

EXAMEN

Atelier de programmation I

SECTION : LGSI 1

Nature de l'épreuve : DC <input type="checkbox"/> DS <input type="checkbox"/> EF <input checked="" type="checkbox"/>	Documents : autorisés <input type="checkbox"/> non autorisés <input checked="" type="checkbox"/>
Date de l'épreuve : 09/01/2023	Calculatrice : autorisée <input type="checkbox"/> non autorisée <input checked="" type="checkbox"/>
Durée de l'épreuve : 1h30	Session : principale <input checked="" type="checkbox"/> contrôle <input type="checkbox"/>

Exercice 1

Un entier N ($N \geq 10$) est divisible par 13, si on multiplie successivement le chiffre d'unité par 4 et on l'additionne au nombre privé de son chiffre d'unité, le traitement s'arrête lorsque le résultat obtenu est composé de 2 chiffres, s'il est un multiple de 13, alors le nombre est divisible par 13.

Exemple :Pour $N=224185$

En effet : $(4*5) + 22418 = 22438$

$(4*8) + 2243 = 2275$

$(4*5) + 227 = 247$

$(4*7) + 24 = 52$

52 est un multiple de 13, donc 224185 est divisible par 13.

Pour $N=6355$

En effet : $(4*5) + 635 = 655$

$(4*5) + 65 = 85$

85 n'est pas un multiple de 13, donc 6355 n'est pas divisible par 13.

Travail demandé :

Ecrire un programme C permettant de saisir un entier $N \geq 10$ et de vérifier si N est divisible par 13 en utilisant le principe décrit précédemment.

$$\begin{array}{r} 224185 \\ \hline 10 \\ \overline{224185} \\ 15 \end{array}$$

Exercice 2

Nous appellerons « point d'équilibre » d'un tableau T de taille N tout indice k qui sépare deux sous-tableaux ayant la même somme des éléments ($S1=S2$), avec $S1$ est la somme des éléments indicés de 0 à $K-1$ et $S2$ la somme des éléments indicés de $K+1$ à $N-1$.

Ex : principe (donner un entier)
 scan (read + m)
 while (m <= 10)
 do

if (res / 13 == 0)
 print ("N est un
 else
 end")

$U = N \% 10$ $res = 4^u U + V$ en
 $V = N / 10$ while (res < 10 || res > 99)

Exemple : pour le tableau T suivant, les points d'équilibre sont 3 et 6.

-2	19	-8	-14	4	5	-4	3	-8	9
0	1	2	3	4	5	6	7	8	9
S1=9						S2=9			

-2	19	-8	-14	4	5	-4	3	-8	9
0	1	2	3	4	5	6	7	8	9
S1=4							S2=4		

En se basant sur la définition d'un point d'équilibre :

1. Ecrire une fonction **chargement** qui permet de charger un tableau par N entiers.
void chargement (int T[], int N)
2. Ecrire une fonction **Somme** qui permet de calculer la somme des éléments d'un tableau qui se trouvent entre deux position **P1** et **P2**.
int Somme(int T[], int P1, int P2)
3. Ecrire une fonction **Point_equilibre** qui permet d'afficher la liste des points d'équilibre dans un tableau T d'entiers (utiliser la fonction **Somme**).
void Point_equilibre(int T[], int N)
4. En utilisant les fonctions ci-dessus écrire une fonction **main** permettant :
 - De saisir un entier N avec : $4 < N < 15$
 - De remplir un tableau T par N entiers
 - D'afficher la liste des points d'équilibre dans le tableau T

Bon courage