

I1101 - TP n°10

Exercice 1

Ecrire un programme qui lit deux entiers positifs A et B, puis appelle une *fonction récursive* qui calcule le produit de A et B en utilisant la méthode Egyptienne. Les règles de la méthode Egyptienne sont les suivantes :

- Si $B=0$, alors $A*B=0$
- Si B est pair, alors $A*B = (2*A) * (B/2)$
- Si B est impair, alors $A*B = A * (B-1) + A$

Exercice 2

Comme déjà expliqué au TP 3, la suite de Fibonacci est une séquence de nombres où les premiers deux nombres sont 1 et 1. Ensuite, chaque nombre est la somme des deux nombres qui le précèdent. Voici le début de la suite de Fibonacci : **1, 1, 2, 3, 5, 8, 13, 21, ...**

$U_1=1, U_2=1$

$U_n=U_{n-1} + U_{n-2}$ for $n>2$

Ecrire un programme qui lit un entier N et puis appelle une *fonction récursive* qui calcule la « N » ième valeur de la suite de Fibonacci. A la fin, le programme affiche cette valeur à l'écran.

Exemple d'exécution :

Saisir un entier : 9

L'entier numéro 9 de la suite de Fibonacci est 34

Exercice 3

On veut concevoir une calculatrice simple qui lit deux nombres de l'utilisateur, puis lui propose la liste d'options suivante :

Saisir 1 pour effectuer une addition
Saisir 2 pour effectuer une soustraction
Saisir 3 pour effectuer une multiplication
Saisir 4 pour effectuer une division
Saisir 0 pour quitter le programme

Concevoir et écrire un programme modulaire avec les fonctions suivantes :

- Une fonction qui affiche la liste d'options,
- Une fonction qui prend en paramètre les deux nombres lus et le choix de l'utilisateur pour appliquer l'opération correspondante et retourner le bon résultat,
- Une fonction qui analyse le choix de l'utilisateur :
 - o Quand il saisit une valeur égale à 1, 2, 3 ou 4, le programme appelle la fonction responsable du calcul correspondant,
 - o Quand il saisit 0 le programme affiche le message « Merci d'avoir utilisé notre calculatrice, à bientôt ! » et puis se termine,
 - o Sinon, il lui affiche « la valeur saisie ne correspond pas à une tâche reconnue par le programme ! », et puis lui affiche la liste d'options de nouveau.

Exercice 4

Supposer un utilisateur possède des informations médicales sur N patients impliqués dans une étude de recherche ($0 < N < 100$). Ecrire un programme qui demande le nombre de patients, puis enregistre leurs **sexes** (0 = homme ou 1 = femme), **âge** et **poids**. Le programme remplit 3 tableaux de N valeurs chacun depuis la saisie de l'utilisateur, par exemple :

Veillez saisir le nombre de vos patients:

4

Veillez saisir le sexe, l'âge et le poids de vos 4 patients:

1	26	50.12
0	31	106.77
1	29	70.89
1	41	95.16

Votre programme doit utiliser une fonction qui calcule la dose à prendre par chacun des patients, et puis la remplir dans un 4^{ième} tableau, selon la formule suivante :

Homme : 0.5 mg pour chaque 4.5 Kg de poids

Femme : 0.3 mg pour chaque 3.6 Kg de poids

La dose de la pilule à donner au patient est également calculée par une fonction du programme. En effet, les pilules de cette étude ont des poids *pairs* en mg (2, 4, 6, etc.). La pilule exacte à donner est celle ayant le poids le plus proche n'excédant pas la dose requise. Par exemple, pour un patient dont il est recommandé de prendre une dose de 5.4 mg, le programme doit suggérer une pilule de 4 mg.

Quand l'utilisateur finit de saisir le sexe, âge et poids des N patients, le programme doit appeler les fonctions nécessaires afin de calculer la dose et la pilule requises, puis une fonction qui affiche toutes les informations dans un format tabulaire bien propre, comme suit :

Sexe	Age	Poids	Dose	Pilule
====	===	=====	====	=====
F	26	50.12	4.17 mg	4 mg
H	31	106.77	11.86 mg	10 mg
F	29	70.89	5.9 mg	4 mg
F	41	95.16	7.9 mg	6 mg