

TI-CONCOURS 2014

Catégorie TI-Basic z80
Finale, épreuve machine

Du 5 au 12 avril 2014

Durée : 3 heures

Coefficient : 4

Le rite de passage



Avant de commencer

Vous avez trois heures pour réaliser les sept programmes qui vous sont demandés, plus vingt minutes pour envoyer vos programmes à l'adresse info@ti-concours.com. Dans cet e-mail vous indiquerez votre pseudonyme, votre code personnel, la catégorie concernée et vous joindrez un fichier .zip ou .rar nommé TZEMXX avec votre code personnel à la place de XX. Si vous dépassez la limite de temps, vous aurez un point de malus par minute entamée de retard (sur un total de 400 points).

La lettre repère de cette épreuve est la lettre R : ainsi, chaque programme sera appelé RXXQN, avec votre code personnel à la place de XX, et le numéro de la question à la place de N. Par exemple, pour la question 3 le candidat dont le code est XE créera un programme nommé RXEQ3. Durant toute cette épreuve, les sous-programmes ne sont pas autorisés.

A chaque question est attribué un niveau de difficulté : facile, moyen ou difficile. Les questions difficiles sont évidemment censées demander un peu plus de réflexion, mais attention, car une question facile ou difficile de notre point de vue ne sera pas forcément perçue de la même manière chez vous, devant votre écran.

Le barème ne sera pas dévoilé : sachez néanmoins que le but est de créer des programmes qui soient rigoureux, optimisés, et efficaces, et non pas esthétiques (du moins dans cette catégorie). Afin de faciliter les choses, on considère que l'utilisateur connaît le fonctionnement des programmes, et qu'il ne commettra jamais d'erreur de saisie.

Bonne chance à tous les finalistes !

Le rite de passage

Au fin fond de l'Amazonie, la tribu Wayaroé se livre, tous les ans, à la cérémonie de passage à l'âge adulte : le « rite de passage ». Un protocole simple, mais un moment très important pour tous les jeunes membres de la tribu, qui peuvent alors partir à la chasse comme leurs aînés, et fonder une famille.

Ces derniers temps, ce système a commencé à poser problème. En effet, en raison de la forte natalité, les prétendants sont de plus en plus nombreux, le nombre de chasseurs augmente pareillement et, en raison de la chasse trop importante, l'équilibre naturel est en danger. Afin de ne pas mettre en colère la déesse de la Nature, les anciens de la tribu décidèrent de mettre en place une procédure de sélection des futurs adultes de la tribu.

Ainsi, si le nombre de candidats est N , seuls 2^P d'entre eux pourront participer, où P est le plus grand entier tel que : $2^P < N$.

Question 1 (facile)

Réalisez un programme qui demande à l'utilisateur un nombre entier strictement positif N , et qui renvoie le nombre de personnes autorisées à participer à la cérémonie s'il y a N participants

Afin que la procédure de sélection soit la plus équitable et la plus juste possible, les futurs adultes sont jugés sur des capacités et des compétences nécessaires pour chasser correctement : rapidité, précision, force physique. Pour ce faire, il a été décidé que les jeunes s'affronteraient en duels (un contre un), dans une arène elliptique tracée à même le sol.

Question 2 (difficile)

Réalisez un programme qui demande à l'utilisateur deux réels positifs I et J , et qui trace sur l'écran graphique de la calculatrice une ellipse, de grand axe I , et de petit axe J (en pixels). L'ellipse devra être centrée, et l'écran graphique ne devra rien comporter à part cette ellipse. On admet que I et J ont des valeurs convenables.

Afin de mieux organiser la procédure de sélection, les prétendants sont réunis en « poules », de 6 ou de 7, dans lesquelles tout le monde affronte tout le monde. Pour n participants, le nombre de poules constituées est donc égal à la partie entière de $n/7$. Pour la répartition des participants dans les différentes poules, ces derniers sont triés selon leur âge (qui est compté en jours), dans l'ordre décroissant.

Voici un exemple :

Participants	Age		Participants	Age
1	6585		7	7020
2	7201		8	6878
3	6935		9	6758
4	6633		10	6949
5	6599		11	6875
6	6723			

On trie d'abord les participants par âge :

Participants	Age	Participants	Age
2	7201	9	6758
7	7020	6	6723
10	6949	4	6633
3	6935	5	6599
8	6878	1	6585
11	6875		

On fait alors la répartition en poules :

Poule 1	Poule 2
2	7
3	10
8	11
6	9
4	5
	1

On met d'abord les premiers du classement effectué dans l'ordre des poules (1, 2, ...), puis les suivants dans l'ordre l'inverse (n, n-1, ...), puis ceux qui suivent à nouveau dans le bon ordre, et ainsi de suite (on s'arrête lorsque tout le monde a été casé quelque part).

Question 3 (moyen)

Réalisez un programme qui demande un entier naturel N représentant le nombre de participants, et qui renvoie une liste, dont le $i^{\text{ème}}$ terme est égal au nombre de participants présents dans la poule numéro i.

Question 4 (difficile)

Réalisez un programme qui récupère une liste L_1 dont la dimension est égale au nombre de participants, et dont le $i^{\text{ème}}$ terme est égal à l'âge du participant i, ainsi qu'un entier strictement positif N inférieur ou égal au nombre de poules constituées, et qui affiche les numéros correspondant aux participants qui ont été placé dans la poule numéro N.

Une fois que tous les participants d'une poule se sont affrontés, un classement est effectué selon les critères suivants :

- le nombre de fois où les participants ont gagné leur duel divisé par le nombre de duels qu'ils ont eu à faire dans leur poule ;
- la note moyenne, qui comme son nom l'indique est la moyenne des notes attribuées à la fin de chaque duel à chaque participant selon la qualité de sa prestation.

Afin de simplifier les choses, on met les résultats des poules sous la forme d'une matrice. Ainsi, la poule dont les résultats sont les suivants donne la matrice ci-dessous :

Participant	1	2	3	4	5	6
1		8	6	9	4	8
2	7		3	6	8	8
3	8	6		5	8	8
4	6	5	7		4	6
5	7	7	6	7		6
6	7	9	6	9	8	

=>

[[0, 8, 6, 9, 4, 8]
 [7, 0, 3, 6, 8, 8]
 [8, 6, 0, 5, 8, 8]
 [6, 5, 7, 0, 4, 6]
 [7, 7, 6, 7, 0, 6]
 [7, 9, 6, 9, 8, 0]]

Note importante : le vainqueur a forcément une note strictement supérieure à celle du vaincu.

Question 5 (difficile)

Réaliser un programme qui récupère une matrice [A] définie comme ci-dessus, et qui affiche dans l'ordre les numéros des participants après qu'ils aient été classés, du premier au dernier (la première ligne correspond au participant 1, et ainsi de suite).

Une fois que toutes les poules sont terminées, un classement général est effectué. Le chef du village étant féru de statistiques, il souhaite étudier la proportion de participants autorisés à participer à la cérémonie, en fonction de leur âge.

Question 6 (moyen)

Réaliser un programme qui récupère deux listes L_1 , L_2 et L_3 , dont la dimension est égale au nombre de participants, et dont les $i^{\text{ème}}$ termes sont respectivement égaux au pourcentage de victoire, à la note moyenne et à l'âge du participant i , et qui renvoie deux listes L_4 et L_5 , dont la dimension est égale au nombre de participants, et dont les $i^{\text{ème}}$ termes sont respectivement égaux au numéro du participant arrivé $i^{\text{ème}}$, et à son âge.

Pour simplifier le côté technique des choses, nous avons jusqu'à présent omis les noms des participants, en les numérotant. Mais, le protocole du rite de passage impose que les noms des participants (à la cérémonie, cette fois) soient lus à voix haute, dans l'ordre alphabétique.

Question 7 (difficile)

Réaliser un programme qui demande à l'utilisateur un entier N, qui demande ensuite N chaînes de caractères de longueur comprise entre 1 et 10 inclus, et qui affiche ces chaînes de caractères dans l'ordre alphabétique. On suppose que les chaînes de caractères ne comportent que les lettres en majuscule, et on veillera à mettre des Pause entre chaque affichage.

Vous avez terminé ? Félicitations ! Grâce à vous, tout va rentrer dans l'ordre dans la forêt Amazonienne, et la tribu Wayaroé va continuer à vivre des jours heureux, sous la protection de leurs dieux... enfin, en supposant que vos programmes fonctionnent. C'est ce que nous allons voir ! ;-)