

**Module : Algorithmique & Programmation C**  
**Filière Informatique & Gestion d'Entreprises (Première Année)**  
**Année Académique : 2017/2018**

**Solutions de TD/TP N° 3**

**Exercice 1**

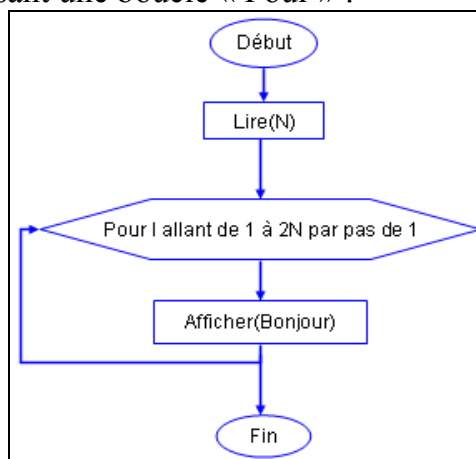
1. La boucle while ne sera pas exécutée.
2. Si la valeur de n est initialisée avec une valeur  $> 100$ , la boucle while ne sera pas exécutée. Si la valeur de n avant le while est  $\leq 100$ , c'est une boucle infinie car la valeur de n reste inchangée.
3.  $x = 33.2$
- 4.

```
i = 1;
while(i <= n)
{
    printf("%d ", i * i);
    i++;
}
```

5. 1 1 3 9 5 9 7 7
6. 1 End of program.  
End of program.  
3 End of program.  
5 End of program.  
End of program.

**Exercice 2**

1. L'organigramme en utilisant une boucle « Pour » :



Le programme C :

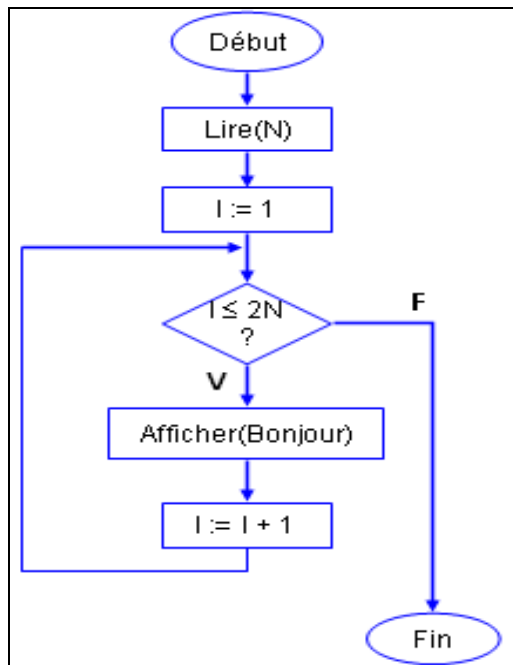
```
#include <stdio.h>
main()
{
    int N, I;
```

```

printf("Donner un entier N : ");
scanf("%d", &N);
for(I = 1; I <= 2 * N; I++)
    printf("Bonjour\n");
getchar();
}

```

2. L'organigramme en utilisant une boucle « Tant Que » :



Le programme C :

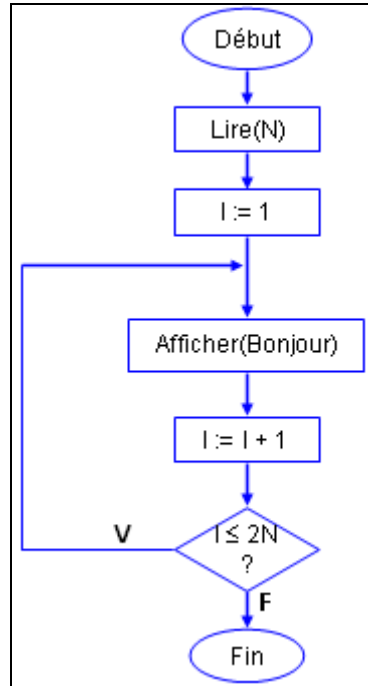
```

#include <stdio.h>
main()
{
    int N, I;
    printf("Donner un entier N : ");
    scanf("%d", &N);
    I = 1;
    while(I <= 2 * N)
    {
        printf("Bonjour\n");
        I++;
    }
    getchar();
}

```

3. L'organigramme en utilisant une boucle « Répéter ... Tant Que » :

**Module : Algorithmique & Programmation C**  
**Filière Informatique & Gestion d'Entreprises (Première Année)**  
**Année Académique : 2017/2018**



4. Traduire l'organigramme en un programme C :

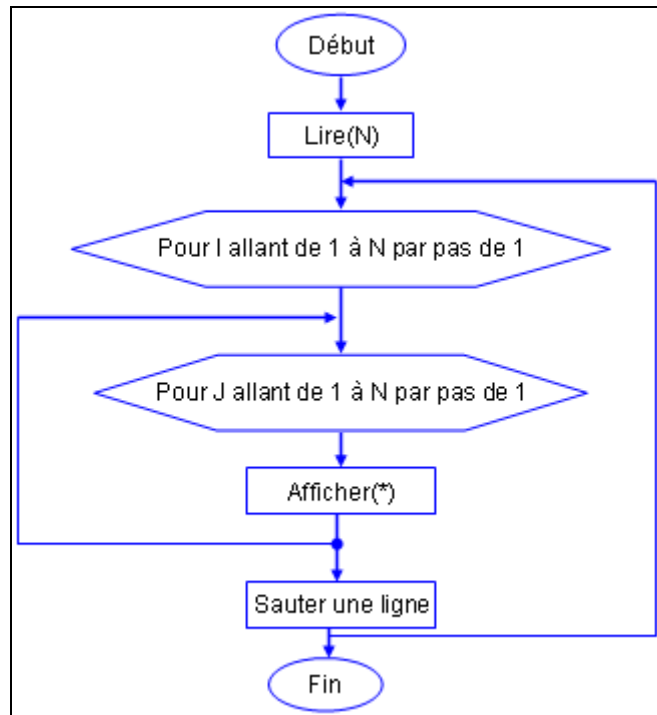
```

#include <stdio.h>
main ()
{
    int N, I;

    printf("Donner un entier N : ");
    scanf("%d", &N);
    I = 1;
    do
    {
        printf("Bonjour\n");
        I++;
    }while(I <= 2 * N);
    getchar();
}
  
```

**Exercice 3**

1. L'organigramme :



2. Le programme C :

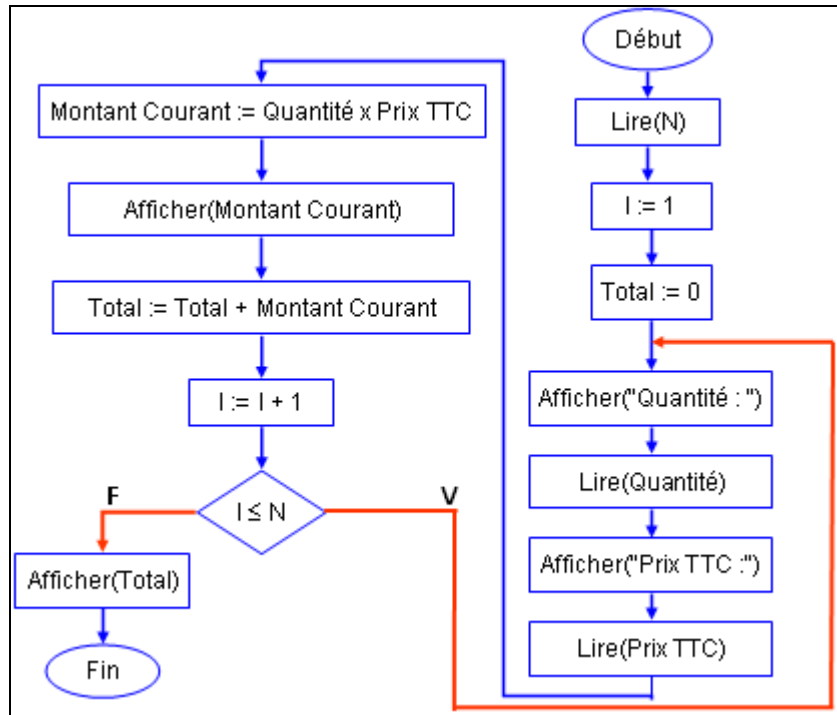
```

#include <stdio.h>
main()
{
    int N, I, J;
    printf("Donner un entier N (N <= 15) : ");
    scanf("%d", &N);
    for(I = 1; I <= N; I++)
    {
        for(J = 1; J <= N; J++)
            putchar('*');
        printf("\n");
    }
    getchar();
}
  
```

### **Exercice 4**

1. L'organigramme :

**Module : Algorithmique & Programmation C**  
**Filière Informatique & Gestion d'Entreprises (Première Année)**  
**Année Académique : 2017/2018**



2. Le programme C :

```

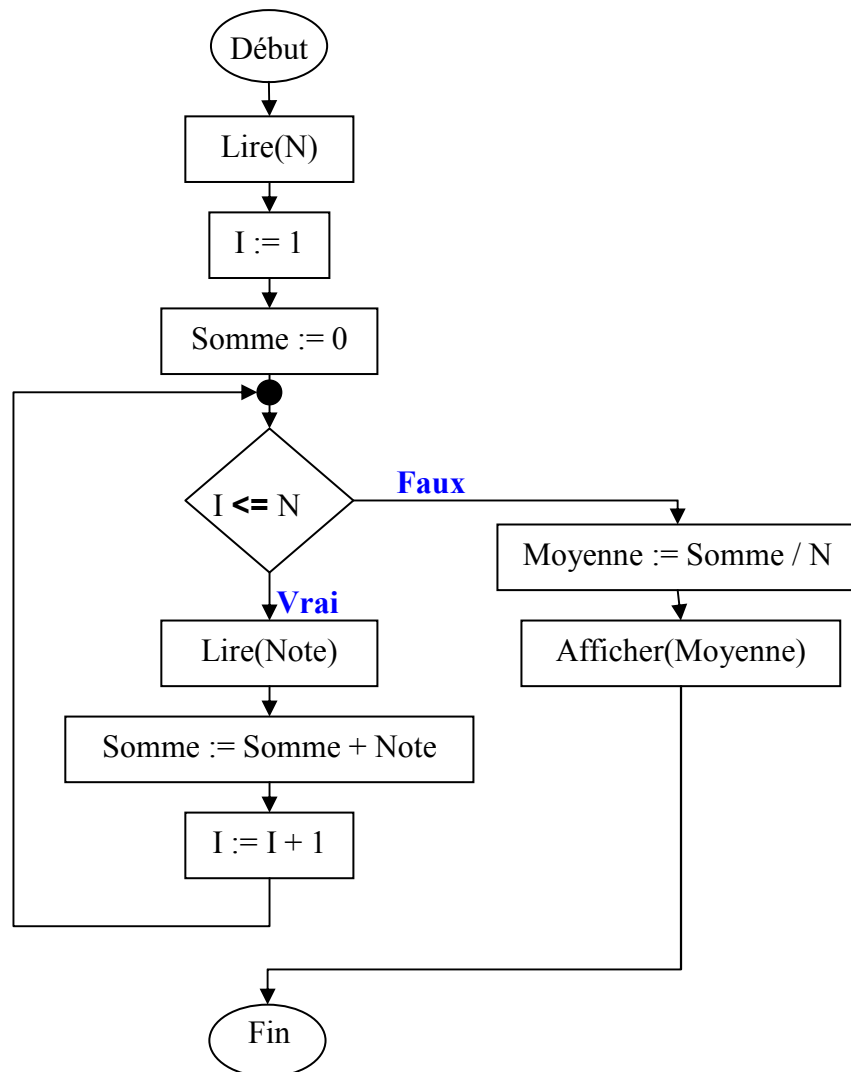
#include <stdio.h>
main()
{
    int N, I;
    float Prix_TTC, Qte, Mt_Courant, Total;
    printf("Donner le nombre de factures a editer : ");
    scanf("%d", &N);
    I = 1;
    Total = 0.0;
    do
    {
        printf("Facture Numero %d\n", I);
        printf("Quantite : ");
        scanf("%f", &Qte);
        printf("Prix TTC : ");
        scanf("%f", &Prix_TTC);
        Mt_Courant = Qte * Prix_TTC;
        printf("Montant Courant : %f\n", Mt_Courant);
        Total += Mt_Courant;
    }
  
```

```

    I++;
}while (I <= N);
printf("\nMontant Total : %f", Total);
system("pause");
}

```

### **Exercice 5**



```

#include <stdio.h>
main()
{
    int I;
    float Note, Somme, Moyenne;

    printf("Donner le nombre des etudiants : ");
    scanf("%d", &N);
    Somme = 0.0;
    for(i = 1; i <= N; i++)
    {
        printf("Note de l'etudiant Numero %d : ", I);
        scanf("%f", &Note);
        Somme += Note;
    }
}

```

**Module : Algorithmique & Programmation C**  
**Filière Informatique & Gestion d'Entreprises (Première Année)**  
**Année Académique : 2017/2018**

```
}  
Moyenne = Somme / N;  
printf("\nMoyenne de la classe : %f\n", Moyenne);  
getchar();  
}
```

*Pour raison de manque de temps, je ne donne pas les organigrammes pour les exercices qui restent.*

### **Exercice 6**

```
#include <stdio.h>  
main()  
{  
    int N = 0;  
    float Note, Somme = 0.0, Moyenne;  
    do  
    {  
        printf("Donner une note : ");  
        scanf("%f", &Note);  
        if(Note != -1)  
        {  
            Somme += Note;  
            N++;  
        }  
        else  
            printf("Arret de la saisie des notes.!!!\n");  
    }while(Note != -1);  
    if(N == 0)  
        printf("Aucune note n'a ete saisie.!!!\n");  
    else  
    {  
        Moyenne = Somme / N;  
        printf("\nMoyenne de la classe : %f\n", Moyenne);  
    }  
    system("pause");  
}
```

### **Exercice 7**

```

#include <stdio.h>
main()
{
    int N, d, somme;
    printf("Donner un nombre entier N : ");
    scanf("%d", &N);
    somme = 0;
    for(d = 1; d <= N; d++)
    {
        if(N % d == 0)
            somme += d;
    }
    if(somme == 2 * N)
        printf("%d est un nombre parfait\n", N);
    else
        printf("%d n'est pas un nombre parfait\n", N);
    system("pause");
}

```

### **Exercice 8**

On lit une valeur M (un entier). On voudrait calculer le plus grand entier N tel que :

$$1 + 2 + \dots + N \leq M$$

Donner l'organigramme, puis le programme C qui permet de réaliser cette tâche.

```

#include <stdio.h>
main()
{
    int M, N, somme;
    printf("Donner un nombre entier M : ");
    scanf("%d", &M);
    somme = 0;
    N = 0;
    while(somme <= M)
    {
        N++;
        somme = somme + N;
    }
    printf("Le N cherche est = %d\n", N - 1);
    system("pause");
}

```