

# TD1/2/3 - Rappels de C

## Exercice 1 - Introduction (tous)

1. Écrire le programme C le plus court possible.
2. En utilisant la directive préprocesseur *define*, créer l'identificateur *PI* et lui associer la valeur *3.14*.
3. Ajouter la déclaration d'une variable entière dont l'identificateur sera *rayon*.
4. Initialiser la variable *rayon* avec la valeur 10, puis calculer la surface du cercle associé en stockant le résultat dans une variable dont l'identificateur sera *surface*.
5. Demander de saisir deux nombres entiers *x* et *y* et les afficher. Inverser leurs valeurs et les afficher de nouveau.
6. Déclarer une variable de type caractère et l'initialiser avec le caractère '7'. Affecter ensuite la valeur numérique 98 à cette variable. Afficher la valeur de la variable après l'initialisation et après l'affectation.

## Solution de l'exercice 1

```
#define PI 3.14

int main() {
    int rayon = 10;
    float resultat = PI*rayon*rayon;
    return 0;
}
```

```
#include <stdio.h>

int main() {
    int x, y, buffer;
    printf("Veuillez saisir deux nombres entiers\n");
    scanf("%d %d", &x, &y);

    buffer = x;
    x = y;
    y = buffer;
    printf("x vaut %d et y vaut %d\n", x, y);

    return 0;
}
```

```
#include <stdio.h>

int main() {

    char x = '7';
    printf("Le code ascii du caractere %c est %d\n", x, x);
    x = 98;
    printf("Le caractere correspondant au code ascii %d est %c\n", x, x);

    return 0;
}
```

## Exercice 2 - Entrées/Sorties (tous)

1. Écrire sur la sortie standard la chaîne de caractères "Bonjour!".
2. Ajouter un saut de ligne avant et après "Bonjour!".
3. Afficher "entrez un caractère : " et laisser l'utilisateur saisir un caractère. Sauter deux lignes et afficher sur une ligne "signe : *<le caractère>*, code ascii décimal : *<le code ascii décimal>*, code ascii hexadécimal : *<le code ascii hexadécimal>*".
4. Lire un entier et l'afficher en décimal, en octal et en hexadécimal.
5. Lire deux réels et les afficher. Calculer la somme et l'afficher.

## Solution de l'exercice 2

```
#include <stdio.h>

int main() {
    printf("\nBonjour !\n");

    return 0;
}
```

```
#include <stdio.h>

int main() {

    char x;

    printf("Entrez un caractere...\n");
    scanf("%c", &x);
    printf("\n\nsigne : %c, code ascii decimal : %d, code ascii hexa : %x", x,
        x,x);

    return 0;
}
```

```
#include <stdio.h>

int main() {

    int x;

    printf("Entrez un entier...\n");
    scanf("%d", &x);
    printf("\n\ndecimal : %d, hexadecimal : %x, octal : %o", x, x,x);

    return 0;
}
```

```
#include <stdio.h>

int main() {

    float a, b, somme;
    printf("Entrez deux reels...\n");
    scanf("%f %f", &a, &b);
    somme = a+b;
    printf("\na vaut %f et b vaut %f.\nLa somme vaut %f\n", a, b, somme);

    return 0;
}
```

## Exercice 3 - Structures conditionnelles (1 à 5)

1. Afficher un message si un nombre entier saisi est compris entre 0 et 10.
2. Afficher un message si un nombre entier saisi est compris entre 7 et 20 ou si il vaut 127.
3. Afficher un message si un nombre entier saisi n'est pas nul.
4. Afficher un message si un caractère saisi est un chiffre et un message différent si c'est une lettre de l'alphabet (minuscule ou majuscule).
5. Lire un caractère (*getchar()*). Afficher "un" si c'est le caractère '1', "deux" si c'est '2', "trois" si c'est '3' et "autre caractère" sinon.
6. Lire une année de naissance. Calculer l'âge et afficher "Pour l'instant ça va" si celui-ci est plus petit ou égal à 50 ans et "Vous êtes sur la mauvaise pente" si il dépasse 50 ans.
7. Ajouter un test logique qui vérifie que l'âge est compris entre 0 et 130 inclus. Si c'est le cas alors effectuer le traitement de l'exercice précédent. Sinon afficher le message "age saisi invalide".
8. Lire un caractère. Afficher "c'est un bon caractère" si ce caractère est 'n', 'N', 'f', 'F', '1' ou '6'.
9. Vérifier si un nombre entier saisi est pair ou impair.

## Solution de l'exercice 3

```
#include <stdio.h>

int main() {

    int nb;
    printf("Veuillez saisir un nombre...\n");
    scanf("%d", &nb);

    if(nb>=0 && nb <=10) {
        printf("Le nombre %d est entre 0 et 10\n", nb);
    } else {
        printf("Le nombre %d n'est pas entre 0 et 10\n", nb);
    }

    return 0;
}
```

```
#include <stdio.h>

int main() {

    int nb;
    printf("Veuillez saisir un nombre...\n");
    scanf("%d", &nb);
    if((nb>=7 && nb <=20) || nb == 127) {
        printf("Le nombre %d est OK\n", nb);
    } else {
        printf("Le nombre %d est KO\n", nb);
    }

    return 0;
}
```

```
#include <stdio.h>

int main() {

    int nb;
    printf("Veuillez saisir un nombre...\n");
    scanf("%d", &nb);
```

```

    if(nb != 0)
        printf("Le nombre %d est OK\n", nb);
    else
        printf("Le nombre %d est KO\n", nb);

    return 0;
}

```

```

#include <stdio.h>

int main() {

    char x;
    printf("Veuillez saisir un caractere...\n");
    scanf("%c", &x);
    if(x >= '0' && x <= '9') {
        printf("C'est un nombre\n");
    } else if(x >= 'a' && x <= 'z') {
        printf("C'est entre a et z\n");
    } else if(x >= 'A' && x <= 'Z') {
        printf("C'est entre A et Z\n");
    }

    return 0;
}

```

```

#include <stdio.h>

int main() {

    printf("Veuillez saisir un caractere...\n");
    switch(getchar()) {
        case '1':
            printf("C'est un\n");
            break;
        case '2':
            printf("C'est deux\n");
            break;
        case '3':
            printf("C'est trois\n");
            break;
        default:
            printf("C'est autre chose...\n");
    }

    return 0;
}

```

```

#include <stdio.h>

#define CURRENT_YEAR 2017

int main() {

    int bornYear, age;
    printf("Veuillez donner votre annee de naissance...\n");
    scanf("%d", &bornYear);
    age = CURRENT_YEAR - bornYear;
    if(age <= 50) {

```

```

    printf("Pour l'instant, ca va\n");
} else {
    printf("Vous etes sur la mauvaise pente\n");
}
return 0;
}

```

```

#include <stdio.h>

#define CURRENT_YEAR 2017

int main() {

    int bornYear, age;
    printf("Veuillez donner votre annee de naissance...\n");
    scanf("%d", &bornYear);
    age = CURRENT_YEAR - bornYear;
    if(age >= 0 && age <= 130) {
        if(age <= 50) {
            printf("Pour l'instant, ca va\n");
        } else {
            printf("Vous etes sur la mauvaise pente\n");
        }
    } else {
        printf("L'age renseigne n'est pas valide...\n");
    }

    return 0;
}

```

```

#include <stdio.h>

int main() {

    printf("Veuillez rentrer un caractere...\n");

    switch(getchar()) {
        case 'n':
        case 'N':
        case 'f':
        case 'F':
        case '1':
        case '6':
        case ' ':
        case 'a':
        case 'A':
        case 'u':
            printf("C'est un bon caractere\n");
            break;
    }

    return 0;
}

```

```

#include <stdio.h>

```

```

int main() {

```

```

int var;

printf("Veuillez rentrer un entier...\n");
scanf("%d", &var);

if(var == 0) {
    printf("L'entier renseigne vaut zero\n");
} else if(var%2 == 0) {
    printf("L'entier %d est pair\n", var);
} else {
    printf("L'entier %d est impair\n", var);
}

return 0;
}

```

## Exercice 4 - Expressions, opérateurs et conversions (1 à 8)

1. Initialiser une variable de type réel avec l'expression  $5/2$  et afficher sa valeur. Étudier les conversions automatiques de type.
2. Déclarer les variables entières  $a$ ,  $b$ ,  $c$ . Demander à l'utilisateur de saisir les valeurs et les afficher.
3. Ajouter le calcul de la somme des valeurs dans une variable *somme* et l'afficher.
4. Ajouter le calcul de la moyenne dans une variable *moyenne* et l'afficher.
5. Ajouter le calcul du produit des nombres dans une variable *produit* et l'afficher.
6. Calculer et afficher le nombre de caractères compris entre le 'a' et le 'A' dans la table des codes ASCII
7. Demander de saisir un caractère. Le transformer en majuscule si c'est une lettre de l'alphabet minuscule (le laisser inchangé sinon) et l'afficher.
8. Déclarer les variables entières  $a$ ,  $b$  et  $c$ . Affecter à  $c$  la valeur 7, incrémenter  $c$ , affecter à  $a$  la valeur de  $c$ , incrémenter  $c$ , décrémenter  $c$ , affecter à  $b$  la valeur de  $c$ , décrémenter  $b$ , affecter à  $a$  la valeur de  $b$ . Que valent  $a$ ,  $b$  et  $c$ ?
9. Lire un nombre entier et indiquer si il est divisible par 7.
10. Lire un nombre entier représentant une température exprimée en degrés Fahrenheit. Convertir en degrés centigrade et afficher les deux températures. La formule mathématique de conversion est :  $\frac{5}{9}(temp\_farh - 32)$ .
11. Ajouter un affichage avec 3 chiffres avant la virgule et 1 chiffre après la virgule.
12. Tester si une année est bissextile. On rappelle qu'il y a aussi une année bissextile tous les 400 ans.

## Solution de l'exercice 4

```

#include <stdio.h>

int main() {
    int a,b,c,result, prod;
    float moyen;
    printf("Veuillez saisir 3 entiers...\n");
    scanf("%d %d %d", &a, &b, &c);
    result = a+b+c;
    printf("%d + %d + %d = %d\n", a, b, c, result);
    moyen = result/3.; // Attention a ne pas faire result/3 => entier
    printf("La moyenne vaut %f\n", moyen);
    prod = a*b*c;
    printf("Le produit vaut %d\n", prod);
    return 0;
}

```

```
#include <stdio.h>

int main() {
    char diff = 'a' - 'A';
    printf("La diff est de %d\n", diff);
    printf("La diff est de %d\n", 'a' - 'A');
    return 0;
}
```

```
#include <stdio.h>

int main() {

    char x;
    char diff = 'a' - 'A';
    printf("Veuillez saisir un caractere....\n");
    scanf("%c", &x);
    if(x >= 'a' && x <= 'z') {
        x -= 'a' - 'A';
    }
    printf("Le caractere apres modification est %c\n", x);
    return 0;
}
```

```
#include <stdio.h>

int main() {
    int a,b,c;
    c = 7;
    c++;
    a=c;
    c++;
    c--;
    b=c;
    b--;
    a=b;
    printf("a vaut %d, b vaut %d, c vaut %d\n", a, b, c);
    return 0;
}
```

```
#include <stdio.h>

int main() {
    int var;
    printf("Veuillez saisir un entier...\n");
    scanf("%d", &var);
    if(var%7==0) {
        printf("%d est divisible par 7\n", var);
    } else {
        printf("Le reste de la division de %d par 7 est %d\n", var, var%7);
    }
    return 0;
}
```

```
#include <stdio.h>

int main() {
    int far;
```

```

    printf("Veuillez saisir une temperature en degres Fahrenheit (entier)...\n"
    );
    scanf("%d", &far);
    float centig = 5/9.*(far-32);
    printf("Far : %d\nCentig : %f\n", far, centig);
    return 0;
}

```

```

#include <stdio.h>

int main() {
    int far;
    printf("Veuillez saisir une temperature en degre Fahrenheit (entier)...\n")
    ;
    scanf("%d", &far);
    float centig = 5/9.*(far-32);
    // largeur/gabarit : 5 => nombre min de caracteres affiches
    // precision : 1 => nombre de chiffres de la partie decimale
    // j'ajoute des * pour bien voir le role du gabarit dans l'affichage du
    nombre
    printf("Far : %d\nCentig : %5.1f*\n", far, centig);
    return 0;
}

```

```

#include <stdio.h>

int main() {
    int year;
    printf("Veuillez saisir une annee...\n");
    scanf("%d", &year);
    if((year%4==0 && year%100 != 0) || year%400==0) {
        printf("L'annee %d est bissextile\n", year);
    } else {
        printf("L'annee %d n'est pas bissextile\n", year);
    }
    return 0;
}

```

## Exercice 5 - Structures itératives (1 à 5)

1. Afficher les nombres de 0 à 9 avec une boucle *for*, puis une boucle *do...while*, puis une boucle *while*.
2. Afficher la table des codes ASCII pour les caractères au delà de la valeur 31 et jusqu'à la valeur 127.
3. Afficher les caractères compris entre 'A' et 'Z' puis entre 'a' et 'z'. Afficher le caractère, le code ASCII en décimal et le code ASCII en hexadécimal.
4. À l'aide d'une instruction *switch* et de la fonction *getchar()* : demander à l'utilisateur de saisir un caractère. Si le caractère est compris dans l'ensemble ('o', 'O', 'n', 'N', '0', '1', '2', '3'), afficher "c'est un bon caractère", sinon afficher "c'est un mauvais caractère ". Répéter la demande jusqu'à ce que l'utilisateur saisisse le caractère 'n' ou 'N'.
5. Calculer la fonction factorielle définie par  $F_0 = 1$ ;  $F_n = n * F_{n-1}$ .
6. Calculer la suite définie par  $S_0 = 1$ ;  $S_n = S_{n-1} + 7$ .
7. Lire des caractères et les afficher jusqu'à ce que l'utilisateur saisisse le caractère '\n'. Afficher le nombre de caractères lus.
8. Calculer les intérêts cumulés à partir d'une somme  $s$ , sur  $n$  périodes de temps et avec un taux d'intérêt de  $t$ .
9. Lire une suite d'entiers strictement positifs et s'arrêter lorsque l'utilisateur rentre la valeur -1. Calculer la moyenne des valeurs saisies et l'afficher.



10. Afficher "Quelle table de multiplication voulez-vous afficher (tapez 0 pour sortir) ?". Laisser l'utilisateur saisir un caractère. Si le caractère est compris entre '1' et '9' alors afficher la table correspondante puis réitérer le processus. Sinon afficher "ce n'est pas dans les possibilités du programme, recommencez !" et réitérer le processus.
11. Modifier le programme précédent pour travailler en base 16 (hexadécimal). On rappelle que les digits licites en base 16 sont ('0', '1', '2', '3', '4', '5', '6', '7', '8', '9', 'A', 'B', 'C', 'D', 'E', 'F'). Veiller à traiter convenablement le cas des caractères.

## Solution de l'exercice 5

```
#include <stdio.h>
int main() {
    int i;
    for(i=0; i<=9; i++) {
        printf("Boucle for : %d\n", i);
    }
    printf("Après la boucle for, i vaut %d\n", i);
    i = 0;
    do {
        printf("Boucle do/while : %d\n", i);
        i++;
    } while(i<=9);
    printf("Après la boucle do/while, i vaut %d\n", i);
    i=0;
    while(i<=9) {
        printf("Boucle while : %d\n", i);
        i++;
    }
    printf("Après la boucle while, i vaut %d\n", i);
    return 0;
}
```

```
#include <stdio.h>

int main() {

    char i;
    for(i=31; i<127; i++) {
        printf("Le code ASCII %d correspond au caractere %c\n", i, i);
    }

    return 0;
}
```

```
#include <stdio.h>

int main() {
    char i;
    for(i='a'; i<'z'; i++) {
        printf("caractere %c, decimal : %d, hexa : %d\n", i, i, i);
    }
    for(i='A'; i<'Z'; i++) {
        printf("caractere %c, decimal : %d, hexa : %d\n", i, i, i);
    }
    return 0;
}
```

```
#include <stdio.h>
```

```

int main() {
    char c;
    do {
        printf("Veuillez saisir un caractere...\n");
        c = getchar();
        while(getchar() != '\n'); // Vide le buffer
        switch(c) {
            case 'o':
            case '0':
            case 'n':
            case 'N':
            case '0':
            case '1':
            case '2':
            case '3':
                printf("C'est un bon caractere\n");
                break;
            default:
                printf("C'est un mauvais caractere\n");
        }
    } while(c != 'n' && c != 'N');
    return 0;
}

```

```

#include <stdio.h>

int main() {
    int i,n,result;
    printf("Veuillez saisir un entier....\n");
    scanf("%d", &n);
    result = 1;
    for(i=2; i<=n;i++) {
        result*=i; // result = result*i;
    }
    printf("%d! = %d\n", n, result);
    return 0;
}

```

```

#include <stdio.h>

int main() {
    int i,n,s;
    printf("Veuillez saisir un entier....\n");
    scanf("%d", &n);
    s = 1;
    for(i=2; i<=n;i++) {
        s+=7;
    }
    printf("s(%d) = %d\n", i, s);
    return 0;
}

```

```

#include <stdio.h>
#include <string.h>
#define NMAX 100

int main() {
    char chaine[NMAX];
    printf("Veuillez saisir une chaine...\n");
    fgets(chaine, NMAX, stdin);
    printf("%s\n", chaine);
}

```

```

int i = 0;
while(chaine[i] != '\0') {
    printf("%c\n", chaine[i]);
    i++;
}
printf("La chaine fait %d caracteres\n", i);
printf("La chaine fait %d caracteres\n", strlen(chaine));
return 0;
}

```

```

#include <stdio.h>

int main() {
    int var, somme, count;
    float moyen;
    somme = 0;
    count = 0;
    do {
        printf("Veuillez saisir un entier strictement positif...\n");
        scanf("%d", &var);
        if(var > 0) {
            printf("Valeur saisie : %d\n", var);
            count++;
            somme+=var;
        } else if(var != -1) {
            printf("Erreur, %d n'est pas un entier strictement positif\n", var);
        }
    } while(var != -1);
    moyen = (float)somme/count;
    printf("La moyenne est de %f\n", moyen);
    return 0;
}

```

```

#include <stdio.h>

int main() {
    char x;
    int tab,i;
    do {
        printf("Quelle table de multiplication (tapez 0 pour sortir) ?\n");
        x = getchar();
        while(getchar() != '\n'); // On vide le buffer
        if(x > '0' && x <= '9') {
            tab = x - '0'; // recup l'entier correspondant au car
            for(i=0; i<=9; i++) {
                printf("%d * %d = %d\n", i, tab, i*tab);
            }
        } else if(x != '0') {
            printf("Ce n'est pas dans les possibilites du programme\n");
        }
    } while(x != '0');
    return 0;
}

```

## Exercice 6 - Pointeurs (tous)

1. Déclarer une variable *i* de type entier et l'initialiser à 5. Afficher l'adresse de *i* et la valeur de *i*.