

5.4 Activités



ACTIVITÉ 1

(Capitale : 0fcb-77439) Écrire un programme qui demande un entier n à l'utilisateur, puis calcule et affiche le résultat de la multiplication

$$\underbrace{2 \times 2 \times 2 \times \dots \times 2}_{n \text{ fois}}$$



ACTIVITÉ 2

Écrire un programme qui calcule et affiche $1 \times 2 \times \dots \times 100$.



ACTIVITÉ 3

Écrire un programme qui demande un entier n à l'utilisateur puis calcule et affiche

1. $1 + 2 + \dots + n$
2. le nombre entier $n*(n+1)/2$

CORRECTION

```
[ ]: n = int(input("entrer un nombre"))

# Question 1
r = 0
for i in range(n):
    r = r + i + 1
print("réponse 1:", r)

# Question 2
print("réponse 2:", n*(n+1)//2)
```

**ACTIVITÉ 4**

(Capytale : 1983-77502) Écrire un programme qui demande à l'utilisateur

- une somme d'argent initiale s déposée sur un livret,
- un taux d'intérêt annuel t exprimé en pourcents
- un nombre d'année n

et qui affiche les intérêts perçus chaque année ainsi que le montant total présent sur le livret après n années.

(chaque année, il faut ajouter à s la quantité $s*t/100$)

CORRECTION

```
[ ]: s = 10_000 # 10000,00 mais écrit plus lisible
      t = 1.5  # 1,50 %
      n = 5

      for _ in range(n):
          interet = s*t/100
          s = s + interet

      print("somme finale :", s)
```

**ACTIVITÉ 5**

(Capytale : e7b7-77504)

1. Écrire un programme qui demande à l'utilisateur un nombre de chiffres n puis n chiffres, et qui calcule et affiche le nombre formé avec les n chiffres fournis dans l'ordre.
2. Écrire une variante du programme précédent dans lequel les chiffres sont donnés dans l'ordre inverse.

CORRECTION

L'idée du code suivant est d'écrire le prochain chiffre à droite de tous les précédents. Cela revient donc à multiplier par 10 le nombre obtenu jusque là et à ajouter le nouveau chiffre.

```
[ ]: n = int(input("nombre de chiffres"))
r = 0
for i in range(n):
    chiffre = int(input("Entrer le prochain chiffre : "))
    r = 10 * r + chiffre
print(r)
```

Dans le cas où les chiffres sont données en sens inverses, chaque chiffre à ajouter est multiplié par une puissance de 10 correspondant à sa position.

CORRECTION

```
[ ]: n = int(input("Entrer le nombre de chiffres : "))
r = 0
for i in range(n):
    chiffre = int(input("Entrer le prochain chiffre : "))
    r = r + chiffre * (10**i)
print(r)
```

**ACTIVITÉ 6**

(Capytale : a7bc-77505) Écrire un programme qui demande à l'utilisateur un nombre entier n et un nombre de chiffres k , et qui affiche successivement les k derniers chiffres de n , en commençant par les unités.

Si n contient moins de k chiffres, il suffira d'afficher des zéros à la fin.

(on rappelle que $n \% 10$ renvoie le chiffre des unités de n)

CORRECTION

```
[ ]: n = int(input("Entrer un nombre entier : "))
      k = int(input("Entrer un nombre de chiffres : "))

      for _ in range(k):
          print(n % 10)
          n = n // 10
```

**ACTIVITÉ 7**

(Capytale : 2294-77506) La suite de Fibonacci est la suite d'entiers (F_n) définie par $F_0 = 0$, $F_1 = 1$ et, pour tout entier n à partir de 2, $F_n = F_{n-1} + F_{n-2}$.

Écrire un programme qui demande à l'utilisateur un entier n qu'on supposera supérieur ou égal à 1, et qui affiche la valeur de F_n .

(on utilisera deux variables pour mémoriser F_n et F_{n-1} ainsi qu'une variable temporaire)

CORRECTION

```
[ ]: n = int(input("Entrer un nombre entier : "))

fib_m = 0    # valeur en n-1
fib_n = 1    # valeur en n

for _ in range(2, n+1):
    tmp = fib_m
    fib_m = fib_n
    fib_n = fib_m + tmp

print(fib_n)
```