



4 TD : Les fonctions

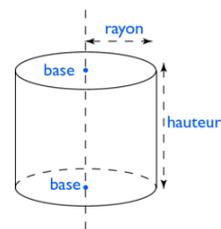
4.1 Âge

Écrire une fonction qui prend en entrée une **année** de naissance d'une personne et l'année courante et qui calcule ensuite l'**âge** de cette personne.

Tester la fonction dans un programme.

4.2 Cylindre

1. Écrire une fonction *Aire* qui permet de calculer l'aire d'un cylindre, sachant que l'aire d'un cylindre est égale à 2π multiplié par le rayon de la base et par la hauteur.
2. Écrire une fonction *Volume* permettant de calculer le volume d'un cylindre, sachant que le volume d'un cylindre est égal à π multiplié par le rayon de la base au carré et par la hauteur.
3. Écrire un programme qui demande à l'utilisateur le rayon R de la base d'un cylindre et sa hauteur H et qui affiche son air et son volume en utilisant les deux fonctions ci-dessus.



4.3 Fonction $f(x)$

Écrire un programme se servant d'une fonction f pour afficher la table de valeurs de la fonction définie par

$$f(x) = \sin(x) + \ln(x) - \sqrt{x}$$

où x est un entier compris entre 1 et 10.

4.4 Suite de Fibonacci

Écrire une fonction qui permet de calculer le $N^{\text{ième}}$ terme U_N de la suite de Fibonacci qui est donnée par la relation de récurrence :

$$U_1 = 1, U_2 = 1, U_N = U_{N-1} + U_{N-2}$$

N est passé comme paramètre à la fonction.

4.5 Réviser la multiplication

On veut écrire une fonction `ReviserMultiplication` qui demande à un l'utilisateur de réciter sa table de multiplication. L'utilisateur commence par entrer un nombre entre 2 et 9 (si le nombre est incorrect, le programme redemande). Ensuite le programme affiche une à une les lignes de la table de multiplication de ce nombre, en laissant le résultat vide et en attendant que l'utilisateur entre le résultat. Si celui-ci est correct, on passe à la ligne suivante, sinon on affiche un message d'erreur donnant la bonne valeur et on termine. Si toutes les réponses sont correctes, on affiche un message de félicitations. On représente ci-dessous une exécution possible (les entrées de l'utilisateur sont affichées en italiques)

```
Valeur de n : 12
Réessayez : la valeur doit être comprise entre 2 et 9
Valeur de n : 6
1 x 6 = 6
2 x 6 = 12
3 x 6 = 21
Erreur! 3 x 6 = 18 et non 21
...
```

1. Écrire la fonction `ReviserMultiplication`.
2. Modifier la fonction `ReviserMultiplication` pour qu'elle ne s'arrête pas quand une réponse fausse est donnée, mais retourne à la fin le nombre d'erreurs commises.

4.6 Nombre premier suivant

1. Écrire une fonction `estDivisible(a, b)` qui renvoie 1 si a est divisible par b
2. Écrire une fonction `premier` d'un paramètre entier m et retournant 1 si le nombre est premier et 0 dans le cas contraire.
3. Écrire une fonction `prochainPremier` prenant un paramètre entier n et retournant le plus petit nombre premier plus grand ou égal à n.
4. Écrire un programme qui demande un entier n à l'utilisateur et affiche le premier nombre premier plus grand ou égal à n.