



## TD 3

### Exercice 1 :

1. Qu'affiche le programme suivant :

```

1  #include<stdio.h>
2  int main(void){
3      int i=0;
4      int n=0;
5  while(n<10){
6      printf("%d\n",i);
7      i++;
8      n=i*i;
9  }
10 }
```

2. Même question si on permute :

- a) la ligne 6 et la ligne 7                      b) la ligne 6 et la ligne 8                      c) la ligne 7 et la ligne 8

### Exercice 2:

Ecrire un programme C qui demande à l'utilisateur d'entrer 1, 2 ou 3 au clavier, puis un entier positif n. Selon le premier nombre saisi, on calcule:

1. la somme des n premiers nombres entiers positifs.
2. le factoriel de n
3. la somme harmonique :  $S = 1 + 1/2 + \dots + 1/n$

### Exercice 3:

Un nombre entier p (différent de 1) est dit premier si ses seuls diviseurs positifs sont 1 et p. Ecrire un programme C qui permet de déterminer si un nombre saisi au clavier est premier ou non.

### Exercice 4 :

Ecrire en C un programme qui calcule le minimum de n nombres entiers entrés au clavier par l'utilisateur.

### Exercice 5 :

Ecrire un programme C permettant de :

1. Lire un nombre donné de notes comprises entre 0 et 20.
2. Afficher la meilleure des notes, la pire des notes et la moyenne des notes.

### Exercice 6 :

Ecrire en C un programme qui affiche un triangle isocèle formé d'étoiles (\*) sur n lignes (n est fourni au clavier)

### Exercice 7 :

Parmi tous les entiers supérieurs à 1, seuls quatre peuvent être représentés par la somme des cubes de leurs chiffres. A titre d'exemple,  $153 = 1^3 + 5^3 + 3^3$  est un nombre cubique. Ecrire un programme C permettant de déterminer les trois autres. Note : les 4 nombres cubiques existants sont compris entre 150 et 410.

### Exercice 8 :

Un nombre parfait est un nombre présentant la particularité d'être égal à la somme de tous ses diviseurs, excepté lui-même. Le premier nombre parfait est  $6 = 3 + 2 + 1$ .

Ecrire un programme C qui affiche tous les nombres parfaits inférieurs à 1000.