

*Atelier1 : Langage C*

**Exercice 0 :** Ecrire un programme permettant d'afficher « je suis un étudiant de Ecole Supérieure de Technologie d' Essaouira » sous forme de pyramide.

**Exercice 1 :** Ecrire un programme permettant de calculer la somme, la division, la soustraction et la multiplication de deux nombres lus au clavier. (afficher les résultats au écran)

**Exercice 2 :** Ecrire un programme permettant d'échanger les valeurs de deux variables x et y (avec  $x=0,3 \times 10^4$  et  $y=0.5$ ).

**Exercice 3 :** Écrire un programme qui consiste a calculer l'air S d'un cercle selon la formule  $S = \pi * R^2$  (Rappel :  $\pi = 3.14159$  et R le rayon du cercle)

**Exercice 4:** Ecrire un programme qui demande un nombre entier à l'utilisateur, puis qui calcule et affiche le double de ce nombre?

**Exercice 5:** Écrire un programme qui consiste a afficher la valeur absolue d'un nombre réel?

**Exercice 6:** Ecrire un programme qui demande un nombre entier à l'utilisateur, puis qui teste et affiche s'il est divisible par 3?

**Exercice 7 :** n étant de type int, écrire une expression qui prend la valeur :

-1 si n est négatif, 0 si n est nul, 1 si n est positif.

**Exercice 8:** Quels résultats fournit le programme suivant ?

```
#include <stdio.h>
main(){
int n=10, p=5, q=10, r ;
r = n == (p = q) ;
printf ("A : n = %d p = %d q = %d r = %d\n", n, p, q, r) ;
n = p = q = 5 ;
n += p += q ;
printf ("B : n = %d p = %d q = %d\n", n, p, q) ;
q = n < p ? n++ : p++ ;
printf ("C : n = %d p = %d q = %d\n", n, p, q) ;
q = n > p ? n++ : p++ ;
printf ("D : n = %d p = %d q = %d\n", n, p, q) ; }
```

**Exercice 9 :** L'utilisateur appuie sur une touche de clavier, le programmeur teste, si c'est une lettre il la affiche sinon il va renvoyer un message d'erreur.

**Exercice 10 :** Soit le code :

```
#include <stdio.h>
main(){
int n, p ;
printf ("donnez une valeur pour n : ") ;
scanf ("%d", &n) ;
printf ("merci pour %d\n", n) ;
printf ("donnez une valeur pour p : ") ;
scanf ("%d", &p) ;
printf ("merci pour %d", p) ; }
```

Que se passe lorsque l'utilisateur donne 12 90 comme valeur ?

**Exercice 11 :** Ecrire un programme qui résout les équations du second degré ( $ax^2 + bx + c = 0$ ). Le programme demande a, b et c à l'utilisateur puis indique le nombre de solutions ainsi que leurs valeurs.

**Exercice 12 :** Ecrire un programme retournant le PGCD ainsi que le PPCM de 2 entiers entrés par l'utilisateur.

**Exercice 13 :** Ecrire un programme qui, à partir de la saisie d'une date de naissance, affiche le signe du zodiaque correspondant.

Capricorne :	22/12 au 20/1	Verseau :	21/1 au 19/2
Poissons :	20/2 au 20/3	Bélier :	21/3 au 20/4
Taureau :	21/4 au 20/5	Gémeaux :	21/5 au 21/6
Cancer :	22/6 au 22/7	Lion :	23/7 au 23/8
Vierge :	24/8 au 23/9	Balance :	24/9 au 23/10
Scorpion :	24/10 au 22/11	Sagittaire :	23/11 au 21/12

Utiliser la construction if ... else puis switch ... case.

**Exercice 14 :** En utilisant l'opérateur conditionnel (? :), calculer et afficher le max et le min de deux nombres lus au clavier ?

Même question pour trois nombres lus au clavier ?

**Exercice 15:** Quel type d'erreur contient ce code ? Corrigez le ?

```
main(){
int n ;
do{
printf ("donnez un nombre : ") ;
scanf ("%d", &n) ;
printf ("voici son carré : %d\n", n*n) ;
}while (n) ;
}
```

**Exercice 16:** Quels seront les résultats fournis par ce programme ?

```
#include <stdio.h>
main (){
int n = 543 ;
int p = 5 ;
float x = 34.5678;
printf ("A : %d %f\n", n, x) ;
printf ("B : %4d %10f\n", n, x) ;
printf ("C : %2d %3f\n", n, x) ;
printf ("D : %10.3f %10.3e\n", x, x) ;
printf ("E : %*d\n", p, n) ;
printf ("F : %*. *f\n", 12, 5, x) ;}
```

**Exercice 17:** Que fait ce programme ? quel est le rôle de l'instruction break ?

```
main(){
int n ;
printf ("donnez un entier : ") ;
scanf ("%d", &n) ;
switch (n){
case 0 : printf ("nul\n") ;
break ;
case 1 : printf ("un\n") ;
}
```

```

break ;
case 2 : printf ("deux\n") ;
break ;
}
printf ("au revoir\n") ;
}

```

**Exercice 18:** Soit le petit programme suivant :

```

#include <stdio.h>
main(){
int i, n, som ;
som = 0 ;
for (i=0 ; i<4 ; i++)
{ printf ("donnez un entier ") ;
scanf ("%d", &n) ;
som += n ;
}
printf ("Somme : %d\n", som) ;
}

```

Écrire un programme réalisant exactement la même chose, en employant, à la place de l'instruction for :

- une instruction while,
- une instruction do... while.

**Exercice 19:** Calculer la moyenne de notes fournies au clavier avec un dialogue de ce type :

```

note 1 : 12
note 2 : 15.25
note 3 : 13.5
note 4 : 8.75
note 5 : -1
moyenne de ces 4 notes : 12.37

```

Le nombre de notes n'est pas connu a priori et l'utilisateur peut en fournir autant qu'il le désire. Pour signaler qu'il a terminé, on convient qu'il fournira une note fictive négative. Celle-ci ne devra naturellement pas être prise en compte dans le calcul de la moyenne.

**Exercice 20:** Ecrire un programme permettant de réaliser les opérations d'une calculatrice.

**Exercice 21 :** Calculez la factorielle  $N! = 1*2*3*...*(N-1)*N$  d'un entier naturel N en respectant que  $0!=1$ .

- a) Utilisez while,
- b) Utilisez for.

**Exercice 22 :** soit le code :

```

main ( ){
float Pi = 3.14159;
float rayon, surface;
scanf (« %f », &rayon);
surface = pi*pow (rayon,2);
printf (« %fn »,surface ) ;
}

```

- 1) Quels résultats fournit le programme suivant ?
- 2) Calculer le périmètre d'un cercle.

3) Discuter le cas où l'utilisateur peut fournir des valeurs négatives.

**Exercice 23:** Ecrire un programme qui demande un nombre entier à l'utilisateur, puis qui teste et affiche s'il est divisible par 3.

**Exercice 24 :** 1) Ecrivez un programme qui demande à l'utilisateur le nombre de photocopies effectuées, qui calcule et affiche le prix à payer

2) Un programme qui détermine le premier nombre entier N tel que la somme de 1 à N dépasse strictement 100

**Exercice 25:** Traduire l'algorithme suivant en langage C :  
Algo puissance1

```
Variables x, puiss : réels
          n, i : entier
Debut
  Ecrire (" Entrez la valeur de x ");
  Lire (x);
  Ecrire (" Entrez la valeurs de n ");
  Lire (n);
  puiss ← 1;

  Pour i allant de 1 à n
    puiss ← puiss*x ;
  FinPour
  Ecrire (x, " à la puissance ", n, " est égal à ", puiss);
Fin
```

**Exercice 26:** Même question que l'exercice précédent et comparez les deux algorithmes?  
Algo puissance2

```
Variables      x, puiss : réel
               n, i : entier
Debut
  Ecrire (" Entrez la valeur de x ");
  Lire (x);
  Ecrire (" Entrez la valeurs de n ");
  Lire (n);
  puiss ← 1,
  i ← 1 ;
  TantQue (i ≤ n)
    puiss ← puiss*x ;
    i ← i+1;
  FinTantQue
  Ecrire (x, " à la puissance ", n, " est égal à ", puiss);
Fin
```

**Exercice 27:** Ecrire en C le programme qui calcul la Racine Carrée suivante :

Algorithme RacinCarre

Variables     n, x : réels  
              i, max : entier

Début

    Ecrire (" Entrez la valeur de n ");  
    Lire (n);

    Ecrire (" Entrez la valeurs de max");  
    Lire (max);

$x \leftarrow 1$ ;

    pour i allant de 1 à max

$x \leftarrow ((n/x) + x) / 2$ ;

        Ecrire (x, n);

    FinPour

    Ecrire (x)

Fin

**Exercice 28:** Ecrire en C un programme qui met en évidence l'intérêt des instructions **break**, **goto** et **continue** (reprenez les exemples du cours)

**Exercice 29:** Ecrire en C un programme permettant de

- a- Convertir un nombre entier N en un nombre représentant sa valeur en binaire (base 2).

Exemple : le nombre 204 est converti en 11001100

- b- Convertir un nombre entier N représentant une valeur en binaire (on vérifiera cette propriété) en un nombre en base 10.

Exemple: le nombre 11001100 est converti en 204

Le nombre 100301 n'est pas Valide et doit être ressayer

**Exercice 30 :** Une année a est bissextile si elle est multiple de 4, et si elle est multiple de 100 elle doit aussi être multiple de 400. Par exemple 1996 oui, 1900 non, 2000 oui.

TD : Ecrire l'expression correspondante et donner l'algorithme qui demande l'année, puis affiche si elle est bissextile.

TP : Ecrire le programme C correspondant (utiliser l'opérateur conditionnel).

**Exercice 31:** Une boulangerie est ouverte de 7 heures à 13 heures et de 16 heures à 20 heures, sauf le lundi après-midi et le mardi toute la journée. On suppose que l'heure h est un entier entre 0 et 23. Le jour j code 0 pour lundi, 1 pour mardi, etc.

TD : Ecrire l'expression correspondante, en essayant de trouver la plus courte.

Ecrire l'algorithme qui demande le jour et l'heure, puis affiche si la boulangerie est ouverte.

TP : Donner le programme C qui traduit exactement votre algorithme.

**Exercice 32 :** Pour vérifier la validité d'un numéro, tel que celui des cartes bancaires, des cartes SNCF, ... un algorithme a été développé. Cet algorithme consiste à prendre chaque chiffre du numéro, le multiplier par deux s'il est pair et on additionne chaque chiffre ainsi

obtenu. Si le résultat est un multiple de 10, le numéro est valide.

Donner l'algorithme détaillé qui permet de lire un tel numéro et afficher s'il est valide ou non (TP) et écrire le programme C qui implémente cet algorithme (TP).

Exemple : numéro : 145789  $\rightarrow 1+(2*4)+5+7+(8*2)+9 = 46 \rightarrow$  numéro non valide  
numéro : 34861  $\rightarrow 3+(2*4)+(8*2)+(6*2)+1 = 40 \rightarrow$  numéro valide

**Exercice 33:** Parmi tous les entiers  $> 100$ , seuls quatre entiers peuvent être représentés par la somme des cubes de leurs chiffres. Par exemple  $153 = 1*1*1 + 5*5*5 + 3*3*3$ .

Ecrire un programme C qui permet de trouver tous les nombres d'AMSTRONG. Le programme doit commencer par l'affichage du message "Les nombres d'AMSTRONG sont :" et se terminer par l'affichage du message "Fin des nombres d'AMSTRONG".

**Exercice 34 :** La constante d'Euler  $e$  peut s'écrire comme :

$$e = 1 + 1 + \frac{1}{2} + \frac{1}{6} + \dots + \frac{1}{n!} + \dots = \sum_{n=0}^{+\infty} \frac{1}{n!}$$

Ecrire le programme C qui permet de calculer une approximation de  $e$  en utilisant  $p$  termes de la série. Le calcul doit s'arrêter quand (1) on arrive  $p=100$  ou bien (2) la valeur absolue de la différence entre les deux dernières valeurs calculées est inférieure strictement à  $10^{-2}$ .