

<https://clx.asso.fr/spip/?Le-concours-du-reporter-2-3>



Concours de programmation

Le concours du reporter (2/3)

- L'Association - Nos actions - 2003 : Concours de programmation -



Publication date: jeudi 16 janvier 2003

Copyright © Club LinuX Nord-Pas de Calais - Tous droits réservés

Une fois compris le principe du concours, voici les règles un peu plus détaillées, les contraintes à respecter, et la façon dont tout cela va se dérouler.

Syntaxe des fichiers résultat

Le fichier résultat (de même nom que le pilote logiciel et d'extension F1) est le fichier que doit écrire votre programme, contenant l'évolution de votre voiture sur la piste.

Le fichier doit être de type 'ASCII' formé de caractères regroupés en différentes lignes (séparées classiquement par les caractères "carriage return" et éventuellement "linefeed").

Par convention, la première ligne est une chaîne de caractères qui vous est propre, et permet d'identifier le pilote automatique qui a généré le fichier.

La deuxième ligne doit indiquer le nom du fichier CIR (donc le nom du circuit) qui a été utilisé pour donner ce résultat.

Les lignes suivantes (donc à partir de la troisième ligne) contiennent la liste des différentes positions de la voiture. La syntaxe imposée est d'abord la position en X (longueur de 0 à 149), puis la position en Y (hauteur de 0 à 99) séparés par une virgule ','.

Le premier point indiqué est obligatoirement celui de la position de départ (40,4).

Puisque la vitesse au départ est nulle, le point suivant est obligatoirement un voisin du point de départ, a priori de coordonnée X supérieure (41) à cause de la progression de la voiture dans le sens des aiguilles d'une montre..

Le dernier point indiqué est celui qui a permis de passer la ligne d'arrivée (position x supérieure ou égale à 40).

Voici un extrait de fichier résultat correct :

```
Pilote automatique de Ferraro, écrit par Jean Vazy  
monaco.cir  
40,4  
41,4  
(...)  
38,6  
42,5
```

Entre (41,4) et (38,6) se trouvent bien entendu la série des coordonnées qui permettent de boucler le circuit...

Structure du pilote automatique

Voyons ensemble l'ordinogramme de base des traitements que doit être en mesure d'accomplir votre pilote logiciel :

```

.
+-----+
| Pilote Logiciel |
+-----+
|
+-----+
| Chargement dans un tableau d'un
fichier .CIR | contenant la description d'un circuit sous la forme | d'une matrice
de 100 lignes de 150 caractères, ou un | caractère espace représente la piste et tout autre |
| le décor. |
+-----+
|
+-----+
| Ouverture du fichier destiné à
contenir les | déplacements du bolide sur la piste. Fichier du même | nom que
votre programme mais d'extension .F1. |
+-----+
|
+-----+
| Positionnement dans le tableau sur la
ligne de départ | fixé par défaut en ordre 40,4 et écriture de cette | position dans
le fichier des déplacements .F1 |
+-----+
|
+-----+
| Calcul de prochaine position du
bolide sur la piste | +>-| (donc dans le tableau) en respectant les règles de |
déplacement et sans collision. |
+-----+
|
+-----+
| Ecriture de la nouvelle position dans
le fichier des | déplacements .F1
+-----+
|
+-----+
| N+----| Test si une touche est pressée *
+----+O | O| +-----+ |U | N|
|I | +-----+ | | | Test si
la ligne d'arrivée | est franchie dans le bon |
| sens (même position dans +----+O | le tableau que la ligne de |U
| départ) | |I |
+-----+
|
+-----+ +-----+ +<-| Non ? On continue
| Si oui fermeture du | fichier .F1
+-----+
|
+-----+
| Sortie du programme |
+-----+
-D- +-----+
Validation et chronométrage du fichier .F1 pour le | circuit traité par le programme fourni
TESTF1.EXE | +-----+

```

Le temps alloué [1] pour créer un fichier .F1 des déplacements étant limité à 5 mn pour un circuit, le programme doit être interruptible par simple pression d'une touche, avec un temps de réaction ne dépassant pas 10 secondes.

Toutes les informations qui vous sont nécessaires pour la réalisation de ce programme sont détaillées dans les

chapitres précédents (ie : règles de déplacement, formats des fichiers etc.)

Création et vérification des circuits

D'abord pour ceux qui bénéficient d'un DOS, d'un écran VGA avec un driver de souris installé : le programme CREECIR (sous DOS, fourni originellement par le Reporter) va vous faciliter la vie !

Le programme CREECIR a un double but : celui de vous aider à dessiner des circuits de façon simple et visuelle, et celui de vérifier la conformité des circuits avec le règlement.

Le programme nécessite un PC avec carte VGA, et une souris dont le driver est installé.

La syntaxe d'appel est :

```
CREECIR [fichier CIR]
```

Si le fichier CIR est indiqué, le programme va le charger et vous permettre de modifier ce circuit à souhait.

Si le fichier CIR est spécifié mais n'existe pas sur disque, le programme va le créer, en partant d'un circuit 'vierge', sauf la zone de départ imposée.

Si le fichier CIR n'est pas indiqué, le programme va créer un nouveau fichier de nom NOUVEAU.CIR, même s'il existe déjà un fichier de ce nom.

Le programme est juste un programme de dessin 'minimal'. On ne peut charger un circuit à l'intérieur du programme par exemple. Il faut quitter et relancer CREECIR. Pour la gestion de fichier, il faudra parfaire vos connaissances du DOS (COPY ou REN ;-)).

En réalité, tout, dans le programme, se fait à la souris. Oubliez donc le clavier !

Bien entendu, la sauvegarde de vos créations est possible dans le programme (quand même !).

En faisant tourner le programme vous constaterez plusieurs zones :

- 1) Sur la colonne de gauche, la palette de couleurs avec celle qui est sélectionnée entourée d'un fin trait blanc. Les couleurs sont celles de la palette de Windows, car c'est la convention utilisée pour les circuits.
- 2) Au centre, le dessin du circuit.. étonnant non ?
- 3) En bas à gauche, une série d'icônes "outil", dont l'un est sélectionné (entouré d'un liseré rouge).

On distingue 7 outils de création graphique :

- le crayon qui permet de dessiner pixel par pixel ;
- le pinceau qui permet de dessiner par carré de 3*3 pixels (très pratique pour faire des circuits rapidement !) ;
- le trait qui permet de faire des lignes joignant 2 points (Bresenham) ;
- le rectangle qui permet de remplir rapidement une grande zone ;
- le pot de peinture, qui effectue un remplissage (fill) de toute une zone. Méfiez-vous car il n'y a pas de retour en arrière ! (sauvegardez souvent) ;
- un outil qui permet de sélectionner une zone graphique et de la copier dans un presse-papier ;
- l'outil inverse, qui permet de recoller la zone du presse-papier sur le dessin.

Puis on trouve 3 icônes d'outils généraux :

- celui représenté par le symbole OK, permet de vérifier la conformité du circuit au règlement ;
- celui indiquant une disquette permet la sauvegarde de votre oeuvre ;
- celui marqué EXIT permet de sortir du programme (sauvegardez avant, il n'y a pas encore de garde fou).

Un essai valant mieux qu'un long discours, téléchargez le logiciel DOS CREECIR (garanti sans virus), avec comme circuit le fichier MONACO.CIR, fourni à titre de circuit de démonstration.

! Nécessite un driver Souris et un écran VGA (et surtout un DOS) !

Pour ceux qui ne bénéficient pas d'écran VGA, ou de drivers de souris, ainsi que pour les inconditionnels de Windows..

Vous trouverez dans les annexes de cette page quelques outils complémentaires (sous DOS, pour le moment) :

- VERIFCIR.EXE programme non graphique permettant de vérifier la conformité d'un circuit vis à vis du règlement.
- CIR2BMP.EXE programme permettant la conversion des fichiers CIR en fichiers BMP non compressés (16 couleurs). Ceci peut vous permettre d'utiliser vos outils classiques de dessin.
- BMP2CIR.EXE programme inverse du précédent, permettant de générer un fichier CIR, à partir d'un dessin BMP 16 couleurs non compressé.

Attention, pour être compatible avec les circuits, les fichiers BMP doivent être de taille 150*100 pixels et n'utiliser que les 16 couleurs standard de Windows dans leur ordre d'origine !

Chronométrage des résultats

Vous avez développé un pilote automatique de compétition, et il génère un fichier résultat (d'extension F1) très performant !

Mais votre programme est-il correct ?

C'est pas qu'on doute de votre bonne foi, mais une erreur est si vite arrivée (Argghhh ! Là je suis passé trop près du

bord !).

Nous avons donc décidé d'écrire un programme arbitre, sous DOS, fourni par le REPORTER originellement et dont nous n'avons pas les sources, du nom TESTF1.EXE, qui vérifie que le fichier résultat généré par les pilotes logiciels est correct.

A savoir, il vérifie que le tour de piste a été effectué, dans le bon sens et à partir du point de départ imposé. Il contrôle, point par point, l'évolution de votre voiture sur la piste, pour savoir si vous ne faites que des déplacements autorisés et si vous n'entrez pas en collision avec le bord de la piste.

De plus TESTF1 joue le rôle de chronomètre officiel de chaque course.

Il compte le nombre de coups nécessaires à votre voiture pour effectuer le tour de piste et extrapole le dernier coup afin de donner un temps au centième de seconde...

Par ailleurs, il calcule la vitesse moyenne de votre Formule 1, qui sert de départager des éventuels ex aequo.

Vous verrez qu'à l'usage, le programme TESTF1 est très utile, pour vous aider à mettre au point votre pilote.

Certaines options du programme ont en effet été rajoutées juste pour vous faciliter le débogage de vos sources.

Le programme TESTF1.EXE joue donc à la fois le rôle d'arbitre et de chronométreur officiel du concours !

TESTF1 se lance à partir du DOS, à l'aide d'une ligne de commande de ce type :

```
TESTF1 fichier_résultat [-nxxx] [-t] [-l]
```

Les paramètres entre crochet [] sont des paramètres facultatifs.

En plus du nom du fichier résultat que TESTF1 est censé contrôler, on peut lui indiquer différentes options :

– **-nxxx** : Dans cette option le xxx correspond à un chiffre (par défaut le programme utilise -n100). Ce chiffre exprime, en centièmes de seconde, le temps d'attente entre l'affichage de deux points successifs du fichier résultat.

TESTF1 est en effet un programme qui vous permet aussi de visualiser la course, comme si vous y étiez ! (si votre ordinateur est équipé d'une carte VGA)

Ce paramétrage permet d'accélérer à volonté l'affichage, pour ceux qui n'ont pas la patience d'attendre.

A noter la possibilité d'indiquer -n0, qui commute le programme en mode pas à pas (Il faut appuyer sur une touche du clavier entre chaque déplacement...). C'est très pratique pour mettre son pilote logiciel au point !

– **-t** : Pour ceux qui n'ont vraiment pas la patience d'attendre, ou simplement ceux qui n'ont pas de carte VGA, l'option -t désactive l'affichage graphique. Le résultat est donné, brut de fonderie, en mode texte.

– -l : TESTF1 ne se contente pas de tout cela... Pour nous faciliter notre futur travail de dépouillement du résultat des courses, TESTF1 génère un fichier résumé, globalisant les résultats obtenus sur un circuit donné. TESTF1 crée et entretient un fichier de nom :

`nom_du_circuit.LOG`

pour chaque fichier circuit (extension .CIR). Par ce biais il recense tous les résultats obtenus sur ce circuit.

Cela peut vous être aussi utile, pour analyser les progrès de votre programme.

L'option -l interdit au programme TESTF1 d'ajouter le résultat au fichier LOG du circuit.

PS:

Attention, ce article est une copie de l'article original publié dans le Reporter n°7.

Les logiciels dont on parle ici fonctionnent sous DOS, vous pouvez utiliser soit DOSemu, sous linux, soit FreeDOS (qui est un OS à part entière).

[1] Sur un 486 DX2 66mhz