

FEUILLET N°2

ILS SONT VRAIMENT
HORDUS D'ORIC

30 janvier 87



SOMMAIRE : LE TOUR DU MONDE EN 80 JOURS — AVIS IMPORTANT —
48K ? — LA SAUVEGARDE — 3 P.G.M. ... — LE
COURRIER DES LECTEURS — PROBLEMES SUR ORIC

Le tour du monde en 80 jours.

Editeur : NO MAN'S LAND

Ecrit par 'DUC' (vous connaissez, celui qui a fait DONJUAN).

Le scénario est inspiré du célèbre roman de Jules VERNE, et vous propose d'effectuer le tour du monde en 80 jours tel PHILÉAS FOG.

A vous de savoir gérer vos ressources financières pour acquérir billets et matériels nécessaires au bon déroulement de chaque étape. Un passage à la banque avant le départ vous permettra ensuite de faire vos emplettes pour remplir vos bagages d'objets qui, s'il vous semblent hétéroclites, n'en ont pas moins leur intérêt par la suite.

Néanmoins, vous ne pourrez pas vous charger de trop. Gardez aussi suffisamment d'argent de poche pour monnayer de menus services pendant la traversée.

Orient-Express, paquebots, cheval pour passer l'Himalaya, etc, vous utiliserez tous les moyens de transport et

Aventure et Stratégie...

rencontrerez des gens intéressants, mais aussi des bandits et pirate de tout poil. A vous de savoir vous comporter !

Tout au long du chemin, vous serez guidé par le carnet de votre prédécesseur qui avait pris soin de porter des 'estimations' en nombre de jours, pour ses différentes étapes. Votre temps personnel sera aussi affiché sur l'écran, superposé à la carte montrant votre position.

doublé d'un graphisme sommaire, mais non désagréable et illustré de quelques messages sonores imitant les bruits du train ou du bateau, ce programme est attrayant surtout par les situations qu'il propose. De plus il a été conçu de sorte de pouvoir être utilisé plusieurs fois sans perdre d'intérêt, en générant des intrigues différentes.

Commencez le voyage par un crochet chez votre diffuseur INNELEC et
..bonne route !

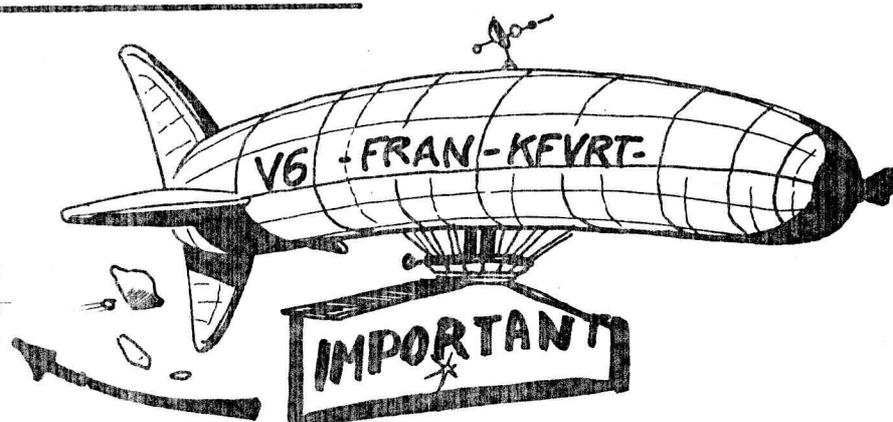
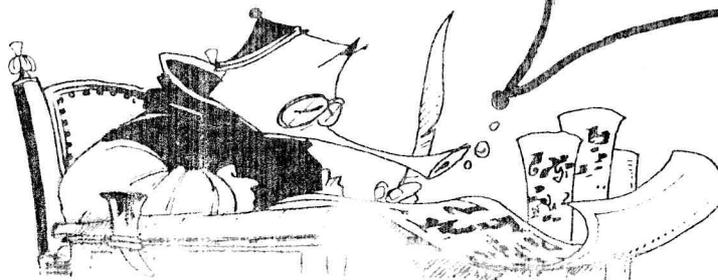
Je me sais pas si vous êtes de mon avis, mais pour moi < feuillet > cela fait un peu banal.

Alors voici mon idée : chacun d'entre vous me remet sur papier un titre convenable pour ce journal et dès que je possède toutes les propositions j'agirai de la manière retenue par la majorité.

A vous donc de choisir la façon de procéder, soit :

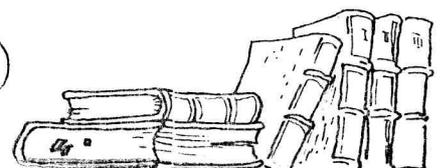
- 1 C'est ORIC qui choisira au hasard le titre. (les résultats seront faits sur imprimante et paraîtront pour éviter toutes triches)
- 2 Le meilleur titre sera élu avec la participation de tous (vote)
- 3 Ou alors je choisi personnellement le titre.

A.S.



Sur le papier que vous me rendrez sera inscrit votre prénom, votre idée pour le titre ??? ainsi que la façon procéder 1, 2 ou 3.

Alors si vous participez n'oubliez pas de me faire parvenir le papier avant le 10 février au plus tard.



- INFORMATIONS -

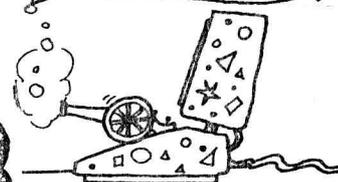
48K?

48K c'est ma capacité de mémorisation!

-Que veut dire 48K ?

Un caractère du clavier aussi appelé caractère ASCII, ou mieux code ASCII (se prononce comme à SKI), est reconnu par l'ordinateur par un Byte : Par exemple le A = 00100101.

Chaque caractère entré dans la mémoire d'ORTIC occupant 8 BITS ou 1 OCTET, ou encore BYTE signifie que après avoir entré 48000 caractères le microprocesseur est saturé et doit être vidé sur une cassette ou disquette.



R.S.

⓪ Sauver un programme c'est simple !!! ⓪

Par quel principe se sauvent les programmes et à quel vitesse ça va ?

Comme cela a été vu dans le numéro précédent, l'ordinateur ou plutôt le microprocesseur ne comprend que le binaire. Donc l'enregistrement se fera simplement par une suite de 1 et de 0. C'est cette suite qui fait un bruit désagréable lorsqu'on retire la fiche du lecteur.

Ces 1 et 0 sont appelés aussi BITS ; attention ne pas confondre BIT et BYTE, car un Byte égal huit Bits bien qu'il se prononce à peu près de la même façon.

Vous aurez remarqué qu'il y a deux vitesses de sauvegarde :

- une rapide > CSAVE""
- une lente > CSAVE"" , S

Tout est question de BAUDS, 2400 BAUDS pour la rapide et 300 pour la lente. Baud cela veut simplement dire 1 BIT par seconde ; sur une bande magnétique un BIT occupe environ 0,03 mm, on comprend pourquoi les bandes doivent être de bonne qualité pour ne perdre aucun bit.

R.S

★ 3 programmes faciles ★

★ à comprendre.

```
10 REM *** SONG FOR CHILDREN ***
20 FORN=1TOS0
30 READA,T,W
45 PRINTA,T,W
48 W=W*1.4
50 MUSIC1,K,T,0
60 PLAY1,0,1,5000
70 WAIT
80 NEXT
90 RESTORE WAIT300:GOTO20
100 DATA2,10,12,3,3,37,3,2,12,3,3,37,3,5,12
110 DATA3,6,37,3,5,12,3,3,37,3,2,12,3,3,50
120 DATA 3,3,50,3,3,87
130 DATA 2,10,12,3,3,37,3,2,12,3,3,37,3,5,12
140 DATA3,6,37,3,5,12,3,3,37,3,2,12,3,3,50
150 DATA3,3,50,3,3,87
160 DATA3,6,12,3,5,37,3,4,12,3,5,37,3,5,12
170 DATA3,6,50,3,5,37,3,5,12,3,6,37,3,5,12
180 DATA3,6,37,3,10,12,4,3,50,3,10,37,3,10,12
190 DATA3,11,37,3,10,12,3,8,37,3,6,12,3,5,37
200 DATA3,3,12,3,2,37,3,2,12,3,3,50,3,3,50
210 DATA3,3,87
```

⓪

```
1 REM *****
2 REM = ELLIPSES =
3 REM =
4 REM = R.S.87 =
5 REM *****
6 ?
7 ?
8 ?
9 ?
10 HIRES
15 FORP=1TOS
20 A=INT(RND(1)*50):B=INT(RND(1)*50)
30 FORT=1TOS200
40 CURSET120+COS(T)*B,100+SIN(T)*A,1
50 NEXTT
60 NEXTP
```

⓪

Etudiez-les et vous pourrez en faire de tels

```
10 READA,B,C:IF C=-1 THEN END
20 MUSIC1,A,B,C:PLAY1,0,1,11000
30 WAIT140:GOTO10
50 DATA3,10,2,4,1,4,4,3,2,4,5,3,4,7,1,4,5,2,4,3,4,3,12,2,3,8,4,3,10,1
60 DATA3,12,1,4,1,4,3,10,2,3,10,3,3,3,1,3,10,2,3,12,4,3,9,2,3,5,4,3,10,2
70 DATA4,1,4,4,3,2,4,5,3,4,7,1,4,5,2,4,3,4,3,12,2,3,8,4,3,10,1,3,12,1
80 DATA4,1,3,3,12,1,3,10,2,3,2,3,2,2,3,3,2,3,10,4,3,10,2,3,10,6,0,0,-1
```

⓪

★

* COURRIER des Lecteurs... *

Question de Sylviane :

* Pourrais-tu expliquer les avantages de CHR\$ et comment on peut s'en servir ?

CHR\$ est une fonction qui permet d'avoir le caractère correspondant à la valeur du code ASCII donné. La valeur de ces codes doit être comprise entre 0 et 255 sinon un message d'Illegal Quantity Error est retourné par l'ordinateur :

Cf. Table des codes ASCII du livre par Ex. PRINT CHR\$(66) donnera un B.

Déjà avec cela on voit l'avantage pour coder en chiffre un mot.

Mais des attributs permettent aussi d'autres avantages :

```
PRINT CHR$(7) .... sonnerie clavier
PRINT CHR$(17) .... curseur U ou I
PRINT CHR$(20) .... mode Maj. / Min
```

Donc grâce à CHR\$ on peut obtenir en programme des fonctions directes.

```
CHR$(7) = CTRL G
```

```
CHR$(17) = CTRL Q
```

```
CHR$(20) = CTRL T
```

..... etc.

Question de Patrice :

* Les DATAS à quoi servent-ils ?
comment s'en servir ?
et dans quels PGMS. ?

- Les données que les instructions DATA permettent d'introduire au programme sont des constantes. Elles peuvent être soit des nombres soit des mots: DATA 1, 2, 3, 4, 66, 555.....

```
DATA JANVIER, MAI, ORIC.....
```

Notez que les données sont séparées par des virgules afin que l'ordinateur les distingue.

- Il y a une foule de moyen de s'en servir et ce pour tous les pgms. Pour s'en servir il faut connaître les 2 mot du Basic qui l'accompagne : READ et RESTORE. Ces dernières sont aisées à comprendre READ lit les DATA et RESTORE remet la lecture donc READ à zéro.

```
Ex. : 10 FOR R = 1 TO 5
      20 READ A : PRINT A
      30 NEXT R
      40 DATA 50, 30, 20, 10, 40
      50 RESTORE = GOTO 10
```

Questions de F. et M. Aesloos :

* Serait-il possible de faire le permis de conduire de la Belgique ?
* Comment se fait-il que Trick Shoot ne soit pas dans la liste ?

Pour la deuxième question, c'est un simple oubli de ma part et dans le prochain ajout il s'y trouvera.

A la première la réponse est oui, car avec un peu d'imagination tout est possible avec Oric. Ce n'est pas par chauvinisme, mais ORIC permet, grâce à sa facilité de programmation, de créer ce projet. Donc en n'ayant un but pour la création du logiciel, ainsi qu'un peu d'expérience l'on peut déjà réaliser un beau programme.

Si dans un temps avenir tu veux réaliser ce projet, tout les renseignements nécessaires pourront t'être donnés.

De plus cela pourrait-être d'un grand intérêt pour tout le monde.

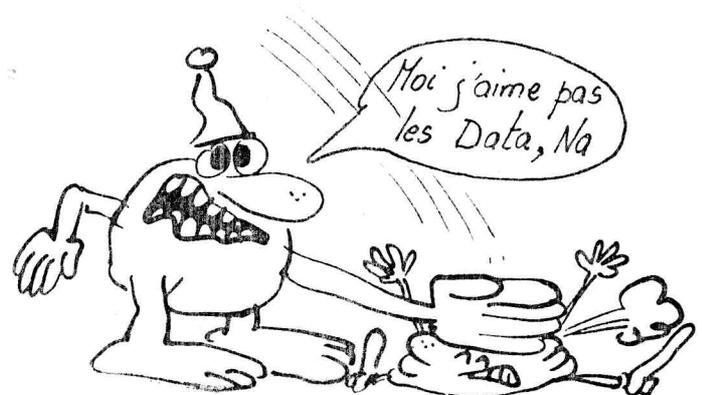
La ligne 20 lit A et l'imprime ce 5 fois grâce à la boucle R.

La ligne 40 stocke les DATA

La ligne 50 permet de boucler le programme à condition d'introduire un RESTORE, essayer sans (50 GOTO 10)

Et bien un 'OUT OF DATA ERROR' est retourné par l'ordinateur ce qui est juste puisque qu'ORIC ne possède plus données après 40 et donc s'arrête.

Regarde les petits programmes de ce numéro, 2 d'entre eux utilisent les DATA. Voilà si tu as un problème pour une utilisation particulière, expose le car le DATA est un domaine très vaste.



..Courrier suite..

Question de Vinciane :

* A quoi sert la fonction CHAR ?

Cette fonction s'emploie en mode graphique et elle sert à afficher à l'endroit où se trouve le curseur un caractère de code spécifié. Si <jeu> = 0, le caractère est pris dans le jeu standard et si <jeu> = 1 c'est dans le jeu semi-graphique qui est utilisé.

Syntaxe de la fonction:

CHAR<Code Ascii>, <Jeu>, C

- Le seul moyen d'effacer un caractère est de le réécrire au même endroit

en donnant à C la valeur 0 ou 2

- Code Ascii => se reporter au tableau des codes.

```
Ex. : 10 HIRES
      20 CURSET100,100,0
      30 CHAR65,0,1
      40 END
```

Question de Christophe :

* Je voudrais savoir à quoi sert la touche FUNCT du clavier ?

* A quoi sert la fonction MID\$?

- La touche FUNCT qui est une touche de fonction ne sert à rien à première vue mais c'est à l'utilisateur de la rendre utile dans ses programmes. Le petit programme qui suit te montrera une façon de l'utiliser, donc à toi de l'utiliser.

```
10 CLS:PRINTCHR$(17)CHR$(4)
15 A$="AUCLINE TOUCHE"
20 A=PEEK(#209)
25 IF A$=165 THEN PING:A$="<FUNCT>"
30 PRINTA$,9;CHR$(27)"JOUIS APPLI
YEZ SUR "A$
40 GOTO15
```

- MID\$ syntaxe MID\$(A\$,N,M) est une fonction très puissante qui permet de retirer une partie d'une chaîne de caractère : La chaîne retournée contiendra M caractères de A\$ en commençant par le Nième.

```
10 A$="IMPRIMANTE"
20 PRINTMID$(A$,3,5)
30 END
```



Ecrivez votre question ici :

C OQ ORIC OH!

LES BONNES ADRESSES DE L'ORIC

Pierre Chicourrat

Voici quelques adresses utiles sur Oric 1 et Atmos. Lorsque deux d'entre elles sont précisées, il s'agit d'un pointeur en page zéro. La valeur contenue dans ces deux octets représente donc une adresse. Ces adresses sont toutes écrites en hexadécimal.

#E9 EA: adresse du prochain caractère à traiter par l'interpréteur. Peut servir à détourner l'exécution d'un programme. Par exemple

DOKE #E9,#500 équivaut à RUN, mais sans réinitialisation des variables. On peut aussi faire un GOTO en milieu de ligne. Entrez par exemple le programme suivant:

```
10 DOKE #E9,#514
20 PING:ZAP
```

puis tapez RUN. Seul le ZAP sera exécuté, car #514 est l'adresse des deux points qui précèdent ce ZAP.

#A8 A9: contiennent le numéro de ligne en cours d'exécution (ce qui est affiché en mode TRON). Si A9 contient #FF, l'Oric se trouve en mode direct. De cette

façon, on peut taper en mode direct POKE #A9,1:GETRS, et le message ?ILLEGAL DIRECT ERROR ne sera pas généré.

#AC,AD: sauvegarde du contenu de #E9 EA en cas de BREAK ou de END, pour permettre un CONT. Idem pour les adresses AA et AB.

De 35 à 83: tampon clavier. Tout caractère tapé y est stocké (ainsi que lors d'un INPUT ou d'un GET). C'est ce principe qui explique que l'Oric ne se base que sur ce qui est frappé au clavier, et non sur ce qui est affiché à l'écran (cf. les CTRL L suivis de RETURN, et qui génèrent un SYNTAX ERROR, eu égard à ce qui a été tapé précédemment).

219: abscisse du curseur HIRES

21A: ordonnée du curseur HIRES



213: motif binaire du DRAW

2F7 (sur Oric 1 seulement): un OU logique est effectué entre le code ASCII de tout caractère à afficher et le contenu de cette adresse, le résultat étant ensuite envoyé.

Les Problèmes d'ORIC

Cette nouvelle rubrique a pour but de permettre à chacun de raisonner sur un petit programme et d'en comprendre son fonctionnement. Ensuite il vous sera demandé de résoudre le problème exposé.

La meilleure solution paraîtra dans le feuillet suivant avec le nom de son auteur tandis que les erreurs éventuelles des autres y seront critiquées, afin de leur permettre de ne plus les refaire.

Les solutions devront être rentrées au max 10 jours après la réception du feuillet.

Les atouts de la meilleure solution :
remettre un programme avec une bonne syntaxe.

programme toujours le plus court possible.

programme tournant sans problème.

Voilà.

Alors à vos plumes et clavier...

(dites-vous toujours qu'il y a plusieurs façons de résoudre un programme)

```
0 REM =====
1 REM = PROGRAMME-EXERCICE No 1 =
2 REM =
3 REM = CLASSEMENT DE NOMBRES =
4 REM =
5 REM = 19 / 01 / 1987 R.S. =
6 REM =====
7 '
8 '
9 '
10 PAPER0:INK7:CLS
11 PRINTCHR$(17);
12 REM -----
13 REM Tirage aleatoire de 3 nbrs
14 REM -----
15 PRINT:PRINT"*** TIRAGE DE 3 NOMBRES
16 HASARD ***"
16 PLOT1,0,20:PLOT1,2,20
20 FOR R=1 TO 3
25 A(R)=INT(RND(1)*1100)
30 IF A(R)<1 OR A(R)>1000 THEN 25
50 NEXT R
51 REM -----
52 REM Echange de variables
53 REM -----
55 X=A(1):Y=A(2):Z=A(3)
57 REM -----
58 REM Verification d'egalite
59 REM -----
60 IF X=Y OR X=Z OR Y=Z THEN 20
```

```
61 REM -----
62 REM Affichage des 3 nombres
63 REM -----
64 PRINT:PRINT:PRINT
65 PRINT"X "; "PREMIER NOMBRE: ";:WAIT10
0:PRINTX
66 PRINT
70 PRINT"Y "; "DEUXIEME NOMBRE: ";:WAIT1
00:PRINTY
71 PRINT
75 PRINT"Z "; "TROISIEME NOMBRE: ";:WAIT
100:PRINTZ
80 PRINT:PRINT"*****"
*****":PRINT:PRINT
101 REM -----
102 REM Classement ordre croissant
103 REM -----
104 PRINT:PRINT:WAIT50:PRINT" 1"," 2","
3":PRINT
105 IF X>Y AND X>Z AND Y>Z THEN PRINTX,
Y,Z
110 IF X>Y AND X>Z AND Y<Z THEN PRINTX,
Z,Y
115 IF X<Y AND X<Z AND Y>Z THEN PRINTY,
Z,X
125 IF X>Y AND X<Z AND Y<Z THEN PRINTZ,
X,Y
130 IF X<Y AND X<Z AND Y<Z THEN PRINTZ,
Y,X
135 IF X<Y AND X>Z AND Y>Z THEN PRINTY,
X,Z
180 REM -----
181 REM TEST
182 REM -----
185 PLOT4,25,12
190 PLOT5,25,"VOULEZ-VOUS RECOMMENCER ?
":GETA$
195 IF A$="O" THEN CLS:GOTO12
196 IF A$="N" THEN CLS:END
197 GOTO190
```

Explication du programme :

- 10 >choix des couleurs + efface écran
- 11 >arrêt du clignotement curseur
- 15 >affichage du titre
- 16 >effet de couleurs d'encadrement
- 20 >début de la boucle R
- 25 >attribution aléatoire à A(R)
- 30 >test des bornes de travail
- 50 >retour bouclé vers 20 (3x)
- 55 >échange de A(1) vers X ... etc
- 60 >test de non répétition
- 64 >ligne de syntaxe
- 65 à 80 >affichage des 3 nbrs tirés
- 104 >ligne de syntaxe
- 105 à 135 >classement proprement dit
- 185 à 197 >test opérateur

- Le problème à résoudre est :
Modifier le programme pour permettre
de classer 4 chiffres ce d'une façon
DECROISSANTE.

Remarque: cette modification est
réalisable sans l'aide de l'organigramme

Si ce mot vous est inconnu, ne vous
inquiéter pas car on n'en parlera bien-
tôt.

BONNE CHANCE.

**Vous pouvez copier sur le programme précédent!*