l'on rédige un programme quelque peu interactif, l'on est souvent amené à utiliser les touches dites " Cursor keys ". Pour savoir si l'une d'elles est enfoncée, il faut réaliser une série de tests à l'aide des fonctions KEY\$, KEYDOWN ou KEYWAIT\$. Cette gestion de touches, longue à mettre en oeuvre et dispendieuse en octets, n'est pas toujours assez rapide ( Cf programme REFLEX de Serge VAUDENAY, JPC No25 de mai 1985 ). Certains ont déjà résolu ce problème, mais les solutions adoptées jusqu'ici, bien qu'elles ne manquent pas d'élégance, ont été trop particulières.

Je vous propose ce mois-ci (en avant première) deux nouvelles fonctions: CURSKEYS et CURSDOWN. La première renvoie une valeur comprise entre 0 et 6, suivant que vous avez appuyé ou non sur une des touches du curseur (Les codes correspondent aux codes utilisateurs suivant: 1=#47, 2=#48, 3=#50, 4=#51, 5=#162 et 6=#163). Cette fonction est autrement équivalente à KEY\$. CURSDOWN est elle équivalente à KEYDOWN, mais ne teste que les touches de codes utilisateurs #47, #48, #50, #51. L'argument renvoyé appartient donc à l'intervalle [0,4].

Vous pouvez désormais remplacer le test fait par S. VAUDENAY et J-P BONDU par, respectivement :

ON CURSKEYS GOTO ...

ON CURSDOWN GOTO ...

Jean-Jacques MOREAU -

LEX 'ONKEYLEX'  
ID #71  
MSG 0  
POLL 0

ENTRY onkey  
CHAR #F  
ENTRY nokey  
CHAR #F  
KEY 'CURSDOWN'  
TOKEN 31  
KEY 'CURSKEYS'  
TOKEN 32

KEYSCN EQU #00D4D Détermine quelles sont les touches sur lesquelles on a appuyé depuis le dernier appel de cette routine. Est exécutée chaque fois que l'on appuie sur une touche. C'est cette routine qui est responsable de la définition de la touche ATTN et du couple de touches (ON,/). C'est donc elle qui, éventuellement appelle la routine de Memory Lost.

POPBUF EQU #010EE Retire une touche du buffer de touches si carry est désarmé

FINDA EQU #023E3 Saut dans une table, suivant la valeur du code en A(B)

FNRTN2 EQU #0F219 Exprime le résultat placé en C(W) puis retourne à l'interpréteur BASIC

ENDTXT

NIBHEX 00 Fonction CURSDOWN, sans paramètres

onkey C=0 P  
C=C+1 P

INTOFF L'instruction qui suit ne peut être interrompue

OUT=CS OUT=1 (Cf article de J-J MOREAU sur les registres IN et OUT, JPC No26, juin 1985). Lit IN en A(3-0).

A=IN

ASL A Ne garde que la partie utile de IN.

A=A+A A



```

nokyl P= 14
      GOTO noky46

nokyr P= 14
      GOTO noky44

nokyu P= 14
      GOTO noky42

nokyd P= 14
      GOTO noky40

nokyt P= 14
      GOTO noky38

nokyb P= 14
      GOTO noky36

      END

```

INDATES

INDATE\$ facilite la saisie de la date dans un format européen et renvoie après la pression de la touche Endline, la date saisie dans un format américain.

- ```

: o ORDRE           ■ Exécution au clavier :
: ■ Fonction        o Mode CALC
: o Opérateur       ■ IF...THEN...ELSE

```



EXEMPLES

```

INDATE$      SETDATE INDATE$
A$=INDATE$

```

COMMENTAIRES

```

: Cette fonction est destinée à contourner le BUG
: 1078-5 qui peut produire une erreur Data Type
: lors d'une concaténation.
: D'autre part, INDATE$ remplace la routine BASIC:
: 20 DIM I$[110]
: 30 C$=CHR$(27)
: 40 I$=C$&"<DATE "&C$&">JJ"&C$&"</"&C$&">MM"
: 50 I$=I$&C$&"</"&C$&">AA"&C$&"<"
: 60 I$[110]= " "
: 70 INPUT "",I$;D$
: 80 D$=D$[7,8]&D$[3,6]&D$[1,2]

```

```

      STITRE PPCPC
      *****
      *****
      **
      ** Nom: INDATE$ saisie la date en format Europ.
      **
      ** Catégorie: Simplification de programme.
      **
      ** Propos:      Saisie date Européenne.
      **
      ** Entrée:
      **
      ** Sortie:
      **
      ** Utilise: Voir le début du source.
      **
      ** Stk lvs:
      **
      ** Historique:
      ** Date      Programmeurs      Modifications
      ** -----
      ** 19/07/85  JJ.D.              Création du mot
      **

```

```

      *****
      *****

```

Jean-Jacques DHENIN (SIG#5 P#177)

```

LEX 'INDATLEX'
ID #71
MSG 0
POLL 0
* TITRE : INDATE Jean-Jacques DHENIN SIG#5
* 336 12 05
=CREDIT EQU #14C99
=ADHEAD EQU #181B7
=CSLC6 EQU #18432

```

```

=CSLC4 EQU #1B438
=CUROFF EQU #B1C3
=CURON EQU #B1E3
=D=AVMS EQU #1A460
=DSP#00 EQU #1B5DB
=BF2DSP EQU #01C0E
=CURSFL EQU #151DF
=FINLIN EQU #18A3A
ENTRY IND
CHAR #F
KEY 'INDATE$'
TOKEN 50
ENDTXT

IND NIBHEX 00 Pas de paramètre
CD1EX )
R1=C ) Sauvegarde des
CDOEX ) pointeurs D0 et D1
R2=C ) dans R1 et R2
GOSUB POP Place dans RSTK l'adresse
$ de l'instruction suivante.
NIBHEX #B1C3 CUROFF ) Chaîne à afficher.
NIBASC 'DATE : "' )
NIBHEX #B1E3 CURON ) CURON et CUROFF
NIBASC 'JJ' ) établissent
NIBHEX #B1C3 CUROFF ) les champs
NIBASC '/' ) protégés.
NIBHEX #B1E3 CURON )
NIBASC 'MM' )
NIBHEX #B1C3 CUROFF )
NIBASC '/' )
NIBHEX #B1E3 CURON )
NIBASC 'AA' )
NIBHEX #B1C3 CUROFF )
NIBASC '""' )
NIBHEX FF ) Fin de la chaîne.
POP C=RSTK Adresse début de chaîne.
D1=C D1=adresse
GOSBVL BF2DSP Affichage de chaîne.
GOSBVL =CURSFL Remplace le curseur.
GOSBVL CHEDIT Entrée des données.
GOSBVL =FINLIN Eteint le curseur.
C=R2 )
CDOEX ) Restitution de D1 et D0.
C=R1 )
D1=C )
ST=1 0 Retour après DSP#00
GOSBVL =DSP#00 Place sur MATSTK les
$ caractères non protégés.
D1=D1+ 16 Passe l'en-tête.
D1=D1+ 2 Passe le CR.
C=DAT1 12 C[[1,0]=JJMAAA

```

```

D1=D1+ 12 Passe la chaîne.
AD1EX ) Sauvegarde dans
R1=A ) R1(A) la fin de la
AD1EX ) nouvelle chaîne.
D1=D1- 16 D1=Début de chaîne.
CSLC )
CSLC ) C= .JJMAAA.
LCHEX 2F C= .JJMAA/
GOSBVL =CSLC6 C= MAAA/.JJ
A=C A A(A)= .JJ
GOSBVL =CSLC4 C= AA/.JJMM
P= 9 P= ^
CSL WP )
CSL WP )
CSL WP )
CSL WP )
CSL WP ) C= AA/MM...
C=A A C= AA/MM.JJ
P= 4 P= ^
LCHEX 2F C= AA/MM/JJ
P= 0
DAT1=C W Pose la chaîne sur MATSTK.
GOSBVL =D=AVMS D(A)= AVMEMS.
ST=0 0 Retour direct à BASIC.
GOVLNG =ADHEAD Retour à BASIC.
END

```

FORMATTONS, FORMATTONS !! (SUITE)

Il y a déjà quelques mois, je vous avais présenté un petit formateur de texte. Nous l'utilisions tellement pour faire JPC qu'il nous fallait un outil plus performant. L'assembleur est venu à notre secours. En un week-end est né FORMALEX. Je devrais plutôt dire en deux nuits !

Plus tard est arrivé CENTER\$, pour parfaire la panoplie.

Ce que nous vous montrons ce mois-ci n'est qu'une étape. Tel quel, ce programme nous rend d'excellents services. Sans lui, ce n'est pas 150h que nous passerions à faire JPC, mais 300 !. La prochaine étape sera l'assemblage complet

sous la forme d'un seul ordre.

Mais ce Lex ne sera pas démodé pour autant. Avec ses fonctions CESURE, CENTER\$, FORMAT\$, SPACE\$ et REDUCE\$, il est devenu indispensable dans bien des circonstances.

Le programme Basic a beaucoup maigri, suite à cette cure. Nous vous le montrons:

```
10 INPUT "Fichier: ";F$
  @ INPUT "Format: ", "50";N @ T=TIME
  @ CALL FORMAT(F$,N) @ T=TIME-T
1000 SUB FORMAT(F$,N) @ DIM A$(256),B$(256+N)
1010 SFLAG -1 @ PURGE FTX @ CFLAG -1
  @ CREATE TEXT FTX @ ASSIGN #1 TO F$
  @ ASSIGN #2 TO FTX
1020 P=0 @ ON ERROR GOTO 1050 @ READ #1,0;A$
  @ OFF ERROR @ IF A$[1,1]="\" THEN 1060
1030 IF A$[1,1]="^"
  THEN A$=CENTER$(REDUCE$(A$[2]),N)
1040 DELETE #1,0 @ PRINT #2;A$ @ GOTO 1020
1050 PURGE F$ @ RENAME FTX TO F$ @ END
1060 A$=A$[2] @ IF A$[1,1]="^"
  THEN A$=A$[2] @ P=CEIL(N/10)
1070 B$="" @ CFLAG 5 @ GOTO 1090
1080 READ #1,0;A$
1090 DELETE #1,0 @ A$=REDUCE$(A$)
1100 IF A$[LEN(A$)]="\"
  THEN SFLAG 5 @ A$=A$[1,LEN(A$)-1]
1110 IF NOT LEN(A$) THEN 1140 ELSE B$=B$&" "&A$
1120 A$="" @ IF LEN(B$)<=N-P
  THEN 1140
  ELSE A$=B$[CESURE(B$,N-P)+2]
  @ B$=B$[1,CESURE(B$,N-P)]
1130 B$=SPACE$(P)&FORMAT$(B$,N-P) @ PRINT #2;B$
  @ B$=A$ @ P=0 @ IF LEN(B$)>N THEN 1120
1140 IF FLAG(5)
  THEN PRINT #2;SPACE$(P)&REDUCE$(B$)
  @ GOTO 1020
  ELSE 1080
```

Voilà, ce petit bout de Basic formate 100 octets en une seconde. Notons pour l'anecdote que 80% du temps est passé à recopier le fichier plutôt qu'à le formater.

Etudions maintenant plus en détail chacune des fonctions:

CENTER\$

Syntaxe: CENTER\$ ( <exp. alpha.> , <exp. num.> )

CENTER\$(A\$,N) centre la chaîne A\$ dans N caractères. En clair: met (N+LEN(A\$))/2 blancs devant.

Si N<LEN(A\$) ou LEN(A\$)=0, A\$ est renvoyée telle quelle.

CESURE

Syntaxe: CESURE ( <exp. alpha.> , <exp. num.> )

CESURE(A\$,N) rend la position du premier blanc à partir du Nième caractère où on peut couper A\$ (vers la gauche). Détail très important: CESURE est programmée de telle manière que les signes de ponctuation ").:;!?,\" ne soient jamais en début de ligne.

Si LEN(A\$)<N, alors CESURE renvoie LEN(A\$). Si la chaîne est insécable, N est renvoyé.

FORMAT\$

Syntaxe: FORMAT\$ ( <exp. alpha.> , <exp. num.> )

FORMAT\$(A\$,N) formate la chaîne A\$ à N caractères (rajoute suffisamment de blancs). La chaîne est d'abord réduite, avant d'être formattée.

FORMAT\$ est la fonction la plus intéressante du lot.

Note: il y a une "bogue" dans FORMAT\$. Nous ne l'avons pas enlevée, car elle donne des résultats meilleurs que ceux prévus...

Si LEN(A\$)>N alors A\$ est renvoyée telle quelle.

REDUCE\$

Syntaxe: REDUCE\$( <exp. alpha.> )

REDUCE\$(A\$) réduit la chaîne A\$, c'est à dire enlève tous les espaces superflus. Utilisée systématiquement par FORMAT\$.

REDUCE\$ enlève les blancs de tête et de queue, et laisse un blanc à chaque fois qu'il en rencontre plusieurs de suite.



```

ST=1  flag  Il faut enlever les
*      blancs suivant
ST=0  boucle aucun passage dans la
*      boucle
ST=0  caract Caractère non blanc non
*      rencontré
LCASC  ' '  C=' '
B=C    B    B[B]=32 (espace)
Enlève A=A-1 A  Longueur en quartets
GOC    Fin  Saut si longueur nulle
ST=1  boucle au moins un passage dans
*      la boucle
A=A-1  A    quartets...
DO=DO- 2  caractère suivant
C=DATO B  Il est recopié dans C
?C#B    B   est-ce un espace ?
GOYES  noblan non
?ST=1  flag Si non, est-ce le premier
*      d'une série ?
GOYES  Enlève non, alors on le néglige
D=D+1  A    oui: un emplacement
*      supplémentaire
ST=1  flag  C'est le premier d'une
*      série
*      B.E.T.
noblan ST=0  flag  Premier d'une série:
*      aucun sens, puisque non
*      blanc
ST=1  caract Au moins un caractère non
*      blanc dans la chaîne
ajoute D1=D1- 2  On range le caractère
*      C[B] ds la chaîne réduite.
DAT1=C  B
GONC   Enlève B.E.T.
Fin    ?ST=0  flag
GOYES  nonnul
?ST=0  boucle
GOYES  nonnul
?ST=0  caract
GOYES  nonnul
D1=D1+ 2  Si le dernier caractère
*      était un blanc, on
*      l'enlève,
D=D-1  A    ainsi que du nombre
*      d'emplacements
nonnul RTN
*****
*      SPACE$      *
*****
NIBHEX 811

```

```

SPACE  GOSBVL RNDAXH
GOC    positif
A=0    A
positif D1=D1+ 16
CD1EX
D1=C
R1=C
GOSBVL D=AVMS
GOSUB  stkblc
GOC    ret2
stkblc LCASC  ' '  On charge " " dans C(B)
GONC   test  Reste-t-il des caractères
*      à empiler ? (B.E.T.)
bsp    GOSBVL STKCHR  Qui: on empile un blanc
*      devant la chaîne
*      alphanumérique.
test   A=A-1  A    A(A) sert de compteur.
GONC   bsp    Boucle si A>=0
RTN
*****
*      REDUCE$      *
*****
NIBHEX 411
REDUCE GOSUB reduire
*****
*      retour
*      But: assurer le retour à l'environnement Basic,
*      et prendre en charge la restauration des
*      pointeurs.
*****
retour C=R1      C=....DO...
GOSBVL CSRC5    C=.....DO
CDOEX          DO=pgm counter
ret2  ST=0  0    ADHEAD ne rend pas la main
GOSBVL D=AVMS
GOVLNG ADHEAD
argerr GOVLNG ARGERR
*****
*      CENTER$      *
*****
NIBHEX 8422
CENTER GOSUB Argnum  Pop, teste et convertit en
*      HEXA le paramètre
*      numérique.
GOSBVL REVPOP      "Pop", test et renverse
*      la chaîne alphanumérique.
*      En sortie de routine, D1

```

```

*      pointe après l'en-tête de      ; * dûment vérifié et testé (>0).
*      la chaîne.                      ; * Le nombre est renvoyé dans A(A) et R0(A) en hexa
CD1EX  Chargement dans C(A) de        ; * En sortie, on a Carry=0
*      la valeur de D1.                ; *****
D1=C   Restitution.                  ; Argnum  GOSBVL RNDAXX  A(A)=nombre hexa
C=C+A  A  C(A) = maintenant           ;          GONC  argerr  Argument Error si <0
*      l'adresse de fin de chaîne     ;          ?A=0  A      Paramètre nul
*      (utilisé plus loin par la      ;          GOYES argerr  ... alors Arg. Error
*      routine ADHEAD).              ;          D1=D1+ 16   POPer le paramètre
R1=C   Sauvegarde dans R1 pour       ;          R0=A      et le mettre dans R0
*      être utilisé dans ADHEAD.      ;          RTN
GOSBVL D=AVMS  D(A) = AVMEMS (AVaila ; memerr  GOVLNG MEMERR
*      ble MEMory Start). L'opération ; *****
*      est faite maintenant car       ; *      FORMAT*      *
*      la routine utilise C(A).      ; *****
C=R0   On charge dans C(A) la        ;          NIBHEX 8422
*      valeur du paramètre           ;          FORMAT  GOSUB  Argnum  A[A],R0[A]=nombre hexa
*      numérique.                    ;          GOSUB  reduire
C=C+C  A  C(A) = paramètre          ;          C=R1      C=.....(^début de la
*      numérique en quartets.        ;          *      chaîne)
GOC   argerr  En cas de dépassement ;          A=R0      A=paramètre
*      l'argument était supérieur    ;          C=C-A  A
*      à la longueur maxi. d'une     ;          GOC  memerr
*      chaîne: ERREUR.              ;          C=C-A  A  C=^fin de la chaîne
?A=0  A  La longueur de la chaîne   ;          *      formattée
*      est-elle nulle ??            ;          GOC  memerr
GOYES  FIN  Si oui: FIN.            ;          R2=C      R2=^fin de la chaîne
?C<=A  A  La longueur de la chaîne ;          *      formattée
*      >= à la longueur demandée?   ;          A=C  A  Pour transfert ensuite
GOYES  FIN  Si oui: on renvoie      ;          *      dans D0
*      simplement la chaîne.        ;          DO=C      }
C=C-A  A  Calcul du nombre de       ;          DO=DO+ 16  }+16 pour l'en-tête
*      blancs: C(A) = 4 fois le      ;          GOC  memerr  }
*      nombre de blancs.            ;          CDOEX   }
A=0   M  ) Division de C(A)        ;          DO=(5)  AVMEMS
A=C   A  ) par 4                    ;          ADOEX
ASRB   ) et restitution dans A(A)   ;          ?C<=A  A  a-t-on dépassé AVMEMS ?
ASRB   ) en octets.                ;          GOYES  memerr non: pas d'erreur
GOSUB  stkb1c                       ;          C=R1      C=^début de la chaîne
FIN  ST=1  0  Obligatoire pour un  ;          A=0   W  }
*      bon retour de ADHEAD.        ;          A=C   A  } A=00...000 ^début
GOSBVL ADHEAD  Mise en place de    ;          CD1EX  C=^fin de la chaîne
*      l'en-tête de la nouvelle     ;          *      réduite
*      chaîne.                      ;          D1=C      D1= idem
GOSBVL REV$   On renverse à nouveau ;          A=A-C  A  A=longueur réduite
*      avant le                      ;          ASRB   ... en octets
GOVLNG EXPR   retour à BASIC.      ;          C=R0      C=longueur formattée
*      *****                      ;          ?A>C  A  erreur si long(réduite) >
*      Argnum                        ;          *      long(formattée)
*      But: renvoyer le paramètre   ;          GOYES  strovf
qui est sur la pile,

```

```

?A=C A Retour si chaîne déjà | D1=D1+ 2
* formattée... | DATO=A B dans la chaîne formattée
GOYES retour | D0=D0+ 2
A=A+1 A long(chaîne réduite) | ?C#A B est-ce un espace ?
?D=0 A Peut-on mettre des blancs? | GOYES bb non: on continue
GOYES retour non: la chaîne réduite | A=B A Boucle interne sur A.
* est renvoyée | * Elle est exécutée A fois
A=A-C A | GONC b2
A=-A A | recopie DATO=C B Un espace de plus dans la
C=D A | * chaîne formattée
R3=C R3= nb total | D0=D0+ 2
* d'emplacements | b2 A=A-1 A
GOSBVL IDIVA A/C | GONC recopie On continue tant que A>=0
P= 0 P=15 après IDIVA | D=D-1 A Un emplacement de moins
* A: Quotient B:C: Reste | bool ?D#0 A Est-ce le dernier
* Il faut mettre C fois | * emplacement ?
* A+1 blancs | GOYES bb Non, on continue
* et (le reste) fois A | RTN Oui: retour à l'envoyeur
* blancs |
R0=C R0 (- C | *****
B=A A | * CESURE *
B=B+1 A | *****
GOSUB rajout | NIBHEX 0422
A=R0 A=longueur de la chaîne | CESURE GOSUB Argnum Retourne un nombre HEXA
* formattée | * dans A(A) et R0(A)
C=R3 C=nb total d'emplacements | B=A A Sauvegarde de la césure
C=C-A A C=(le reste) | * dans B(A).
B=B-1 A (un espace en moins par | GOSBVL POP16 Vérifie que l'on est bien
* boucle) | * en présence d'une chaîne
GOSUB rajout | * et retourne sa longueur
C=R2 | * dans A(A).
D1=C pointeur de pile-math | C=A A Division de A(A) par 2.
retour GOTO retour | A=0 M "
| A=C A "
***** | ASRB "
* strovf | CD1EX C(A) ^ Début de la
* But: renvoyer l'erreur "Dépassement de chaîne" | * chaîne.
* (pour les heureux possesseurs de FRALEX). | D1=C "
* Pour les autres, il n'y a plus rien à faire... | C=C+A A "
***** | C=C+A A "
strovf LC(2) 37 String Overflow | D=C A On sauvegarde l'adresse de
GQVLNG MFERR | * début de chaîne dans D(A)
| * longueur(chaîne) < césure?
* rajout | GOYES FinC Oui, on renvoie la chaîne
* But: rajouter C(A) espaces à la chaîne | * tout simplement
***** | C=C-B A On place D1 sur le
rajout D=C A D=compteur. | * premier caractère concerné
LCASC ' ' C=' ' | C=C-B A par la césure
GOTO bool | D1=C "
bb A=DAT1 B Recopier le dernier | LCASC ');,.,!?' * Tous les
* caractère de la chaîne | * caractères de comparaison
* réduite | ST=C On met ST(0) à 0: flag

```

```

Boucle A=DAT1 B      A(B)=caractère à comparer.
?A=C   B      Est-ce un espace ?
GOYES  TestP   Oui: on va en TestP pour
*
*           comparaison avec les
*           autres caractères de C(W)
ST=1   flag    On a rencontré au moins
*           un caractère différent
*           d'un espace
AutreB B=B-1 A    Décrément de la césure
GDC    Nobl    Reste-t-il encore des
*           caractères à tester ?
D1=D1+ 2      Oui, on place D1 sur le
*           suivant
GONC   Boucle  B.E.T.
Nobl   A=RO    A(A) = césure
GDC    FinC    B.E.T.
TestP  A=RO    A(A) = Valeur de la césure
?A=B    A      La position de l'espace =
*           césure ?
GOYES  NotstP  Oui, on retourne chercher
*           l'espace suivant en
*           décrémentant la césure

* La séquence suivante teste si A(B) est un des
* caractères de ponctuation.
* Elle prend moins de place dans ce Lex que
* l'appel à la routine interne: MEMBER
P=     7      Autrement test: P=
*           compteur
D1=D1- 2     D1 ^ le caractère suivant
*           l'espace
A=DAT1 B    A(B) = caractère à tester
Bcl2  CSR   W  Décalage du registre C
CSR    W    "
?A=C   B    Le caractère est-il égal ?
GOYES  Suite  Oui: on branche en "Suite"
P=P-1    Décrémentation du compteur
GONC   Bcl2  Il reste des caractères à
*           tester dans C
GDC    Décr  Il n'en reste plus
NotstP ?ST=1 flag  A-t-on déjà rencontré un
*           caractère différent d'un
*           espace
GOYES  AutreB  Oui: on ne décrémente pas
*           la césure (elle pointe ce
*           caractère !)
A=A-1 A    On décrémente la césure
RO=A      RO <- la césure
GONC   AutreB  B.E.T.
Suite  C=RO    La position du blanc =
*           la césure ?
?B=C    A    " " "

```

```

GOYES  Décr   Oui: on branche en fin de
*           programme directement
B=B+1  A      Autrement on ajuste le
*           résultat pour
B=B+1  A      englober le caractère de
*           ponctuation
Décr   A=B   A  Passage de B(A) dans A(A)
*           pour l'appel de la
*           routine FLTDH
A=A-1  A      Décrément de A(A) pour
*           être juste
FinC   C=D   A
D1=C    D1 pointe maintenant le
*           début de la chaîne.
D1=D1- 16    D1-16 nous sommes prêt à
*           placer le paramètre
*           numérique
GOSEVL HDFLT Transformation de A(A) en
*           nombre décimal dans A(W)
C=A    W
GOVLNG FNRTN4 Retour à BASIC.

END      Sig !

```

```

* La séquence suivante teste si A(B) est un des
* caractères de ponctuation.
* Elle prend moins de place dans ce Lex que
* l'appel à la routine interne: MEMBER
P=     7      Autrement test: P=
*           compteur
D1=D1- 2     D1 ^ le caractère suivant
*           l'espace
A=DAT1 B    A(B) = caractère à tester
Bcl2  CSR   W  Décalage du registre C
CSR    W    "
?A=C   B    Le caractère est-il égal ?
GOYES  Suite  Oui: on branche en "Suite"
P=P-1    Décrémentation du compteur
GONC   Bcl2  Il reste des caractères à
*           tester dans C
GDC    Décr  Il n'en reste plus
NotstP ?ST=1 flag  A-t-on déjà rencontré un
*           caractère différent d'un
*           espace
GOYES  AutreB  Oui: on ne décrémente pas
*           la césure (elle pointe ce
*           caractère !)
A=A-1 A    On décrémente la césure
RO=A      RO <- la césure
GONC   AutreB  B.E.T.
Suite  C=RO    La position du blanc =
*           la césure ?
?B=C    A    " " "

```

-----

Chers amis,

Félicitations à tous les membres qui nous ont offert des routines en assembleur, mais à ce sujet, quelques questions:

- En cas d'erreur à l'assemblage, peut-on stopper l'assemblage sur le message d'erreur, ou au moins avoir le temps de le lire ?
- Le module Forth contient la nouvelle fonction Basic (non décrite) EDPARSE\$ (variable alphanumérique), son exécution donne des résultats étranges, que fait-elle ?

Pour terminer, voici une nouvelle définition du mot VLIST en Forth, qui liste les mots un par un en attendant l'appui d'une touche.

```

: VLIST
  41 0
  DO
    I 0=

```

```
IF LATEST
  ELSE E0000 I + @
THEN
BEGIN
  DUP COUNT IF AND TYPE
  KEY DROP
  CR 5- @
  ?DUP 0=
UNTIL
  5
+LOOP
;
```

Amicalement  
Christian BACHELET

Cher Christian

Voir les erreurs d'assemblage n'est guère aisé, surtout sans imprimante ou interface vidéo. La solution, s'appelle alors "Le Listing-File".

C'est une option de l'assembleur qui permet de produire des listages comme celui paru dans JPC No20. Les erreurs apparaissent alors comme le nez au milieu de la figure.

Le problème principal qui peut se poser est le manque de place en mémoire. En effet, le listing produit est très gros. Il faut alors essayer de localiser l'erreur, puis d'encadrer la région susceptible avec un LIST ON et un LIST OFF.

Note: on peut simplement avoir le nombre d'erreurs (et quelques autres choses en plus) en faisant LIST OFF à la deuxième ligne du LEX, et LIST ON juste avant le END.

Quant à la fonction EDPARSE\$, elle est utilisée par l'éditeur de texte. Son rôle est d'analyser la commande entrée, d'isoler les paramètres, et ceci en tenant compte du traducteur de messages (FRALEX ou autre).

Tu vois donc que son usage en dehors du contexte de l'éditeur de texte est délicat. Bon courage !!!

Pierre DAVID

## NAVIGATION

Je voudrais vous soumettre ci-après un bien modeste programme qui, toutefois, me rend de grands services lors de la préparation de mes navigations aériennes.

Ce programme ne s'intéresse pas à la préparation "statique" de la navigation; c'est-à-dire que le tracé de la route a déjà été reporté sur la carte et les caps magnétiques calculés. De plus, étant donné que je pratique la navigation à vue, les distances entre repères ont été déterminées.

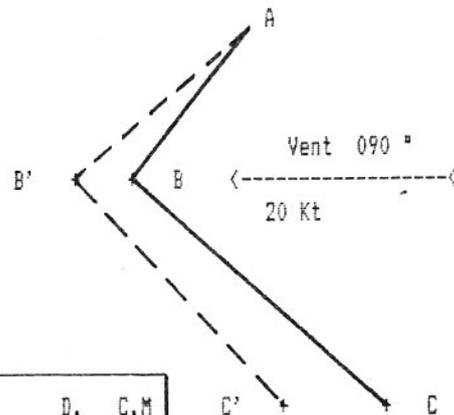
Ces éléments constituent l'étape préalable qui précède la préparation "dynamique" dans laquelle notre 71 excelle tout particulièrement (mais a-t-il vraiment beaucoup de mérites dans ce cas précis...). En effet, avant tout voyage aérien, il est absolument impératif de se renseigner auprès d'un Centre Météo afin de connaître les conditions atmosphériques en cours !

Les données qui sont transmises sont diverses. Parmi celles-ci nous intéressent particulièrement, dans ce programme, la direction du vent et sa force; la direction du vent est toujours exprimée en degrés à partir de son origine : un vent du 180 est un vent qui vient du Sud et qui souffle plein Nord. La force du vent quant à elle est exprimée en noeuds...

Ne soyez pas surpris : en Aviation ne règnent que des unités de barbares... Je ne citerai que les distances en milles, les altitudes en pieds et les vitesses en noeuds !!! (Les pleins d'essence se font quand même en litres et non pas en gallons).

Le meilleur moyen pour illustrer le fonctionnement du programme est de prendre un exemple: imaginons une navigation partant du point A et se rendant au point C en passant par B. Nous aurons donc déterminé sur la carte que la distance entre A et B est de 35 Kms et que pour aller de A à B il faut prendre un cap magnétique théorique au 230, c'est-à-dire en fait s'il n'y a pas de vent. Il faut préciser

que ce cap magnétique théorique est le résultat de ce que l'on appelle la Route Vraie, qui est déterminée sur la carte à l'aide d'un rapporteur, à laquelle a été appliquée la déclinaison magnétique du lieu (par ex. - 5° Ouest en Région Parisienne). Puis de B à C, nous aurons calculé une distance de 50 Kms avec une route au 120.



|       | D. | C.M |
|-------|----|-----|
| A - B | 35 | 230 |
| B - C | 50 | 120 |

Malheureusement, tout cela n'est que théorie, car au moment du départ, le météorologiste nous a indiqué un vent du 090 soufflant à 20 noeuds. Si nous conservions les caps magnétiques calculés sur la carte, de A nous nous retrouverions en B' et de B' en C' : gênant, non ?

Il va donc falloir rectifier ces caps et recalculer les temps de vol en fonction du vent, car ces derniers se trouvent augmentés lorsque l'on vole avec le vent dans le nez et diminués dans le cas contraire.

Le programme nous interrogera donc tout d'abord sur la direction du vent qui devra être entrée en degrés, puis sur sa force en noeuds. Ces 2 données, direction et force, sont supposées définitives et sans changement sur le parcours entre les repères; elles ne seront donc entrées qu'une seule fois au début de l'exécution. Enfin, nous indiquerons la distance entre A et B (35 Kms) et le cap magnétique (230).

Il faudra alors choisir pour l'édition des résultats entre l'affichage sur l'écran et l'impression sur une imprimante. A ce propos, je dois vous indiquer que je me sers pour cela de l'imprimante thermique HP82162A car celle-ci est portable, peu encombrante et bien suffisante pour cette utilisation.

Les résultats suivants apparaîtront donc, soit sur l'écran, soit sur l'imprimante:  
(pour l'écran, appuyez à chaque fois sur une touche)

|                   |   |                       |  |
|-------------------|---|-----------------------|--|
| T.S.V             | = | 14'                   |  |
| Dérive (X)        | = | 14.4°                 |  |
| Dérive sur axe    | = | - 9.3°                |  |
| Correction temps  | = | - 11''                |  |
|                   |   |                       |  |
| Temps corrigé     | = | 11' 25"               |  |
| Nouveau cap magn. | = | 221° (au lieu de 230) |  |

Le T.S.V, ou Temps Sans Vent, est calculé en multipliant la distance par le Facteur de base (Fb). Ce facteur de base est un rapport établi en divisant 60 par la vitesse de croisière de l'avion (sans vent). Les avions sur lesquels je vole ayant une vitesse moyenne de 150 Kms/h, le facteur de base est donc  $60 / 150 = 0.4$ . Celui-ci, ayant été placé dans une variable en début de programme, peut-être facilement modifié et adapté à tous types d'avions. Ce T.S.V est donc le temps que mettrait l'avion pour parcourir la distance de 35 Kms séparant A de B s'il n'y avait pas de vent...

Ensuite nous avons besoin de connaître quelle serait la dérive maximale, appelée X, si l'avion se trouvait en plein travers, à 90°, par rapport au vent. Celle-ci serait de 14.4°. Ce résultat nous sert après pour déterminer la correction d'axe que nous devons effectuer pour conserver notre route entre A et B. Cette correction s'appelle la dérive sur l'axe et elle est ici de - 9.3°, c'est-à-dire qu'au lieu de garder un cap au 230, qui nous mènerait en B', nous allons suivre un cap magnétique au 221; ce que nous confirmera l'ordinateur avec son dernier résultat. Pour les puristes, que vous êtes certainement, je dois préciser que 221 est un résultat arrondi

et que dans la pratique le cap à suivre serait même arrondi à 220 car il est très difficile en vol à vue de conserver une route à 5° près.

La correction de temps donnée ensuite par le programme est exprimée en secondes par minutes de vol, bien entendu en plus ou en moins selon que l'on vole contre le vent ou avec celui-ci dans le dos. Ici le vent nous pousse, donc nous irons plus vite et mettrons moins de temps pour aller de A à B que s'il n'y avait pas de vent. La correction de temps est alors négative et sera à soustraire du résultat T.S.V.

En fait, jusqu'à maintenant, ce ne sont que des résultats intermédiaires qui ont été indiqués, pour contrôle du bon fonctionnement du programme, et seuls nous intéresseront pour notre navigation les résultats finaux exprimés dans les deux dernières lignes.

Le temps corrigé, quant à lui, est obtenu en soustrayant (sur cette portion de navigation) au T.S.V (14') :  $14 * (-11) = 154''$ . Nous mettrons donc, pour aller de A à B, 11'25" au lieu de 14' puisque le vent nous est favorable.

Pour l'autre partie de cette navigation imaginaire, ou pour les autres parties, car nous pourrions très bien aller ensuite de C à D puis de D à E..., le programme nous interroge directement sur la distance et le cap magnétique suivants sans revenir sur la force et la direction du vent qu'il suppose sans changement, comme je vous l'avais déjà indiqué. Pour sortir du programme, répondez donc 0 pour la distance !

Ah ! J'oubliais, pour se rendre de B à C il faut 24'09" en temps corrigé avec un cap magnétique au 113. Si tels ne sont pas les résultats que vous délivre votre 71, vérifiez à tout hasard votre programme... Et attention aux vents trop forts !...

J'espère que cet exposé un peu long vous aura intéressé et que certains d'entre vous trouveront, si ce n'est l'utilisation de ce programme, peut-être l'envie de l'appliquer en situation réelle "sur le terrain" (d'aviation bien sûr...).

A bientôt dans les airs de notre chère vieille France.

Jean-Marie SIMON P#160 T#516

FORMULES UTILISEES:

$$Fb = \frac{60}{\text{Vit. croisière}}$$

$$\text{Dérive (X)} = Fb * \text{Vit. Vent}$$

(en degrés) (en Kms/H)

$$\text{Vit. Vent} = (2 * \text{Vit. Vent}) - 10\%$$

(en Kms/H) (en noeuds)

$$\text{Dérive sur l'axe} = X * \sin \theta$$

(en degrés)

$\theta$  étant l'angle au vent (angle formé par l'axe de l'avion avec la direction du vent).

$$\text{Correction temps} = X * \cos \theta$$

(en "/' de vol)

$$T.S.V = Fb * \text{Distance} \quad (\text{en Kms})$$

$$\text{Temps corrigé} = T.S.V +/- (\text{Correct. temps} * T.S.V)$$

-----

UTILISATION DE LA TABLE TRACANTE X710  
AVEC LE HP71B ET LE FICHER PLOTLEX

(Ce complément, fait suite au banc d'essai paru dans le JPC No 15 de juin 1984.)

I.- Utilisation générale:

Certains membres possédant le HP71 et la table traçante canon X710 éprouvent quelques

difficultés d'utilisation. Pour retrouver l'ensemble des ordres qui pilotent le X710, reportez vous à l'article paru dans le numéro 15. Avec la 41, vous deviez faire: (1) "code dec en X, XTOA, QUTA" pour envoyer les codes, l'utilisation en basic avec le 71B est beaucoup plus simple. Il suffit simplement d'envoyer "PRINT CHR\$(code)...". Pour envoyer un ensemble de plusieurs codes, vous devez séparer les différents "CHR\$(...)" par le symbole "&" de concaténation de chaîne. Dans son utilisation générale avec le 71B, vous devez définir le X710 comme imprimante par:

-RESTORE IO (surtout ne pas l'oublier),  
-PRINTER IS %64, ou PRINTER IS INTRFCE (définit le convertir comme PRINTER),  
-DISPLAY IS \* ou DISPLAY IS "" ou DISPLAY IS NULL (ce mode est conseillé, sinon vous risquez d'avoir des caractères indésirables à l'impression. Si vous désirez imprimer tout ce qui se passe à l'affichage, vous devez définir le mode DISPLAY IS %64).

II.- Utilisation avec le fichier PLOTLEX: (2)

Le fichier PLOTLEX est un utilitaire pour HP71 et table traçante X710. C'est un fichier Lex écrit par Stéphane Parizien et Jean-Jacques Dhénin. Il est disponible au Club sur envoi soit d'un support magnétique, soit sur commande de ce support.

Après avoir listé toutes les fonctions contenues dans ce Lex, nous allons maintenant les étudier une à une.

III.- Fonctions contenues dans PLOTLEX:

-----  
! COLOR ! équivalent à l'ordre C.  
!-----

Permet de changer la couleur utilisée. La syntaxe est "COLOR 0, COLOR 1, COLOR 2, COLOR 3"; pour respectivement, les couleurs noire, bleue, verte et rouge. La table traçante utilise le paramètre de COLOR modulo 4 si celui-ci est supérieur à 3.

!-----!

| CSIZE |      équivalent à l'ordre S.  
|-----|

Permet de changer la taille des caractères utilisés. La syntaxe utilisée est CSIZE n où n est un nombre compris entre 0 et 15. Si ce nombre est supérieur à 15, la machine utilise le résultat de n modulo 16.

| DRAW |      équivalent à l'ordre D.  
|-----|

Tracé absolu. Exécute le tracé trait de la position actuelle de la plume à un point de coordonnées (x,y). La syntaxe est: DRAW x,y.

| HOME |      équivalent à l'ordre H.  
|-----|

L'ordre HOME, qui s'utilise sans paramètre, soulève la plume et la ramène à l'origine courante.

| LABEL |     équivalent à l'ordre P.  
|-----|

La syntaxe utilisée est LABEL "chaîne...", et permet d'écrire du texte en mode graphique.

| LDIR |      équivalent à l'ordre Q.  
|-----|

Cet ordre permet d'écrire dans les quatre directions. La syntaxe utilisée est LDIR 0, LDIR 1, LDIR 2 ou LDIR 3. Les autres valeurs utilisées seraient prises modulo 4.

| LTYPE |     équivalent à l'ordre L.  
|-----|

Cet ordre permet de sélectionner le type de trait voulu. La syntaxe utilisée est LTYPE n où n peut prendre n'importe quelle valeur de 0 à 15. Si n est supérieur à 15, n modulo 16 sera utilisé. Dans le cas particulier où n=0, le plotter trace

un trait continu.

| MOVE |      équivalent à l'ordre M.  
|-----|

La syntaxe utilisée est MOVE x,y. Cet ordre soulève la plume, et l'amène au point de coordonnées (x,y). L'ordre provoque un déplacement absolu de la plume. L'ordre MOVE 100,200 soulève la plume et l'amène au point de coordonnées (100,200).

| PLOTTER |  
|-----|

Cet ordre permet l'initialisation de la table traçante comme PLOTTER. Il valide toutes les fonctions incluse dans le Lex PLOTLEX et doit donc être utilisé avant toute opération utilisant ces fonctions. La syntaxe utilisée est: PLOTTER IS %64 (%64 étant le numéro d'ID du convertisseur.), au même titre que les ordres de l'HPIL: PRINTER IS ... et DISPLAY IS ...

| RDRAW |     équivalent à l'ordre J.  
|-----|

Cet ordre permet le tracé relatif. Trace un trait de la position actuelle du crayon jusqu'au point de coordonnées (x,y). La syntaxe utilisée est RDRAW x,y.

| RESET |     équivalent à l'ordre A.  
|-----|

Ce mode permet une réinitialisation totale de la table traçante. Le passage du mode graphique au mode texte, la coordonnée Y est remise à 0 et l'origine est placée à l'extrémité gauche. La rotation des caractères est remise à 0 et les données fixées par L, C et S restent inchangées.

| RMOVE |     équivalent à l'ordre R.  
|-----|

Cet ordre permet le déplacement relatif de la plume. L'ordre REMOVE 0,-100, avance le papier de 100 points.

```
-----  
| SETORG | équivalent à l'ordre I.  
|-----|
```

Cet ordre qui s'utilise sans paramètre, redéfinit la position actuelle de la plume comme origine.

J'espère que ces quelques explications vous aideront à mieux comprendre l'utilisation de votre X710 munie de son meilleur pilote, j'ai nommé: Le HP71B.

Philippe.

(1) - Note de J.J. Dhénin. Le module Paname HP41 contient, parmi bien d'autres, tous les ordres décrits ci-dessus.

(2) - Il existe deux autres versions de ce Lex:  
- HPGL pour HP7470A ou PAC-SCREEN  
- CE516 pour la nouvelle table traçante sharp format A4.

Demandez les au Club.

---

#### A PROPOS DE DISP%

Lorsque l'on rédige un programme que l'on souhaite voir utiliser par d'autres, il est particulièrement important de contrôler de près la gestion des entrées/sorties, afin de limiter les "plantages" toujours mal venus. L'acquisition de données se faisant exclusivement (sauf rares exceptions...) par l'intermédiaire de INPUT ou LINPUT, je ne m'intéresserai qu'à ces ordres. La lecture attentive de JPC vous a permis de constater que nombreux sont les programmes commençant par POKE "2F441", "F". Ceci permet de

banaliser la touche ATTN, c'est à dire de la gérer comme n'importe quelle autre: sa pression n'interrompt plus le programme. Le problème n'est pas pour autant résolu, car même dans ces conditions son emploi rend la main à l'utilisateur lors d'un INPUT. Et c'est justement ce que nous voulons éviter... Bien sûr il reste le langage Machine, mais ne brûlons pas les étapes et profitons des 240 fonctions BASIC que nous avons à notre disposition.

Tout de suite une question pour ne pas vous endormir: connaissez-vous DISP% ?

A ceux qui ont répondu oui, un Super Banco: savez-vous vous en servir ? De toute manière comme dans l'école des fans, tout le monde a gagné et est renvoyé à son Manuel de Référence!... Soyons sérieux (un exemple nous y aidera).

Suite à une question, nous voulons que l'utilisateur puisse librement répondre (accès à tous les symboles du clavier), mais ne soit pas en mesure d'interrompre le programme. Le remède est simple: nous redéfinirons les touches "sensibles", c'est à dire ON(#43), OFF(#99), USER(#0) ainsi que celles de votre choix. Leur définition sera: [':], qui signifie: aucune action ([']) et action immédiate ([:]). L'effet résultant s'apparente à un retour chariot à une nuance près: l'affichage n'est pas assigné à la variable d'entrée. Le mot affichage désigne la réponse à la question, et non l'énoncé de la question. Par exemple tant que vous n'avez pas répondu à la question INPUT "Nom ";A\$, l'affichage est considéré vierge, bien que l'on y lise le mot "Nom ". Tapez "TOTO", et alors DISP% est égal à TOTO, bien que vous n'avez pas encore validé par la touche [END LINE]. Il est important de comprendre cette subtilité car elle est la clef de notre problème. N'hésitez pas à relire les quelques lignes précédentes si je ne me suis pas clairement fait comprendre. Cette précision apportée, voyons sur l'exemple suivant:

```
05 POKE "2F441", "0"  
06 DEF KEY "#43", ":" : @ DEF KEY "F0", ":" : ! etc...  
10 INPUT "Fichiers:";R$  
20 IF R##DISP% THEN 10  
30 IF NOT(LEN(R#)) THEN RETURN  
40 . . .
```

En ligne 20, on a R\$=DISP\$ à 2 conditions:  
1- vous avez utilisé [End Line] pour valider la réponse: l'affichage est affecté à R\$.  
2- vous avez utilisé une des touches redéfinies sans rien avoir répondu.

C'est la ligne 30 qui lèvera cette ambiguïté. Cela signifie que le renvoi en ligne 10 n'aura lieu qu'à une seule condition: ayant commencé à inscrire une réponse, vous avez utilisé une des touches redéfinies. A ce moment DISP\$<>"", mais R\$="". L'effet produit est le même que si l'on avait pressé sur ATTN dans un INPUT normal: la réponse est effacée, la question posée (ce n'est pas comme moi ...).

Ligne 30: si aucune réponse n'a été faite, retour. Sinon on continue le traitement avec le nom du fichier en R\$. Simple, non ?

J'espère que mes explications sont assez claires et que cette astuce vous aidera dans la réalisation de vos propres programmes. Elle me semblait suffisamment utile pour mériter votre attention. Si vous n'en êtes pas tout à fait convaincu, sachez qu'elle me fût révélée lors de l'étude attentive de l'Editeur de Texte du module FORTH/ASS, ce qui est quand même une référence, non ?

Jean-Pierre BONDU (PC 33, SIG 4)

PS: Si cela ne vous suffit pas, peut-être étudierons nous la fonction INLINE\$. Histoire de patienter, sachez qu'elle admet 7 paramètres et est toujours d'origine HP...

-----  
SUPREM

Il y a quelques lundis, je me mis à écrire un programme dont la valeur technique n'a d'égale que sa longueur. Fort de ce morceau BASIC, je me fis un devoir de le documenter, afin de le rendre plus compréhensible au commun des mortels.

Peut-être un jour verrez-vous le résultat de ces élucubrations informatiques...

Quand je dis documenter, j'entends bien sur, ajouter çà et là quelques petites remarques, qui en aucun cas ne sauraient nuire à la qualité esthétique de l'oeuvre.

La taille du programme de base étant passée de 8 Ko à 14 Ko (que de place utilisée), il ne me restait plus qu'à faire 2 versions de mon travail. La première destinée uniquement et prochainement à vous, et devant donc satisfaire à tous les critères de compréhensibilité et de lisibilité et possédant donc un maximum d'observations. La seconde, elle beaucoup plus concrète, puisque servant, comme tout programme qui se doit à être utilisée en tant que telle.

Mais, un gros problème se posa à moi. En effet, comment faire pour supprimer toutes ces remarques qui maintenant me gênaient plus qu'autre chose ?

Un travail long et fastidieux eût consisté en la traque de celles-là tout au long de mes élucubrations dantesques. Mais quelle est la tâche d'un ordinateur, sinon bien entendu d'éviter à l'homme des tâches longues et fastidieuses ? Je m'attelai donc à la conception d'un programme dont l'unique but avoué était de supprimer (et hop) les remarques devenues inutiles lorsqu'il s'agissait de faire tourner quelques lignes (toujours dans le sens des aiguilles d'une montre).

Voici donc, livré, là, devant vos yeux ébahis, le résultat d'une cogitation rondement menée.

Laurent ISTRIA

-----  
TRI D'UNE TABLE PAR ORDRE CROISSANT  
UNE METHODE SIMPLE "LE TRI BULLE"

Cette méthode consiste à se déplacer dans la table de la gauche vers la droite, de manière à permuer les éléments non ordonnés, qui se suivent;

Exemple:

-----  
Soit le tableau suivant:

|               |     |     |     |     |
|---------------|-----|-----|-----|-----|
|               | 32  | 15  | 6   | 1   |
| 1ère étape -> | (32 | 15) | 6   | 1   |
| 2ème étape -> | 15  | (32 | 6)  | 1   |
| 3ème étape -> | 15  | 6   | (32 | 1)  |
| 4ème étape -> | (15 | 6)  | 1   | 32  |
| 5ème étape -> | 6   | (15 | 1)  | 32  |
| 6ème étape -> | 6   | 1   | (15 | 32) |
| 7ème étape -> | (6  | 1)  | 15  | 32  |
| 8ème étape -> | 1   | 6   | 15  | 32  |

Si au cours d'un déplacement dans la tablee, aucune permutation n'a lieu, alors la table est triée.

Jean-Claude FOURES

-----  
COMPTÉ BANCAIRE  
COMPTÉUR TÉLÉPHONIQUE

Je me permets de vous faire parvenir deux programmes sur HP71B qui me rendent personnellement beaucoup de services. J'espère qu'il en sera de même pour tous les membres du Club. Leurs caractéristiques sont les suivantes:

PROGRAMME COMPTÉ

But:

: Permet les opérations Débit/Crédit/Consultation.  
: A tout moment, on peut connaître les chèques qui  
: n'ont pas encore été encaissés par la banque,  
: ainsi que le solde disponible.

: Utilisation:

: Après [RUN], on a le choix entre cinq options:

: 1- CLST: consultation des chèques non encore  
: débités ou crédités. On se déplace dans le  
: fichier grâce aux touches [v] et [^]. Pour  
: signifier que l'on a reçu le relevé de banque, il  
: suffit de taper [END LINE] après l'affichage du  
: montant. Pour revenir au menu, taper [#] après  
: l'affichage du montant.

: 2- ou 3- Débit ou Crédit. Répondre à ce qui est  
: demandé. toutefois, si la date du chèque  
: correspond à la date du jour, il suffit de taper  
: [END LINE] pour la rentrer.

: 4- TT: Totaux. Fournit le solde du compte, ainsi  
: qu'entre parenthèses le nombre de chèques non  
: encore enregistrés par la banque.

: 5- I: Initialisation. A n'utiliser qu'avant une  
: première utilisation.

: Remarques:

: La construction est très classique. Les seuls  
: points un peu originaux sont:

: - Une consultation facilitée grâce à la routine  
: SUB CLAV(N) qui simule un éditeur d'écran.

: - Une réorganisation automatique du fichier dès  
: que celui-ci est rempli (Une mini-"Garbage  
: collection" en quelque sorte).

PROGRAMME TELE

: But:

: Permet de savoir combien va nous coûter une  
: communication téléphonique lorsque l'on téléphone  
: hors de sa circonscription.

: Utilisation:

: Après [RUN], entrer la période de taxation après  
: avoir consulté l'annuaire. Par exemple, pour  
: Fontainebleau, après 21h30: 48 secondes par taxe  
: de base.

: Ensuite, l'affichage fournit à droite l'heure  
: présente, au centre le prix facturé et à gauche  
: la durée de l'appel.

En fin de communication, appuyer sur une touche pour stopper les compteurs.

Remarques:

J'ai préféré conserver une entrée manuelle de la période de taxation plutôt que d'apprendre au HP la carte de France. A l'usage, cela se révèle beaucoup plus efficace, les circonscriptions téléphoniques ne correspondant pas toujours aux circonscriptions administratives.

Bien amicalement,  
Laurent SERAND (PPC 216)

CARACTERES ETHIOPiens

Ci-joint un programme de graphisme: transcription des chiffres arabes (1, 2, 3 etc.) en chiffres Ahmaric (ሀ, ለ etc.). L'Ahmaric est la langue officielle de l'Ethiopie. Il se trouve que ma famille habite l'Ethiopie et que j'avais dans un fond de tiroir tout l'alphabet de cette langue. L'alphabet comporte 36 lettres de bases qui chacune se subdivise en cinq syllabes. Leurs ressemblances étant trop grandes, je me suis limité au dessin des chiffres - 20 signes:

ሀ à ፩ représentent 1 à 10: les unités soit 10 signes.

ለ à ፲፱ représentent 20, 30, 40, ..., 100: les dizaines: 9 signes.

፳ représente 1000. Le zéro n'existe pas ainsi que les nombres négatifs.

35 s'écrit ሃ ፩ - Les barres signifient que  
          ^ ^ ce sont des chiffres.  
30 5

Cet alphabet et la langue qu'ils représentent sont encore très usités. L'alphabet complet est à votre entière disposition.

Amicalement,  
Xavier BILLE



*Serand*  
65

Programme "NAV" (Navigation aérienne)

```

100 DESTROY ALL @ STD @ DELAY .5,.125
110 Z$=CHR$(27)&'*b56'&CHR$(0)&CHR$(0)&CHR$(23)&CHR$(21)&CHR$(23)
    - Définition du symbole degré pour l'imprimante
130 IMAGE 15a,ddd.d,10a
140 IMAGE 13a,2x,sdd
150 IMAGE 16a,dd,a,x,zz,a
160 IMAGE 15a,ddd.d,a
170 IMAGE 13a,2x,sdd,3a
180 IMAGE 15a,dd," " ,zz,a
190 A=0 @ B=.4 ! Facteur de base
-----
210 INPUT 'Dir. vent : ';W
220 IF W>360 THEN 210
230 INPUT 'Force vent : ';F
240 IF F>50 THEN DISP 'Vent trop fort...' @ WAIT 2 @ GOTO 230
250 INPUT 'Distance : ';D
260 IF D=0 THEN DISP 'Fin...' @ WAIT 2 @ PUT '##43' @ END
270 IF D>100 THEN 250
280 INPUT 'Cap magn. : ';R1
290 IF R1>360 THEN 280
300 DISP 'Ecran / Imprimante'
310 A$=KEY$ @ IF A$=' ' OR A$<>'E' AND A$<>'I' THEN 310
320 IF A$='I' THEN RESTORE ID @ PRINTER IS :PRINTER
----- CALCULS -----
340 T=D*B
350 D1=B*(90/(2*F))
360 D2=D1*SIN(W-R1)
370 C1=D1*COS(W-R1)
380 C2=T+C1/60*T
390 R2=R1+D2 @ R2=IF(R2+.5*SGN(R2)) @ IF R2>360 THEN R2=R2-360
400 R2$=STR$(R2)
410 IF LEN(R2$)<3 THEN LET R2$='0'&R2$ @ GOTO 410
420 IF A$='I' THEN PRINTER IS :PRINTER @ GOSUB 'IMPR' ELSE GOSUB 'ECRAN'
430 GOTO 250
----- IMPRESSION -----
=====
450 'IMPR':
460 IF A=1 THEN 520
470 W$=STR$(W)
480 IF LEN(W$)<3 THEN LET W$='0'&W$ @ GOTO 480
490 PRINT 'Direction vent = ';W$;Z$
500 PRINT 'Force - = ';F;'kt'
510 FIX 1 @ PRINT 'F. de base (Fb) =';B @ PRINT '- - - - -' @ PRINT @ A=1
520 STD @ PRINT 'Distance = ';D;'km'
530 R1$=STR$(R1)
540 IF LEN(R1$)<3 THEN LET R1$='0'&R1$ @ GOTO 540
550 PRINT 'Cap magn. = ';R1$;Z$ @ PRINT
560 STD @ PRINT 'T.S.V = ';
570 IF T>60 THEN PRINT IP(T/60);'h';
580 PRINT IP(MOD(T,60)+.5);'"' @ PRINT

```

```

590 FIX 1 @ PRINT USING 130;'Derive (X) = ',D1,Z$
600 IF ABS(D2)<3 THEN PRINT 'Der. s/axe negligeeable' @ GOTO 620
610 PRINT USING 130;'Der. s/axe =',D2,Z$
620 IF ABS(IP(C1))<5 THEN PRINT 'Corr. Temps negligeeable' @ PRINT @ GOTO 640
630 STD @ PRINT USING 140;'Corr. Temps =',IP(C1),'"/"&"'
640 STD @ PRINT USING 150;'T. corrige =',IP(C2),'",IP(FP(C2)*60),'"'
650 PRINT 'Nouv. Cap magn. = ';R2$;Z$
660 PRINT @ PRINT '-----'
670 RETURN

```

----- E C R A N -----

```

=====
690 'ECRAN':
700 DELAY 8 @ Z$=CHR$(1)
710 DISP 'T.S.V = ';
720 IF T>60 THEN DISP IP(T/60);'h';
730 DISP IP(MOD(T,60)+.5);"'
740 FIX 1 @ DISP USING 160;'Dérive (X) = ',D1,Z$
750 IF ABS(D2)<3 THEN DISP 'Dér. s/axe négligeable' @ GOTO 770
760 DISP USING 160;'Dér. s/axe =',D2,Z$
770 IF ABS(IP(C1))<5 THEN DISP 'C. Temps négligeable' @ GOTO 790
780 STD @ DISP USING 170;'C. Temps =',IP(C1),'"/"&"'
790 STD @ DISP USING 180;'T. corrigé =',IP(C2),IP(FP(C2)*60),'"'
800 DISP 'Nouv. Cap magn. = ';R2$;Z$
810 DELAY .5,.125 @ Z$=KEY$ @ RETURN

```

\*\*\*\*\*

Programme "SUPREM" (Suppression des remarques dans un programme Basic)

```

=====
10 SUB SUPREM(A$)
  - A$ contient le nom du fichier sur lequel on travaille.
20 DIM C$(96)
  - C$ contiendra chacune des lignes à analyser.
30 TRANSFORM A$ INTO TEXT @ ASSIGN #1 TO A$ @ CREATE TEXT 'P'&A$(2) @ ASSIGN #2 TO 'P'&A$(2)
  - Il faut transformer le fichier source en fichier TEXT, et créer le fichier objet.
40 ON ERROR GOTO 'ERR'
  - Servira de test pour l'atteinte de la fin du fichier.
50 READ #1;C$ @ DISP C$(1,4)
  - Lecture de la ligne courante, et affichage de son numéro.
60 IF NOT (POS(C$,'!') OR POS(C$,' REM ')) THEN PRINT #2;C$ @ GOTO 50
  - Si la ligne courante ne contient ni '!' ni 'REM' alors il faut la passer entière dans le
  fichier objet.
70 IF C$(6,6)='!' OR C$(6,8)='REM' THEN 50
  - Si au contraire le premier caractère est soit un '!' soit un 'REM', alors la ligne entière
  est une remarque, et il convient de ne pas la passer dans le fichier objet.
80 FOR J=6 TO LEN(C$)
  - Si aucun des 2 premiers cas n'a résolu le problème, il faut faire une analyse syntaxique.
90 IF C$(J,J)='"' THEN J=POS(C$,'"',J+1) @ GOTO 120

```

```

- Si le caractère analysé est un ", il s'agit d'un message, et il faut sauter au " suivant.
100 IF C#[J,J]="'" THEN J=POS(C#,"'",J+1) @ GOTO 120
- Même remarque avec '.
110 IF C#[J,J]='!' OR C#[J,J+2]='REM' THEN C#=C#[1,J-2] @ GOTO 130
- En revanche, si on a atteint une remarque, Il faut éliminer la partie superflue,
120 NEXT J
- Et passer au caractère suivant.
130 IF C#[LEN(C#)]='@' THEN C#=C#[1,LEN(C#)-1]
- Mais, gros piège, s'il reste un @ terminal, il est nécessaire de l'oter,
140 PRINT #2;C# @ GOTO 50
- et de réécrire la ligne corrigée dans le fichier objet.

```

```

=====
150 'ERR': OFF ERROR @ ASSIGN #1 TO * @ ASSIGN #2 TO *
- Enfin, si la fin du fichier source est atteinte, on ferme les deux,
160 PURGE A# @ RENAME 'P'&A#[2] TO A# @ TRANSFORM A# INTO BASIC @ BEEP
- on efface le fichier source, et on renomme le fichier objet de son nom définitif.
170 END SUB

```

\*\*\*\*\*

Programme "TRIBULLE"

```

- Tri-Bulle
20 DESTROY ALL
30 INPUT "Dim. de la table=";N
40 FOR I=1 TO N
50 INPUT A(I)
60 NEXT I
70 PRINT "Table en entrée :"
80 GOSUB 'ECR'
90 FOR I=2 TO N
100 J=I @ F=0
110 IF F=0 THEN GOSUB 'REPEAT' ELSE 'EDL'
120 GOTO 110

```

```

=====
130 'EDL': NEXT I

```

```

=====
140 'FIN': PRINT "Table en sortie :"
150 GOSUB 'ecr' @ END

```

```

=====
160 'ECR':
170 FOR I=1 TO N
180 PRINT TAB(6*I);A(I);
190 NEXT I
200 PRINT
210 RETURN

```

```

=====
220 'REPEAT': IF J=1 THEN F=1 @ GOTO 'ret'
230 IF A(J)>=A(J-1) THEN F=1 @ GOTO 'ret'
240 K=A(J) @ A(J)=A(J-1) @ A(J-1)=K
250 J=J-1

```

```

=====
260 'RET': RETURN

```

```

*****
Programme "TELE" (Coût d'une communication téléphonique)

```

```

10 DESTROY ALL @ DELAY 0,0
20 INPUT "Periode de taxation: ";P
30 S=.75
40 T=IP(TIME)
50 DISP USING @C DIV 60,MOD(C,60),S,TIME$
60 IF KEYDOWN THEN END
70 U=IP(TIME-T) @ IF U<P THEN C=@+U @ GOTO 50 ELSE S=S+.75 @ Q=@+P @ GOTO 40
80 IMAGE 2Z":",2Z"<","3Z.DD">","8A

```

```

*****
Programme "COMPTE" (Gestion d'un compte bancaire)

```

```

10 DESTROY ALL @ FIX 2 @ DELAY 8,0 @ BEEP ON
20 DIM D1#[16],I#[20]
30 E#=CHR$(27)&"<"&"-" @ A#=CHR$(27)&">"&" "
40 D1#=" "%E##A%E##A$

```

```

=====
50 DEF FND2$(D#)=D#[1,2]#/"&D#[3,4]#/"&D#[5,6]

```

```

=====
60 ASSIGN #1 TO CHEQUE

```

```

=====
70 'CHOIX': CFLAG 2,3
80 INPUT "CSLT DBT CRD TT I : ",C#;C#
90 ON POS("12345F",C#)+1 GOTO 'CHOIX','CNCL','DBT','CRDT','TOTAL','CREAT','FIN'
100 END

```

```

=====
110 'FIN': ASSIGN #1 TO * @ END

```

```

=====
120 'CRDT': SFLAG 3 @ GOTO 'SAISIE'

```

```
=====
130 'DBT': SFLAG 2
=====
```

```
140 'SAISIE': INPUT "DATE: ",D1$,E$
150 IF E$="" THEN E$=DATE$(7,8)&DATE$(4,5)&DATE$(1,2)
160 LINPUT "Intitule: ";I$ @ I#=I$(1,20)
170 INPUT "Montat: Frs ";M
180 IF FLAG(3) THEN I$="C"&I$ ELSE I$="D"&I$ @ M=-M
190 CALL ECRIT(#1,E$,I$,M)
200 GOTO 'CHOIX'
```

```
=====
210 'CNGL': READ #1,0;L,T,X
220 J=1
230 IF X=0 THEN BEEP @ DISP "Fichier vide" @ GOTO 'CHOIX'
240 READ #1,J;D$,I$,M
250 IF M=0 THEN J=J+1 @ GOTO 320
260 IF I$(1,1)="D" THEN DISP "Debit du "; ELSE DISP "Credit du ";
270 DISP FND2$(D$)
280 DISP "Intitule: ";I$(2) @ DISP "Montant: ";M
290 CALL CLAV(J)
300 IF FLAG(4) THEN CFLAG 4 @ PRINT #1,J;"*", "*" ,0 @ J=J+1 @ X=X-1 @ PRINT #1,0;L,T,X
310 IF FLAG(0) THEN CFLAG 0 @ GOTO 'CHOIX'
320 IF J>=L THEN J=1
330 IF J<=0 THEN J=L-1
340 GOTO 230
350 END
```

```
=====
360 'CREAT': PURGE CHEQUE @ CREATE DATA CHEQUE,18,36
370 ASSIGN #1 TO CHEQUE @ PRINT #1,0;1,0,0 @ PRINT #1,1;0,0 @ ASSIGN #1 TO * @ END
```

```
=====
380 'TOTAL': READ #1,0;L,T,X
390 DISP "Solde: ";T; (" ;STR$(X)(1,LEN(STR$(X))-3);")
400 GOTO 'CHOIX'
```

```
=====
410 SUB ECRIT(#1,D$,I$,M)
420 READ #1,0;L,T,X
430 PRINT #1,L;D$,I$,M
440 L=L+1 @ T=T+M @ PRINT #1,0;L,T,X+1
450 IF L<15 THEN 520
460 BEEP
470 P1=1 @ P2=1
480 READ #1,P1;D$,I$,M
490 IF I$#"*" THEN PRINT #1,P2;D$,I$,M @ P2=P2+1
500 P1=P1+1
510 IF P1=L THEN PRINT #1,0;P2,T,X+1 ELSE GOTO 480
520 END SUB
=====
```

```

530 SUB CLAV(N)
540 P$=KEY$
550 IF P$="" THEN 540
560 IF P$="#38" THEN SFLAG 4
570 IF P$="*" THEN SFLAG 0
580 IF P$="#50" THEN N=N-1
590 IF P$="#51" THEN N=N+1
600 END SUB

```

Programme "AHMARIC" (caractères éthiopiens)

```

- Initialisation
10 DISP "ETHIOPIA (<=>);CHR$(148)&CHR$(149)&CHR$(150)&CHR$(151)&CHR$(152) @ WAIT .5
20 DESTROY A$ @ INPUT "Ara => Ahmaric";N
- Tests pour savoir si n est compatible pour une transformation
30 IF N>=10^6 OR N#IP(ABS(N)) THEN BEEP @ DISP "Pas de représentation possible" @ GOTO 20
35 IF N=0 THEN DISP " Le zéro n'existe pas" @ END
- Traitement des chiffres au dessus du millier
  traitement des chiffres au dessus de 2000
40 IF N>1999 THEN A$=FNC$(N DIV 10^3)
- A$ chaîne contenant la représentation en Ahmaric
  CHR$(147) représente le caractère des milliers
50 IF N>999 THEN A$=A$&CHR$(147)
- Traitement des chiffres entre 1 et 1000-1 et affichage des résultats
  @ fin du programme principal.
60 A$=A$&FNC$(RMD(N,10^3)) @ DISP ";"<>";A$;" @ END

```

```

=====
70 DEF FNC$(X) @ Y$="" @ IF X>199 THEN Y$=CHR$(127+X DIV 100)

```

```

=====
- On vient de traiter le cas où le chiffre des centaines > 2,
  X: passé par valeur = local. CHR$(146) est le caractère des centaines.
80 IF X>99 THEN Y$=Y$&CHR$(146) @ X=RMD(X,100)
- Traitement des dizaines (à partir du CHR$(137))
90 IF X>9 THEN Y$=Y$&CHR$(136+X DIV 10) @ X=RMD(X,10)
- Traitement des unités (à partir du CHR$(128))
100 IF X>0 THEN Y$=Y$&CHR$(X+127)
- Fin de la fonction traitement
110 FNC$=Y$ @ END DEF
- Centaines - Dizaines - Unités
  Sous-programme de création de caractères

```

```

=====
120 'CREATE': RESTORE @ FOR I=1 TO 25 @ C$=""
- Cumul dans C$ les codes du caractère
130 FOR J=1 TO 5 @ READ D @ C$=C$&CHR$(D) @ NEXT J
140 C$=C$&CHR$(0) @ CHARSET C$&CHARSET$ @ NEXT I
- LEN(C$)=6, un nouveau caractère est créé

```

150 GOTO 10

=====

160 DATA 192,38,57,9,6,204,51,18,44,200,6,25,233,6,2,12,5,254,4,12  
170 DATA 240,13,10,112,128,255,139,255,139,133,129,255,139,133,129,133,131,227,151,137  
180 DATA 133,255,131,255,133,185,197,199,203,167,197,163,175,179,197,129,135,137,233,159  
190 DATA 135,137,143,229,159,157,161,157,131,253,229,147,159,227,133,129,129,255,129,129  
200 DATA 191,201,201,197,191,197,163,191,163,197,133,227,211,203,231,133,227,211,219,167  
210 DATA 177,201,199,205,161,153,165,195,165,153,253,131,131,131,133,203,219,171,133  
220 DATA 177,201,197,203,177,0

\*\*\*\*\*

## LE COIN DES LHEX

Comme vous avez pu le remarquer, depuis le mois de février dernier, la rubrique Assembleur HP71B est devenue de plus en plus importante. Mais aujourd'hui un gros problème se pose !

La quasi-totalité des programmes BASIC figurant dans notre Journal préféré utilise des fonctions supplémentaires (KEYWAIT\$, INVERSE, REV\$ etc.) alors que beaucoup d'entre vous ne possèdent pas le module FORTH/ASSEMBLER et ne peuvent donc même pas utiliser ces logiciels.

Ceci est le but de cette nouvelle rubrique. Nous y pensions déjà: un petit essai avait d'ailleurs été fait avec un programme de Jean-Jacques Moreau au mois de mars dernier, mais par manque de place, l'idée avait été mise de côté. Vous avez été très nombreux durant la dernière réunion mensuelle à réclamer cette rubrique et à juste titre: la ROM F/A coûte 1700 Frs. Plus de très nombreuses heures d'apprentissage, sans compter l'acquisition des IDS !

Pour résoudre ce problème, nous vous proposerons donc dorénavant cette nouvelle rubrique ("Le coin des LHEX"), et cela à chaque parution de JPC. Je dis nous, mais n'oubliez surtout pas que ce "nous" veut dire "nous tous" (vous en faites partie) !

Dans cette rubrique, vous trouverez tous les fichiers LEX du numéro courant, listés sous forme hexadécimale et, pour introduire ces fichiers dans la mémoire de Titan, un petit programme BASIC (de base ! comme certains disent). Ce programme s'appelle "MAKELEX", il est la modification (très humble) d'un programme fait par des membres de PPC-Suède paru dans le nouveau journal américain CHHU. J'y ai apporté quelques modifications. Notamment rendre compatible la taille du LEX créé avec le nombre d'octets affichés durant son catalogue, permettre une introduction "implantable" du fichier HEX sans avoir à réintroduire toute la ligne courante en cas d'erreur et une vérification tenant compte de toutes les lignes

précédentes pour éviter de réintroduire plusieurs fois la même ligne.

Ce programme, relativement court vous sera donné tous les mois, ceci pour que les nouveaux adhérents (ils sont de plus en plus nombreux) ne soient pas lésés par rapport aux anciens.

Son utilisation est fort simple: faites [RUN], le programme vous demande alors la taille du fichier à créer. Introduisez la taille EXACTE donnée avec le listage hexa (EXACTE est la seule restriction !), puis entrez le fichier ligne par ligne sans omettre la somme de vérification (Checksum) à la fin de chaque ligne.

Son principe: on "poke" dans un fichier DATA créé en fonction du nombre d'octets introduit en début de programme la réplique exacte du fichier LEX. Enfin, on substitue l'en-tête du fichier DATA par celle d'un fichier LEX et le tour est joué.

Il vous faudra tout de même éteindre puis rallumer votre machine pour valider le LEX ainsi créé.

Exceptionnellement, dans ce numéro, vous trouverez ci-dessous un récapitulatif de tous les LEX parus dans JPC depuis février dernier. Ces LEX tiennent compte de la renumérotation des tokens et sont des versions optimisées.

Voilà ! Nous espérons vivement que cette rubrique excite encore plus chez vous l'envie de programmer avec de beaux outils (cf. MENU, REPEAT ou FORMAT), ou si vous n'êtes que de simples utilisateurs que vous trouviez dans JPC de beaux logiciels (cf FANTOME, DESSIN ou FORMAT).

Four le bureau,  
Michel MARTINET (SIG#2, P#12)

DESAL ID#71 718 octets

0123456789ABCDEF sm

000: 44543514C4020202 26  
 001: 802E009101727058 E4  
 002: 1A5001710506E200 7C  
 003: FB30000000000000 3F  
 004: 012100FF0019100F F9  
 005: A10BD100F5200500 92  
 006: 0F03025200FB1444 14  
 007: 2455644210714353 5E  
 008: 4422071445844230 A4  
 009: 7844514424072554 2B  
 00A: 4442501FF4118F83 E4  
 00B: DB0D60A871A51360 92  
 00C: 6137C2134135AD2D A7  
 00D: 681E81ED7C6BF2BF B4  
 00E: 2A0E109CF4811831 94  
 00F: 5A32172201C11495 9C  
 010: 7E071341CF1198D8 D3  
 011: 32F08D91FB080CF2 D1  
 012: 08F670B15F18FAA2 03  
 013: 51331464BFC70B14 9A  
 014: 4D309A0ABF1A88BF 05  
 015: 4BF4A4E59CAF4034 9A  
 016: 118F13DB0D23068A 9E  
 017: 63A15B5175137062 86  
 018: 2769FD68F1CB1158 83  
 019: 11C613324AC2B0FF 5C  
 01A: 8F84171071355901 E3  
 01B: C915991098508F06 DD  
 01C: 4A18F7B1818FE83B 44  
 01D: 18DC32F041113713 1F  
 01E: 5068FB3DB0AD1D88 93  
 01F: 1D31E20A3302E7D7 A6  
 020: CD4C1148DB8FC70B B1  
 021: 15700914D17153E0 6B  
 022: 71358DC32F04118F 5F  
 023: 83DB013606137135 0D  
 024: 1348F064A11CF137 D3  
 025: E28B37413506D6C6 93  
 026: BF2BF230F155717F A3  
 027: AF2D681ED7A4ECF4 3D  
 028: 3214ABCABF841711 DF  
 029: 59316117353E8DD4 3C  
 02A: 49007135071348DC AD  
 02B: 32F04117B007B207 55  
 02C: 3004248F13DB0AD1 F7  
 02D: D881D3102CD48214 E3  
 02E: B966001715FED4E4 D0  
 02F: C4BF0BF0A0C1CF15 FF  
 030: 17010772EF8DC32F 7D  
 031: 010F304BF100F020 1E  
 032: 0000000000000151 07  
 033: 00FD00B2000D9055 A7  
 034: 454E442F3705F4B4 B4  
 035: 54041FFF70001800 E0  
 036: 08F681F070901F17 F4  
 037: 8F2D41411618FB71 F3

038: F08F83DB025A8020 EC  
 039: 81C1B178F2D6D714 37  
 03A: 21311BE95F214213 CE  
 03B: 0CF4B1181AC214A7 BD  
 03C: E70D4159017054EB C0  
 03D: D84A80BD394508FA 5B  
 03E: B6308FEA2304908D 56  
 03F: B2E208DD97308F13 22  
 040: DB0D230A8B6F2D68 20  
 041: 0D00D442133C2131 01  
 042: AF01531135D1207F 14  
 043: 009680056FBD91FB 5E  
 044: 08F670B1401BF1A8 A3  
 045: 8BF4BF4018FAA251 06  
 046: 331464BFC70B14CC 1B  
 047: 3173B6A53D84228F 57  
 048: DF8E017F1361BB98 CC  
 049: F21448F322B15D9D 87  
 04A: 6C64691007F4F118 D3  
 04B: D7DAC9109D9108C4 E6  
 04C: D68FA90F0110130C AD  
 04D: F4D1AC215A08FB41 79  
 04E: 7114917116063EF1 B5  
 04F: 19135208FEB3B11B F7  
 050: B98F21461348DC32 75  
 051: F025101086000F71 62  
 052: 00000000000000E1 2D  
 053: 000FFB4549575149 47  
 054: 44542101FF001361 4D  
 055: 081371098F2C6004 09  
 056: 511191351181348D 95  
 057: 8ACA18F127006BDF C1  
 058: 25C0C000000F7100 56  
 059: 00000000000007100 3F  
 05A: 0F725546542C01FF 92  
 05B: 4118FE83B18DC32F 0A  
 05C: 0 3A

KEYWAIT ID#52 55 octets

0123456789ABCDEF sm

000: B454957514944502 BB  
 001: 802E000201727058 42  
 002: 3700025101000000 14  
 003: F710000000000000 BB  
 004: 0E1000FFB4549575 57  
 005: 14944542101FF001 B3  
 006: 361081371098F2C6 43  
 007: 0045111913511813 63  
 008: 48DBACA18F127006 9B  
 009: BDF 37

REVLEX ID#52 33 octets

0123456789ABCDEF sm

000: 255465C454850202 B1  
 001: 802E000201727058 38  
 002: 6400025C0C000000 4B

003: F710000000000000 F2  
 004: 071000F725546542 F0  
 005: C01FF4118FE83B18 90  
 006: DC32F0F 23

PKLEX ID#01 257 octets

0123456789ABCDEF sm

000: 05B4C45485020202 58  
 001: 802E000201727058 DF  
 002: 6020010F30400000 68  
 003: F020000000000000 04  
 004: 015100FD00B2000D F1  
 005: 9055454B442F3705 BE  
 006: F4B454041FFF7000 2D  
 007: 180008F681F07090 DE  
 008: 1F178F2D41411618 5A  
 009: FB71F08FB3DB025A 00  
 00A: 802081C1B178F2D6 65  
 00B: D71421311BE95F21 FD  
 00C: 42130CF4B1181AC2 0F  
 00D: 14A7E70D41590170 62  
 00E: 54EBD84A808D3945 47  
 00F: 08FAB6308FEA2304 4B  
 010: 908DB2E208DD9730 65  
 011: 8F13DB0D230A8B6F B2  
 012: 2D680D00D442133C BA  
 013: 2131AF01531135D1 37  
 014: 207F009680056F8D 53  
 015: 91FB08F670B1401B D6  
 016: F1A88BF4BF4018FA 90  
 017: A251331464BFC70B B7  
 018: 14CC3173B6A53D84 EF  
 019: 228FDF8E017F1361 AD  
 01A: BB98F21448F322B1 1D  
 01B: 5D9D6C64691007F4 96  
 01C: F118D7DAC9109D91 27  
 01D: 08C4D68FA90F0110 88  
 01E: 130CF4D1AC215A08 9D  
 01F: F841711491711606 18  
 020: 3EF119135208FE83 F2  
 021: B11BB98F21461348 CC  
 022: DC32F0F 5F

REPLEX ID#71 320 octets

0123456789ABCDEF sm

000: 255405C454850202 93  
 001: 802E001201727058 21  
 002: 5820017606000000 5F  
 003: F710000000000000 06  
 004: 0D5000FF255405C4 55  
 005: 14345442601FFBA9 BA  
 006: 001B11C114B11892 1C  
 007: E70962A014E96600 AA  
 008: CDCD58D11C8DB14B 2E  
 009: 18D91FB044443480 A5  
 00A: DF884468FB3DB08A CC

00B: 88534200008B64D8 01  
00C: 0C214F1081715541 CE  
00D: 1B8AEB08A8D06110 14  
00E: 8AB80017C9F1351C 0C  
00F: F076AA11371F495F 6A  
010: 2143135010C80C21 B2  
011: 088F83DB01371358 B9  
012: F534B110C137C213 25  
013: 58F83DB0137C2135 1E  
014: 1038F83DB0718F13 F1  
015: 710AC2D7D2109118 D2  
016: ABA136BF2BF2BF2A 3B  
017: B610B1BB98F2DB14 12  
018: 411311AD5DBE98B6 D3  
019: 65DB13411A1351CF D0  
01A: 11BD574BEBADA375 40  
01B: DE1351CD1471C113 1F  
01C: 7111E2137DA78BEC 28  
01D: 2136D87D1F704011 B7  
01E: BE35E27D0FDB1341 E9  
01F: 19D57F8EE91351CF 65  
020: 3420000D57110136 BB  
021: D711A8A3E3666F13 9E  
022: 7E98B692C91358A9 CB  
023: 001811C114E14D11 95  
024: 9E6E6109CDDC50E8 4A  
025: DD44901FB98F2147 36  
026: 135702EDA3401000 17  
027: EA1198F601B11CF1 49  
028: 18BF6BF6BF613411 06  
029: 9BF2BF230F15D68D 34  
02A: C32F0 7B

FILELEX ID#71 64 octets

0123456789ABCDEF sm

000: 6494C454C4548502 7B  
001: 802E002201727058 10  
002: 5800017707000000 5A  
003: F710000000000000 01  
004: 091000F96494C454 23  
005: F3701FF411136068 B1  
006: F59B9057097C908D C4  
007: A9390137068F77F9 1A  
008: 0AF0480B44BF4071 99  
009: 351CF1517071348D C3  
00A: C32F0 0A

KBEEP ID#71 38 octets

0123456789ABCDEF sm

000: B424545405020202 AA  
001: 802E002201727058 3F  
002: 1500017000000000 01  
003: FE0000000800001F B0  
004: F31B19614000DB10 07  
005: B0420348E300D732 A9  
006: 0208F04BE011BD70 A6

007: 0 D6  
ATTNLEX ID#71 68 octets  
0123456789ABCDEF sm

000: 144545E4C4548502 96  
001: 802E002201727058 2B  
002: C800017808000000 95  
003: F710000000000000 3C  
004: 0E1000D7144545E4 3B  
005: 801FF05000F20001 C9  
006: 37061F144F214A31 A3  
007: 1E96240CE15D0071 A2  
008: 358D84A808FA2C20 DE  
009: 0E1101EC00008DB2 6E  
00A: E20038DCF250F 29

PLOTLEX ID#5F 855 octets

0123456789ABCDEF sm

000: 05C4F445C4548502 A5  
001: 802E003201727058 41  
002: 2B600F510D000000 70  
003: F380000000000000 24  
004: 054400DD00C4400D 6B  
005: A103F300D520A440 D7  
006: 0D03087400DD30FC B5  
007: 400D8406D400D550 6A  
008: 6B300D0604D400D1 35  
009: 704D300DE7081400 E7  
00A: DB802B300D8909C3 99  
00B: 00D934F4C4F42510 BE  
00C: 9343594A55420744 D9  
00D: 25147530784F4D45 69  
00E: 4409C4142454C450 D2  
00F: 7C4449A25609C445 A1  
010: 950554707D4F4655 08  
011: 480D05C4F4454554 0A  
012: 259092544251475A 99  
013: 092554355445B092 01  
014: 5D4F46554C0B3554 47  
015: 45F42574D01FF8D1 31  
016: 0FA18D156818DD96 D7  
017: 3020311F96200017 4E  
018: A5486470018FA2C2 1C  
019: 0FEF590C00008D53 41  
01A: 0300372CF75CF591 C8  
01B: 8DE6A20038FAB630 C4  
01C: 8D271308DF3E207B 3D  
01D: 04E3B70018498F99 38  
01E: 1508D003508D3945 05  
01F: 08D303508F019E04 1D  
020: 3317F32300B24A2C B9  
021: 4709BED1AC8BF2EA 6F  
022: 21AF68FE14B1AC90 7F  
023: 40133B0008DA9390 AB  
024: 10A84694AD031D27 4E  
025: 35011A0484425302 A9

026: BF22080F680C510A 1E  
027: 9AE7086441854203 C6  
028: 038127D1011A25A0 70  
029: ESEC208740031035 E6  
02A: 6031C27333EAA605 9A  
02B: 008908589135892E F5  
02C: 488434731326B308 3A  
02D: 6C22061361B244F2 2B  
02E: 15641340794A80B4 52  
02F: 653175E219760570 7A  
030: 884A02480C02380C 67  
031: 12231FF206F1F7E4 66  
032: 275F007135147757 96  
033: F64B0742E681F072 E6  
034: 2E72BE10B72227AA 2F  
035: 007135147794F11B 8F  
036: 7CDEB75AC31D0753 6A  
037: F3111647070ED681 AC  
038: F07EDD7E6E10B7ED 3A  
039: 1758007135147750 C1  
03A: F6D3074BD681F072 7C  
03B: BD724E10B7B3E10C 0D  
03C: 7BA1725007135147 B1  
03D: 72DE11C756E73CE1 B4  
03E: 1B7A5E31D078BE7B 60  
03F: E1220106ED17D008 A3  
040: 750031216D9E1F7F C6  
041: 6F214F0A017DDF7A 9A  
042: EF8550914D01EBDF 78  
043: F47DFF7B6FD4EADF D7  
044: F46DFF7B5F44E9DF 0A  
045: F45DFF7B4F25E8DF 19  
046: F44DFF7B3FA458DF E5  
047: F64DFF73AE94E6DF 0D  
048: FF2DFF7B8E34E5DF 77  
049: FF1DFF7B9E3555DF D2  
04A: F61DFF737E84D2DF 47  
04B: F9DCFF75B07E4F84 CD  
04C: 50914D312178DD31 3C  
04D: 14671F41DFFEDCFF D6  
04E: 797C681F07280792 9A  
04F: F31057DAD776C8FE 20  
050: 83B1170157494E12 B3  
051: 17014381C17DB847 26  
052: AB0A1960560668D6 7F  
053: 6BE7BCFF87CFF783 29  
054: E157ACFF86CFF782 71  
055: EC4B7CFF72CFF342 2F  
056: A7F275706D010011 B3  
057: B2A7F215E61A388F 80  
058: 15C61A078F31F314 E7  
059: C8F73321E04D0831 A5  
05A: C0333200609C1F27 B4  
05B: BF21471351C41433 A9  
05C: 4500008A2D07E002 56  
05D: 6B30038D84A80061 E8  
05E: 371F0F7F21451D5F 8D  
05F: 0715571E508F1517 B3  
060: 1D51D980F526A825 22  
061: 50A0E27A82A0E15D C9

062: 7042032CFB8FAB81 57  
063: 154331FFD5D0E415 C3  
064: F596A1296561BF6F D5  
065: 69E2C0F6F6B62551 A8  
066: 17A5AD3332008DA9 B6  
067: 390175147D507135 DB  
068: 17413706147C9061 24  
069: F50BF2153717F15F 24  
06A: 7D527B0644005269 3C  
06B: 0E2080D51E5F7F15 D0  
06C: 771D0F0614713507 D3  
06D: 01F 37

DRIVELEX ID#71 277 octets

0123456789ABCDEF sm

000: 4425946554C45485 D4  
001: 802E004201727058 77  
002: E22001790A000000 3F  
003: F0200E3008600000 00  
004: 0BA000D110EC000D 7F  
005: D4494351424C4549 2A  
006: 0B54E41424C454A0 48  
007: 1FF10200120402C4 0D  
008: 568702C811054427 EC  
009: 96675627D20EAECF 76  
00A: F31D29614000D2A6 7A  
00B: E8A2400076100005 44  
00C: 044D2C454851FF00 F1  
00D: FF07135AC0B44821 04  
00E: 01E5100321007260 E3  
00F: D2A6E8A2C434802E E0  
010: 0BA674D2A6E4A223 C5  
011: 1007F00076303480 43  
012: 2E08A2E1D5D2A6E8 D1  
013: A691D915D38408F2 8A  
014: 12018DB4A8031F34 C6  
015: 36048F67B90D3CFB 5A  
016: F77F90484AC2B469 B7  
017: C3551371088FD979 EE  
018: 0137135C2D735179 FA  
019: 0A02515B59122317 E3  
01A: A1378BF0513758E3 91  
01B: 1938D39390331017 3C  
01C: 8DA939031C347E17 B8  
01D: 51432034E3000EA1 A6  
01E: 33D61537970FC137 34  
01F: 174679F201181351 F1  
020: 7FD01583018FC9E3 E3  
021: 0502B7722314C8FB A4  
022: EC20AEAB0DF8D624 BF  
023: 508D20F208D27130 3A  
024: 8F957508D00350F 01

CURLEX ID#71 90 octets

0123456789ABCDEF sm

000: 345525C454850202 A0

001: 802E005201727058 4A  
002: 9B00017000000000 2E  
003: FE0000000800001F AD  
004: F31B196140001103 70  
005: 1B09664FDBD5317F C5  
006: 8FC4631D9D75ED1F D3  
007: 788F2D231C28FFA0 24  
008: B11FE74F2AF214FB 47  
009: F4BCE0B64041F7B8 E8  
00A: F23103814908E0A6 2F  
00B: A149171AE0810A6A 29  
00C: 1491F788F28F7415 53  
00D: 182184001 27

FKEYLEX ID#71 81 octets

0123456789ABCDEF sm

000: 64B45495C4548502 74  
001: 802E005201727058 1E  
002: 6A00017B0B000000 41  
003: F710000000000000 EB  
004: 0E1000D764B45495 FD  
005: B01FFA6000E50008 8B  
006: F29D80D910834C10 5E  
007: 00D534444F2DAE6E 44  
008: 6808F8F803101101 C6  
009: B444F21481801524 78  
00A: B44550A4C1504808 F2  
00B: 08D84A808DD97308 ED  
00C: D39450F 25

DESLEX ID#71 281 octets

0123456789ABCDEF sm

000: 445435C454850202 A2  
001: 802E006201727058 53  
002: 7320017C0F000000 91  
003: F230000000000000 34  
004: 0CE000F31016000D 67  
005: 4200B000FF20DF00 39  
006: 0FF34F4E44525143 E5  
007: 545C0D94E4655425 EE  
008: 3554D0794E46542E C8  
009: 09051494E445F01F C4  
00A: FF30001400034C50 63  
00B: 001F401E27C20310 7B  
00C: 61E002E7E1031C41 67  
00D: E003E70108D84A80 C9  
00E: 8D303500314BBEC1 15  
00F: 49171CECEBAEDE01 6A  
010: 4111371351088FB3 2A  
011: DB0D68AA607BCF12 34  
012: 81358DC32F000136 A1  
013: 1081BEF3E2D015A0 C7  
014: 8FB13B11201188D9 FC  
015: 12F088B331361081 8E  
016: 371091358FC8CB01 EA  
017: 0ABF322B11228F32 DF

018: 2B112210317F8FC1 EC  
019: DB08F322B1123313 C1  
01A: 89E65112231709E6 90  
01B: 901226700608034D C5  
01C: 50001B442E28B680 07  
01D: 1A4A1E31D28B6801 60  
01E: A401E136C2C21361 FA  
01F: 4A11AAE5AF230196 C1  
020: 9C0A66A6D64FFAE7 78  
021: 0E62123AEB11310A 53  
022: AEB969A00E6E6A00 7F  
023: BEE0E661481128FB 88  
024: 13B1120119137118 1C  
025: 8D912F0 39

ENDUPLEX ID#71 252 octets

0123456789ABCDEF sm

000: 54E4445505C45485 C7  
001: 802E006201727058 78  
002: DF10017012100000 9A  
003: F920000002500000 93  
004: 0AB100FF00AC000D 7A  
005: C10E6100FB54E444 2D  
006: 55054201954E4445 9B  
007: 50511F3545142545 76  
008: 550542211FF31BF9 2D  
009: 613131CF961C1531 18  
00A: 118D73200A8FFF81 2A  
00B: 1D0E4003200A8FAB EC  
00C: B115DEDB10831FCB EE  
00D: FC463159C31DC8FC F3  
00E: 46314BB31DC8FAF5 4E  
00F: 311378D626208DD9 91  
010: 7308D394509FFFFD B7  
011: EFFF8F681F08FE44 FD  
012: A131D021840137FA 76  
013: 809FABB354137155 14  
014: 1208F7B181D23021 72  
015: 083200A10A8F13DB 1F  
016: 01371288B6A111A8 8E  
017: F14A118D84A808DD C0  
018: 4490D8D2310C8BA9 93  
019: 08DA114110111A8F C5  
01A: D791157D11013011 DF  
01B: 98F771B18D84A800 6A  
01C: 032808590003200A FF  
01D: 1331018FABB11137 EA  
01E: 136D58F064A11111 15  
01F: 315E114A31D09622 B4  
020: 1AE68F4058116156 11  
021: ED91348D7B181 E7

CMD16LEX ID#71 77 octets

0123456789ABCDEF sm

000: 34D4441363C45485 B8  
001: 802E007201727058 70

002: E900017313100000 98  
003: F710000000000000 3F  
004: 002000D934D44413 93  
005: 63311FF95000B500 FA  
006: 01F675F2143131AF AD  
007: 2A4E2380C315D517 DE  
008: 5A4E55F1371F085F 4D  
009: 20D1451740D57F1F BA  
00A: 679F215548D84A80 66  
00B: 8D303508D2BF30F 93

TYPELEX ID#02 67 octets

0123456789ABCDEF sm

000: 45950554C4548502 26  
001: 802E007201727058 DE  
002: AB00020030300000 C7  
003: F710000000000000 6E  
004: 071000F745950554 97  
005: 031FFC1115B6B148 E2  
006: 14AF22E30998680A A2  
007: 82503A4494C2230D D4  
008: B0EB044B017117F5 1D  
009: 31137C2137580307 EE  
00A: B0E8D832F0F 30

COMBARR ID#71 168 octets

0123456789ABCDEF sm

000: 34F4D42414252502 BA  
001: 802E007201727058 42  
002: 5510017415100000 71  
003: F020000000000000 0D  
004: 0A2000F90073000F 68  
005: 514252541734F4D4 CD  
006: 24511FF882275508 9A  
007: F499C08D612F0882 14  
008: 27F308F004D01118 6A  
009: FFB6C08F757E0AF9 43  
00A: AF7AF6113AFB1128 CA  
00B: FCA4C08F499C08D6 D1  
00C: 12F08FC8CB017F7C 9C  
00D: 40110119720001BC 1B  
00E: E8F439C08F363C08 1A  
00F: F757E08F004D0110 73  
010: 8FFB6C08F757E08F 01

011: C14D08FCA4C00110 03  
012: 01097D1011876101 76  
013: 10119268F674D040 60  
014: 06C2080DF89922AF 77  
015: A20322009B6318FF C0  
016: B6C08F4F6C04008D EB  
017: 91FB0 50

HMSLEX ID#71 279 octets

0123456789ABCDEF sm

000: 84D435C454850202 D3  
001: 802E008201727058 92  
002: 3320017619100000 F6  
003: F230000000000000 99  
004: 0E4000FB00E6000F 0B  
005: 61058000FF10FB00 93  
006: 0F784D435B261784 9A  
007: D435D271584D435B BE  
008: 138425911FF8822B 35  
009: 507880799170807B 29  
00A: 618F363C074E0840 1B  
00B: 6E2088228507F507 E9  
00C: 071775072418FA91 57  
00D: E056DB1184074A07 0C  
00E: B117EE0AF22E306A AB  
00F: F7CE8FA34C08FC14 D0  
010: D08F363C077B08FC 09  
011: A4C06A4081184071 E5  
012: 607E907BA0AF22D3 B0  
013: 163AF7E68FCA4C08 64  
014: FE04D074B08FCA4C 3C  
015: 08FC14D08F363C08 41  
016: F5F3D08F363C0870 B0  
017: 008218FA99C08D83 E3  
018: 2F08FDF8E08FFB6C 32  
019: 08F9D3D08F4F6C08 5A  
01A: F5F3D08F9D3D0948 F8  
01B: 80AC0550A4C8F363 04  
01C: C003AF22E301AF7E 4B  
01D: 6E6018FA34C08F00 14  
01E: 4D08F4F6C08FC14D BC  
01F: 08F004D071BF01AF B6

020: 22E306AF7E601137 DD  
021: 1081FBABF21577AF 48  
022: 71CF157712813511 FB  
023: 8011371FB98F2151 59  
024: 717FAF4151713517 CB  
025: F01 01

XERRLEX ID#71 90 octets

0123456789ABCDEF sm

000: 85542525C4548502 17  
001: 802E009201727058 DD  
002: BB00017A1A100000 06  
003: F710000000000000 AD  
004: 071000F785542525 BF  
005: A11FFB118FBC631D EF  
006: 68FC2530346E7F21 BE  
007: 37AF0AF214B328E3 CA  
008: D78F943B1D21C114 0D  
009: FCAD88F74030DAAC DC  
00A: 08A0E18B7B034863 1B  
00B: 00CAAF23108CABA4 E8  
00C: 50B44AF2AC6BF617 39  
00D: F8D612F0F 94

STKLEX ID#71 107 octets

0123456789ABCDEF sm

000: 3545B4C454850202 E9  
001: 802E009201727058 AF  
002: BD00017B1B100000 F8  
003: F710000000000000 9F  
004: 002000D935451434 60  
005: B4B11FFD90001900 D3  
006: 0048F681F08FBC63 F5  
007: 1440D08AC40E4D23 17  
008: 1018BA40DA1001F6 B8  
009: 75F2147D7D23068F EE  
00A: 943B1DBC21742217 B1  
00B: 41450D57FDB13511 16  
00C: OCCDB23AF230315D D1  
00D: 5175CC56F1F679F2 F2  
00E: D415908D84A808DE 21  
00F: 6A208D39450 02

```
0010 CALL MLEX @ SUB MLEX @ SFLAG -1 @ PURGE AH @ INPUT "Nb. d'octets: ";N @ LC OFF
0020 CREATE DATA AH,1,N-4 @ A=HTD(ADDR$("AH")) @ B=A @ GOSUB 130
0030 Q=1 @ X=0 @ INPUT "000: ",P$;A$ @ C$=A$ @ S=0 @ GOSUB 90
0040 Q=2 @ X=1 @ GOSUB 80 @ A$=A$&C$ @ A=A+37 @ N=N*2+37 @ Q=3 @ SFLAG 5 @ FOR X=2 TO N DIV 16-1
0050 GOSUB 80 @ C$=C$[5*FLAG(5)+1] @ POKE DTH$(A),C$ @ A=A+16-5*FLAG(5,0) @ NEXT X @ Q=4
0060 DISP DTH$(X)[3]; @ INPUT ": ",P$[1,MOD(N,16)];C$ @ GOSUB 90
0070 POKE DTH$(A),C$ @ POKE DTH$(B),A$ @ CFLAG -1 @ END
0080 DISP DTH$(X)[3]; @ INPUT ": ",P$;C$
0090 DISP DTH$(X)[3]; @ INPUT " sm ", "--";D$
0100 M=S @ FOR Z=1 TO LEN(C$) @ M=IP(NUM(C$[Z,2])#Z+M) @ NEXT Z
0110 IF D$=DTH$(MOD(M,256))[4] THEN GOSUB 130 @ S=M @ RETURN
0120 DISP "Erreur de somme" @ BEEP @ P$=C$ @ PDF @ DN @ BOTO 30,40,50,60
0130 P$="-----" @ RETURN
```

REPEAT ID#71 266 octets

0123456789ABCDEF sm

000: 2554055414450202 E1  
 001: 802E008040101000 37  
 002: 9120017C1C100000 69  
 003: F710000000200000 26  
 004: 03B100DB25540554 7E  
 005: 1445C11FF31C1961 0B  
 006: 4000248FDD410716 1E  
 007: 0482BF08B108F898 53  
 008: 108F851318F2C600 57  
 009: 4C18FEE010D91082 99  
 00A: 48FBA41082103731 30  
 00B: 08FD7521863EC313 AA  
 00C: 65ADB08BB108D127 55  
 00D: 00315C8FC4631350 10  
 00E: 00007492108AFOAF C4  
 00F: 21B989F21560B8EA CC  
 010: 0C8FBBC0E0118AE67 17  
 011: 9B0315C8FAF53128 29  
 012: 8F7FC001F064F2D0 4F  
 013: 14BCCD231E08FE6C 60  
 014: E02034F64F2E9135 FE  
 015: 15723030E06A0C49 6B  
 016: 0A2666FFA2647131 7C  
 017: 5C8F10631AF2B266 5B  
 018: C401BDF3E21F344F 5D  
 019: 215203079020278D DD  
 01A: F1B064F2153292CF 0D  
 01B: B14AD1AE80215329 FD  
 01C: 2CCA7CFE655F188F 81  
 01D: 3E22515C515A5912 3E  
 01E: 6015C52003140003 EF  
 01F: 40008F681F08FBC6 65  
 020: 315C0D2A0E8BA908 30  
 021: D029E088C1B989F2 4C  
 022: 15008DB84A808D394 0D  
 023: 508DE6A20 FE

MARGELEX ID#71 92 octets

0123456789ABCDEF sm

000: D414257454C45485 BF  
 001: 802E008040101000 15  
 002: DB00017D1D100000 80  
 003: F710000000200000 3D  
 004: 085000DBD4142574 18  
 005: 94E4D11FF31C1961 E1  
 006: 4000DB10B1F789F2 7E  
 007: 14B1EE74F14F9669 30  
 008: 08FA5CE011BD7007 F8  
 009: 4000B30008FB71F0 3D  
 00A: 8FBC6315708AC40D A4  
 00B: 031069EA40DA1F78 92  
 00C: 9F2CC1498D84A808 C7  
 00D: DE6A208D39450 2A

MENULEX ID#71 308 octets

0123456789ABCDEF sm

000: D454E455C4548502 8F  
 001: 802E009040101000 EC  
 002: D620017121200000 03  
 003: F710000000000000 AA  
 004: 0F1100F7D454E455 56  
 005: 121FFBFC6315C0D 9D  
 006: 2A6E8BA008D029E0 B1  
 007: 1B765F2146D71A29 6F  
 008: 6F1468AE01A265F 33  
 009: 14613414A31C2962 DB  
 00A: A218390C50163316 AD  
 00B: C8FB6AB04117FB03 E1  
 00C: 1028D39390161136 A1  
 00D: 13410A31C2AE531D EB  
 00E: 014A960A01619662 31  
 00F: F118A6E10896E991 21  
 010: 011FB98F21B045F2 3C  
 011: 7E108FFD15111111 62  
 012: A1358F327908D572 AF  
 013: 511577154717F16F 78  
 014: 271571154120018F A1  
 015: E92208DEE0108812 89  
 016: 71FE17F103D0E4A4 FB  
 017: EA4E4C07ADE17F12 E2  
 018: 311BF2F2AE61328F AA  
 019: B1DE01338FB1DE01 31  
 01A: 0410810B740002FF A7  
 01B: 071358FE0C101BB9 3F  
 01C: 8F21F045F2746FBF 60  
 01D: 8BB70779E7878FE C4  
 01E: 21208F3E320B2990 DD  
 01F: 33A30237102AD103 2B  
 020: AC3062640005DC11 CB  
 021: BA6E96EB011BAE2B 84  
 022: 6610810B65AF11BD 6B  
 023: AF4F4B669E67E11B 6D  
 024: DAF4F4AE658D7EFE E5  
 025: 8FE9220D011BAEAB 21  
 026: FB13B104AF6114BF 4A  
 027: A0DE01318FA0DE08 17  
 028: D912F08D84A80 29

PRIMLEX ID#71 148 octets

0123456789ABCDEF sm

000: 052594DAC4548502 9D  
 001: 802E009040101000 FA  
 002: C210017E1E100000 59  
 003: F710000000000000 00  
 004: 071000F7052594D4 00  
 005: E11FF8118FC1DB01 B0  
 006: 00AF210922719021 35  
 007: 78B0227580227F70 D5  
 008: 1118F322B1AF2D6A 24  
 009: FA048FBBC08F4BC 17

00A: E08F223B11189722 6A  
 00B: 4248F674D04A3247 9C  
 00C: C302276302470302 91  
 00D: 27A20247420267E1 07  
 00E: 0227B10267210629 8B  
 00F: F60501188D612F0A 4A  
 010: F280C010A1118F32 05  
 011: 2B1D6AF0DA11ACAB F7  
 012: FB13B11011101198 67  
 013: F8A4C08F4F6C0500 03  
 014: 071198D612F0F 6E

PRINTLEX ID#71 565 octets

0123456789ABCDEF sm

000: 052594E445C45485 34  
 001: 802E000140101000 5A  
 002: E640017721300000 B3  
 003: F170000000000000 60  
 004: 0BC100DB000D100D 86  
 005: 6106E100DD10DF10 63  
 006: 0FB2042200DF2003 93  
 007: 200D63034200D140 33  
 008: BB200DC4093300D3 F3  
 009: 50CA300D660EC300 33  
 00A: D72454C4C472724F 0C  
 00B: 4C44482334259275 1B  
 00C: 4353442A236464B2 DB  
 00D: 3C464C27D4F44454 66  
 00E: D2705542564E2305 9B  
 00F: CAF2F55E4445425C 56  
 010: 494E403775251405 05  
 011: 131FFB4466008542 3D  
 012: 0048F73321F0831D 9D  
 013: 031048D393900713 81  
 014: 5E6D7700007D534D 03  
 015: 1000C906D606D020 A3  
 016: 153101874008DB4A F5  
 017: 8014A310E9620003 9F  
 018: 8D303503354021B1 C6  
 019: 8FDFC2014B8DC25 3A  
 01A: 08FB8F303154966F 2E  
 01B: 11718FA2C200E1CF 58  
 01C: 1ECBF008D530308D 18  
 01D: B2E208FE7A204B17 15  
 01E: 5108D6CA208DE6A2 2F  
 01F: 08D074508D271301 1A  
 020: 8FFFA7FFF780F170 2A  
 021: A8FFFF9FFF745F49 A6  
 022: 070FE1F079EE1E01 54  
 023: 5FFFAAFF74006D2 2E  
 024: F1F659F2AE214D7A 6B  
 025: CE1D001001371091 A5  
 026: 351C12031B114D8F A2  
 027: 064A18408D7B1811 A0  
 028: 0FFFAFEFF74BF748 4E  
 029: E1C0CEEFF5EEFF7F 34  
 02A: 9F7F6E1A08DC2B90 2E  
 02B: 14FFF53FFF7460AF 6E

02C: 2A96976D7043103A 3F  
02D: 6ABFE1DE08F12DE0 B5  
02E: 3A5B162B60035A72 BA  
02F: 75BF15CA13606671 5D  
030: E99EFFEAFF736ED 55  
031: 031C4540E47D206C 3E  
032: 4E048F871F08F2EA E2  
033: 21044F0AF2328219 CC  
034: F2008D91FB0AE7DB B2  
035: 755F367B162C615C B4  
036: 6166815815310322 C7  
037: D4D1A6A14B161811 FE  
038: 0D5DEAEB14C16131 9E  
039: 1014C18E700007DA 54  
03A: 349BEFFC20613606 9C  
03B: 6B6D6F8F93EFF81E 7D  
03C: FF14ABF50A505D17 D0  
03D: 24DBB162C6732307 B4  
03E: 63636613C4743F17 B9  
03F: F1331011BE95F214 16  
040: 68BE1213331647A3 B7  
041: F111318FDF8E079 9E  
042: 0F13313331057B1F C2  
043: 6A3D35DFFB6DFF70 2F  
044: 3D4F07CCC4B1624644B 0A  
045: 047FBC4B1624644B 0A  
046: 3DFF05DFF750D411 47  
047: 71AC5B1623713347 91  
048: 29C5B162370334F 3A

DNKEYLEX ID#71 195 octets

0123456789ABCDEF sm

000: F4E4B45495C45485 67  
001: 802E001140101000 94  
002: B810017F10200000 3F  
003: F020000000000000 DB  
004: 0A3000F3108C000F DD  
005: F3455253544F475E B3  
006: 4F1F34552535B454 7E  
007: 9535021FF00A82B0 15  
008: 6808FB00802F0C4C 45  
009: 4AF38AC606D40D82 74  
00A: 77E505802E69302B 09  
00B: 7F405802E67202A7 0A  
00C: 0405802E65102B71 86  
00D: 305412E6300B07B0 5F  
00E: 7B07B0720A82A0E8 73  
00F: 00B0B0AFB1328D91 56  
010: 2F0D4C40C5BFC401 E0

011: 0013610A1371088F E3  
012: D4D008FEE010AF34 73  
013: C2AE48F3E320F294 79  
014: 003A4023B4033C40 DB  
015: 2AD403AE4000AFB1 4F  
016: 101311128D912F0B EE  
017: 07B07B07B07B07B0 B0  
018: 72068DF2E64FF2E6 9F  
019: BEF2E62EF2E69DF2 E8  
01A: E60DF2E67CF E5

INDATLEX ID#71 147 octets

0123456789ABCDEF sm

000: 94E4441445C45485 CB  
001: 802E002140101000 FF  
002: A210017232300000 2E  
003: F710000000000000 D5  
004: 0C1000FD94E44414 03  
005: 455442231FF00137 78  
006: 10913610A7040B1C 9C  
007: 34414455A02A3022 19  
008: 2B1E3A4A4B1C3F2B 8F  
009: 1E3D4D4B1C3F2B1E 30  
00A: 31414B1C322FF071 C8  
00B: 358FE0C108FFD151 72  
00C: 8F99C418FA3A8111 15  
00D: A1361191358508FB D6  
00E: D58117F17115FB17 8A  
00F: B1331011331CFB12 2C  
010: 81231F28F234B1DA FE  
011: 8F834B129B92B92B 93  
012: 92B92B92B92D6243 17  
013: 1F22015578F064A1 39  
014: 8408D7B181F 21

FORMALEX ID#71 454 octets

0123456789ABCDEF sm

000: 64F425D414C45485 27  
001: 802E002140101000 5B  
002: 0930017226200000 A1  
003: FB30000000000000 64  
004: 066100F110EA200F DE  
005: 020EC100F130E110 3A  
006: 0F2407D000FD3454 7E  
007: E44554254222B345 A8  
008: 43555255432D64F4 1D  
009: 25D414454242D255 8D

00A: 4445534544252B35 D9  
00B: 0514345442621FF0 FD  
00C: 48F83DB01368F534 92  
00D: B1137C2109134135 28  
00E: D38508418423102A 27  
00F: E5CC4E2851CC1811 B3  
010: 4E965F08708EE785 3F  
011: 05808408521C114D 28  
012: 51D86011861C0862 95  
013: 70171CF018118FBC 96  
014: 631440D017F13713 EA  
015: 51098F064A173004 05  
016: B231025908F40581 5F  
017: CC56F01411714F11 C7  
018: 98FB14B11368408F 62  
019: 064A18D7B1818D91 5E  
01A: FB0842274508F13D ED  
01B: B0137135C21098F0 ED  
01C: 64A1118C642D8A88 3F  
01D: 18BE31E2AD0DA81C EF  
01E: 81C7B7F8508F7B18 9F  
01F: 18FE83B18DC32F08 59  
020: FBC6315598A80917 7D  
021: F100018DD4490842 33  
022: 27ADF769E119110E F9  
023: 244EE24FD10ADA13 DF  
024: 416F41D1361B495F 14  
025: 21328BEFB119AF0D F4  
026: A137135EA81C1188 79  
027: B6348A2A3E48AB33 03  
028: EAF8DB1088FE6CE0 66  
029: 20108D8E57320110 34  
02A: 11BE2CD751011A13 FE  
02B: 56CCE31528D39390 46  
02C: D73102662014B171 19  
02D: 1481619661FD4580 65  
02E: 14C161CC57FC8BAF A2  
02F: AD018422730FD88F 6E  
030: 83DB0D6AD0DA81C1 A9  
031: 37135C2C2D78BA97 1A  
032: E9E91353F02F312E FD  
033: 2C2A3B3920A14896 B7  
034: 261850CD4801715C 0A  
035: E1104241108A0D12 5C  
036: 71C1148BF68F6962 63  
037: 710D52F4B18709CC 3E  
038: C10051C1188A160E AD  
039: 5E5D4CCDB1351CFB 50  
03A: FB13B1AF68D832F0 EF  
03B: F 35

ASSOCIATION REGIE PAR LA LOI DE 1901, ENREGISTREE  
A PARIS LE 2 DECEMBRE 1982 SOUS LE NUMERO 82/3240

BULLETIN D'ADHESION

NOM \_\_\_\_\_  
 PRENOM \_\_\_\_\_ DATE DE NAISSANCE \_\_\_\_/\_\_\_\_/\_\_\_\_  
 ADRESSE \_\_\_\_\_  
 \_\_\_\_\_  
 \_\_\_\_\_  
 COMMUNE \_\_\_\_\_  
 CODE POSTAL \_\_\_\_\_ PAYS \_\_\_\_\_  
 TELEPHONE DOMICILE \_\_\_\_/\_\_\_\_/\_\_\_\_ BUREAU \_\_\_\_/\_\_\_\_/\_\_\_\_

PROFESSION \_\_\_\_\_  
 INTERETS \_\_\_\_\_

MATERIEL HP EN VOTRE POSSESSION \_\_\_\_\_  
 \_\_\_\_\_

AUTRE MATERIEL MICRO-INFORMATIQUE \_\_\_\_\_  
 \_\_\_\_\_

COMMENT AVEZ-VOUS CONNU PPC PARIS CHAPTER ?  
 PUBLICITE \_\_\_\_\_ MAGAZINE \_\_\_\_\_  
 AUTRE CLUB \_\_\_\_\_ HP \_\_\_\_\_  
 RELATIONS, MEMBRES DU CLUB, AUTRES \_\_\_\_\_

QUE RECHERCHEZ-VOUS AU SEIN DU PPC PARIS CHAPTER ? \_\_\_\_\_  
 \_\_\_\_\_  
 \_\_\_\_\_

Je souhaite adhérer au club PPC PARIS CHAPTER conformément aux statuts de l'association. Au mieux de ma connaissance, je déclare avoir le droit de fournir tous les programmes et informations que je vous enverrai (sans enfreindre des obligations de secret à l'égard d'autres personnes ou organismes) pour publication dans le journal de liaison, sans obligations ni responsabilité d'aucune sorte (en cas d'utilisation frauduleuse) de la part des dirigeants du PPC PARIS CHAPTER.

DATE \_\_\_\_/\_\_\_\_/\_\_\_\_  
 SIGNATURE, PRECEDEE DE LA MENTION "LU ET APPROUVE" \_\_\_\_\_

LE MONTANT DE LA COTISATION AU PPC PARIS CHAPTER S'ELEVE A 300.00 FF.  
 ETUDIANTS: 250.00 FF. (JUSTIFICATIF INDISPENSABLE)  
 PAIEMENT EFFECTUE LE \_\_\_\_/\_\_\_\_/\_\_\_\_ A L'ORDRE DE PPC PARIS CHAPTER.  
 PAR  CHEQUE BANCAIRE N° \_\_\_\_\_ BANQUE \_\_\_\_\_  
 CHEQUE POSTAL 3 VOLETS N° \_\_\_\_\_  
 MANDAT LETTRE

EVENTUELLEMENT: JE M'ABONNE A COMPTER DU \_\_\_\_/\_\_\_\_/\_\_\_\_  
 JOINDRE A VOTRE INSCRIPTION UNE PHOTO D'IDENTITE ET UNE ENVELOPPE TIMBREE A VOTRE ADRESSE.

VEUILLEZ ENVOYER TOUTE CORRESPONDANCE A:  
 MR PHILIPPE GUEZ, 56 RUE J.J. ROUSSEAU, 75001 PARIS (FRANCE)

Le journal J.P.C. est un bulletin de liaison entre les membres de l'association loi de 1901 PPC.P.C.. Le club est éditeur du J.P.C. et son siège est au 56 rue Jean-Jacques Rousseau - 75001 PARIS.

Directeur de la publication: Philippe GUEZ  
Numéro d'ISSN: 0762 - 381X