

### Algorithmique et structures de données 1

## TP 3 - Solution

**Exercice 1.** Ecrire un programme C qui calcul la somme de tous les nombres naturels entre 1 et n en utilisant la boucle **for**. Réécrire le programme en utilisant la boucle **do...while** et la boucle **while**. Par exemple : Entrée : 10. Sortie : Somme des nombres naturels 1-10 : 55.

```
#include <stdio.h>

int main()
{
    int i, n, somme=0;

    /* Entrée de la limite supérieure de l'utilisateur */
    printf("Entrez un nombre entier positif : ");
    scanf("%d", &n);

    /* Trouver la somme de tous les nombres */
    for(i=1; i<=n; i++)
    {
        somme = somme + i;
    }

    printf("Somme des %d premiers nombres naturels = %d", n, somme);

    return 0;
}
```

```
#include <stdio.h>
int main() {
    int n, i, somme = 0;

    printf("Entrez un nombre entier positif : ");
    scanf("%d", &n);
    i = 1;

    while (i <= n) {
        somme += i;
        ++i;
    }

    printf("Somme des %d premiers nombres naturels = %d", n, somme);
    return 0;
}
```

```

#include <stdio.h>
#include <conio.h>
void main()
{
    int n, i, somme = 0;

    printf("Entrez un nombre entier positif : ");
    scanf("%d", &n);
    i = 0;
    do
    {
        somme = somme + i; // stocker la somme des nombres naturels
        i++; // incrémenter de 1
    } while (i <= n);

    printf("Somme des %d premiers nombres naturels = %d", n, somme);
    getch();
}

```

**Exercice 2.** Ecrire un programme C qui calcul la puissance en utilisant uniquement la multiplication (utiliser la boucle **while** et la boucle **for**).

```

#include<stdio.h>
int main()
{
    int base, exposant, puissance, i;
    //Lecture de la base et de l'exposant
    printf("Entrez la base : ");
    scanf("%d",&base);
    printf("Entrez l'exposant : ");
    scanf("%d",&exposant);

    puissance = 1;
    i = 1;
    //calculatif de la puissance d'un nombre donné
    while(i <= exposant)
    {
        puissance = puissance * base;
        i++;
    }
    printf("La puissance : %d", puissance);
    return 0;
}

```

```

#include<stdio.h>
int main()
{
    int base, exposant, i, puissance;

    printf("Entrez la base : ");
    scanf("%d",&base);
    printf("Entrez l'exposant : ");
    scanf("%d", &exposant);

    puissance = 1;
    //calculatif de la puissance d'un nombre donnÃ© en utilisant la boucle for
    for(i=1; i<=exposant; i++)
        puissance = puissance * base;
    printf("La puissance : %d", puissance);
    return 0;
}

```

**Exercice 3.** Ecrire un programme C qui calcul la factorielle d'un nombre entier en utilisant la boucle **while** et la boucle **for**. Par exemple, la factorielle de 7 :  $7!=7*6*5*4*3*2*1=5040$ .

```

#include <stdio.h>
int main()
{
    int n,i;
    long long fact=1;

    printf("Entrez un nombre quelconque pour calculer la factorielle : ");
    scanf("%d",&n);
    i = 1;
    //Faire tourner la boucle de 1 à un nombre entré par l'utilisateur
    while(i <= n)
    {
        fact = fact * i;
        i++;
    }
    printf("Factorielle de %d = %lld", n, fact);
    return 0;
}

```

```

#include <stdio.h>
int main()
{
    int i, n;
    long long fact=1;

    printf("Entrez un nombre quelconque pour calculer la factorielle : ");
    scanf("%d", &n);

    for(i=1; i<=n; i++)
    {
        fact = fact * i;
    }
    printf("Factorielle de %d = %lld", n, fact);
    return 0;
}

```

**Exercice 4.** Ecrire un programme C pour afficher la table de multiplication d'un entier donné.

Par exemple pour 9 :

```

9 x 1 = 9
9 x 2 = 18
9 x 3 = 27
9 x 4 = 36
9 x 5 = 45
9 x 6 = 54
9 x 7 = 63
9 x 8 = 72
9 x 9 = 81
9 x 10 = 90

```

```

#include <stdio.h>
int main()
{
    int i, n;

    printf("Entrez le nombre pour afficher la table de multiplication : ");
    scanf("%d", &n);
    for(i=1; i<=10; i++)
    {
        //Afficher la table de multiplication du nombre entré par l'utilisateur
        printf("%d x %d = %d\n", n, i, (n*i));
    }
    return 0;
}

```

**Exercice 5.** Ecrire un programme C qui permet de vérifier si un nombre est un nombre premier ou non. Utiliser les deux boucles **while** et **for**.

```

#include<stdio.h>
int main()
{
    int n, i, f;

    printf("Entrez un nombre quelconque : ");
    scanf("%d", &n);
    f = 0;
    i = 2;
    while(i <= n/2)
    {
        if(n%i == 0)
        {
            f=1;
            break;
        }
        i++;
    }
    if(f == 0)
        printf("%d est un nombre premier", n);
    else
        printf("%d n'est pas un nombre premier", n);
    return 0;
}

```

```

#include<stdio.h>
int main()
{
    int n, i, f;

    printf("Entrez un nombre quelconque : ");
    scanf("%d",&n);
    f=0;
    for(i=2; i <= n/2; i++)
    {
        if(n%i == 0)
        {
            f=1;
            break;
        }
    }
    if(f==0)
        printf("%d est un nombre premier", n);
    else
        printf("%d n'est pas un nombre premier", n);
    return 0;
}

```

**Exercice 6.** Ecrire un programme C qui permet de trouver le PGCD (Plus Grand Commun Diviseur) de deux nombres entiers. Le PGCD est le plus grand nombre qui divise exactement les deux nombres entiers (sans reste). Par exemple, le PGCD de 54 et 24 est de 6.

```
#include <stdio.h>

int main()
{
    int a, b, pgcd, i;

    printf("Entrez deux nombres entiers : ");
    scanf("%d %d", &a, &b);

    for(i=1; i <= a && i <= b; i++)
    {
        if(a%i==0 && b%i==0)
            pgcd = i;
    }

    printf("PGCD de %d et %d = %d", a, b, pgcd);
    return 0;
}
```

**Exercice 7.** Ecrire un programme C qui permet d'afficher la suite de Fibonacci jusqu'à n termes, sachant que :

$$u(n) = \begin{cases} 0 & \text{si } n = 0 \\ 1 & \text{si } n = 1 \\ u(n-2) + u(n-1), & \text{si } n > 1 \end{cases}$$

Par exemple : le nombre de termes : 10. Suite de Fibonacci : 0, 1, 1, 2, 3, 5, 8, 13, 21, 34.

```

#include <stdio.h>
int main() {

    int i, n;

    // initialiser le premier et le second terme
    int t1 = 0, t2 = 1;

    // initialiser le terme suivant (3éme terme)
    int termeSuivant = t1 + t2;

    // obtenir le nombre de termes de l'utilisateur
    printf("Entrez le nombre de termes : ");
    scanf("%d", &n);

    // afficher les deux premiers termes t1 et t2
    printf("Suite de Fibonacci : %d, %d, ", t1, t2);

    // Afficher du 3éme au n-éme terme
    for (i = 3; i <= n; i++) {
        printf("%d, ", termeSuivant);
        t1 = t2;
        t2 = termeSuivant;
        termeSuivant = t1 + t2;
    }

    return 0;
}

```

**Exercice 8.** Pour chaque suite, écrire un programme C qui permet d'afficher la somme.

1)  $1 + 1/2 + 1/3 + 1/4 + 1/5 \dots + 1/n$

```

// 1+1/2+1/3+1/4+...+1/n
#include<stdio.h>
int main ()
{
    int n,i;
    float s;
    printf("Entrez le nombre de termes : ");
    scanf("%d", &n);
    s=0;

    for(i=1;i<=n;i++)
    {
        s = s + (float)1/i;
    }
    printf("La somme de %d termes de la suite est %f\n", n, s);
}

```

2)  $1/1! + 1/2! + 1/3! + \dots + 1/n!$

```
#include<stdio.h>

int main()
{
int n,i,j,fact=1;
float s=0;
printf("Entrez le nombre de termes : ");
scanf("%d",&n);
for(i=1;i<=n;i++)
{
    fact=1;
    for(j=i;j>0;j--)
    {
        fact=fact * j;
    }
    s=s+(1.0/fact);
}
printf("La somme de %d termes de la suite est %f\n", n, s);
return 0;
}
```

3)  $1/1! + 2/2! + 3/3! + \dots + n/n!$



```

#include <stdio.h>

int main()
{
    int i = 1, j, n;
    float s = 0.0, fact;

    printf("Entrez le nombre de termes : ");
    scanf("%d", &n);

    while(i <= n)
    {
        fact = 1;
        for(j = 1; j <= i; j++)
        {
            fact = fact * j;
        }

        s = s + (i / fact);

        i++;
    }

    printf("La somme de %d termes de la suite est %f\n", n, s);

    return 0;
}

```

4)  $1 + 1/3 + 1/5 + \dots + 1/n$

```

//1+1/3+1/5+...+1/n

#include<stdio.h>
int main ()
{
    int n,i;
    float s;
    printf("Entrez le nombre de termes : ");
    scanf("%d", &n);
    i=1;
    s=0;

    while(i<=n)
    {
        s = s + (float)1/i;
        i=i+2;
    }
    printf("La somme de %d termes de la suite est %f\n", n, s);
}

```

5)  $1 - 1/3 + 1/5 - \dots \pm 1/n$

```

#include<stdio.h>
int main ()
{
    int n,i,signe;
    float s;
    printf("Entrez le nombre de termes : ");
    scanf("%d", &n);
    i=1;
    s=0;
    signe=1;

    while(i<=n)
    {
        s = s + signe*(float)1/i;
        signe = signe*(-1);
        i=i+2;
    }
    printf("La somme de %d termes de la suite est %f\n", n, s);
}

```

6)  $x + x^2/2 + x^3/3 + \dots x^n/n$

```

#include<stdio.h>
int main ()
{
    int n,i,x,p;
    float s;
    printf("Entrez la valeur de x : ");
    scanf("%d", &x);
    printf("Entrez le nombre de termes : ");
    scanf("%d", &n);

    s=0;
    i=1;
    p=1;

    while(i<=n)
    {
        p=p*x;
        s = s + (float)p/i;
        i=i+1;
    }
    printf("S = %f\n", s);
}

```

7)  $1 + x + x^2/2! + x^3/3! + \dots + x^n/n!$

```

#include <stdio.h>

void main()
{
    float x,s,p;
    int i,n;
    printf("Entrez la valeur de x : ");
    scanf("%f",&x);
    printf("Entrez le nombre de termes : ");
    scanf("%d",&n);
    s =1; p = 1;
    for (i=1;i<n;i++)
    {
        p = p*x/(float)i;
        s =s+ p;
    }
    printf("La somme de %d termes de la suite est %f\n", n, s);
}

```