

## Exercice 1

Écrire un programme qui demande à un commercial son chiffre d'affaires, qui le lise et calcule le montant de sa prime de rendement. La prime est égale à 1% de la partie du chiffre d'affaires comprise entre 20 000€ et 50 000€, plus 2% de ce qui dépasse 50 000€.

Le programme comportera deux fonctions : la fonction `main()` et une fonction de prototype `float prime(float)` qui calcule la prime selon la formule ci-dessus.

## Exercice 2

1. Tracer le programme suivant :

```
#include<stdio.h>

// Suites de Syracuse

int suivant(int u){
    int parite;

    parite = u%2;

    if (parite == 0) u = u/2;
    else u = 3*u+1;

    //Point d'observation 1

    return u;
}

main(){
    int n;

    n=5;

    while (n!=1){
        printf("%i ", n);
        n=suivant(n);
        //Point d'observation 2
    }

    return 0;
}
```

2. Que donnerait le traçage en remplaçant la ligne  $n = 5$  par  $n = 27$  ?

### Exercice 3

On cherche à éprouver la célèbre formule :

$$e^x = \lim_{n \rightarrow +\infty} \sum_{i=0}^n \frac{x^i}{i!}$$

1. Programmer une fonction qui calcule la factorielle d'un nombre entier. Prototype de la fonction : `int fact(int)`.

2. Programmer une fonction qui calcule une valeur approchée de  $\lim_{n \rightarrow +\infty} \sum_{i=0}^n x^i/i!$ . On additionnera des termes de la forme  $x^i/i!$  tant que  $x^i/i! \geq 0,000\,001$ . Prototype de la fonction à programmer : `float devexp(float)`.

3. Comparer le résultat obtenu avec celui de la fonction exponentielle de la bibliothèque `math.h` du langage C. Pour accéder à cette fonction, `#include<math.h>` et selon les systèmes, option `-lm` à la compilation. Prototype de la fonction : `float exp(float)`