

# Calculatrice en C

## Avec des conditions (if) :

**Boucle avec la condition d'arrêt tant que fin n'est pas égal à zéro la boucle ne s'arrête pas.**

**Affichage des instructions à suivre**

**Affichage du calcul est du résultat selon l'opération : addition, soustraction, multiplication et modulo (reste de la division)**

**Calcul du carré soit avec la méthode « pow ». Soit avec une boucle.**

**Calcul de la racine carrée avec la méthode « sqrt »**

**Calcul de la valeur absolue soit avec la méthode « abs ». Soit avec des conditions.**

**Si l'utilisateur rentre « 0 », la boucle s'arrête. Sinon c'est reparti pour un tour.**

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

int main()
{
    int nb1, nb2;
    char op, fin = 1;

    printf ("Ce programme effectue l'operation des deux nombres saisie par l'utilisateur\n");

    while (fin != '0')
    {
        printf ("\nSaisir le 1er nombre, l operation voulue puis le 2eme nombre\n( addition : + | soustraction : - | multiplication : * | division : / |\n puissance : ^ | modulo : % | racine carre : r | valeur absolue : a ) :");
        scanf ("%d %c%d",&nb1,&op,&nb2); //fflush(stdin);

        if (op=='+')
            printf ("La somme de %d + %d est %d \n", nb1, nb2, (nb1+nb2));
        else if (op=='-')
            printf ("La difference de %d - %d est %d \n", nb1, nb2, (nb1-nb2));
        else if (op=='*')
            printf ("Le produit de %d x %d est %d \n", nb1, nb2, (nb1*nb2));
        else if (op=='/')
        {
            float a;
            a = nb1;
            printf ("Le quotient de %d / %d est %.2f \n", nb1, nb2, (a/nb2));
        }
        else if (op=='%')
            printf ("Le resultat de %d modulo %d est %d \n", nb1, nb2, nb1%nb2);
        else if (op=='^')
        {
            //printf ("La puissance de %d ^ %d est %.0f \n", nb1, nb2, pow(nb1,nb2)); ou
            int a, cpt;
            a = nb1;
            cpt = 1;
            while (cpt<nb2)
            {
                nb1 = (nb1*a);
                cpt = (cpt+1);
            }
            printf ("La puissance de %d ^ %d est %d \n", a, nb2, nb1);
        }
        else if (op=='r')
            printf ("La racine carre de %d est %.2lf et de %d est %.2lf \n", nb1, sqrt(nb1), nb2, sqrt(nb2));
        else if (op=='a')
        {
            //printf ("la valeur absolue de %d est %d et de %d est %d \n", nb1, abs(nb1), nb2, abs(nb2)); ou
            if ((nb1<0) && (nb2<0))
                printf ("la valeur absolue de %d est %d et de %d est %d",nb1,(nb1*(-1)),nb2,(nb2*(-1)));
            else if ((nb1<0) && (nb2>0))
                printf ("la valeur absolue de %d est %d et de %d est %d",nb1,(nb1*(-1)),nb2,nb2);
            else if ((nb1>0) && (nb2<0))
                printf ("la valeur absolue de %d est %d et de %d est %d",nb1,nb1,nb2,(nb2*(-1)));
            else
                printf ("la valeur absolue de %d est %d et de %d est %d",nb1,nb1,nb2,nb2);
        }
        else
            printf ("ERREUR\n");

        printf ("\nSaisir 0 si vous voulez arreter ou taper autre chose pour continuer\n");
        scanf (" %c", &fin); //fflush(stdin);
    }

    printf ("Fin\n");
    return 0;
}
```

**Importer les bibliothèques (fichiers) utiles**

**Déclaration des variables : les 2 nombres en integer (nombre), l'opération en char (caractère) comme « fin » qui sera notre condition d'arrêt du programme.**

**Affichage du contexte**

**Stockage des valeurs saisi dans les variables**

**Même chose, mais il a fallu utiliser une variable en nombre réel (float) pour stocker le résultat de la division qui ducoup n'était pas forcément un int.**

**On crée une variable compteur (incrémenter d'un à chaque tour de boucle) qui sera notre condition d'arrêt et a pour pouvoir afficher le premier nombre non modifier. Et on multiplie le nombre 1 par lui-même le nombre de fois du nombre 2.**

**Affichage des nombres sans modification s'il son positif et multiplier par -1 s'ils sont négatif (pour les passer en positif).**

**Si l'opérateur n'est pas valide**

**Affichage du message « Fin ». return 0 : fin du programme.**

**\* : #include : pour ajouter le contenu/morceau de code d'un fichier d'en-tête réservé dans notre fichier de code avant la compilation de notre programme C.**

## Vu du terminal à l'exécution :

```
Ce programme effectue l'operation des deux nombres saisis par l'utilisateur

Saisir le 1er nombre, l'operation voulue puis le 2eme nombre
( addition : + | soustraction : - | multiplication : * | division : / |
  puissance : ^ | modulo : % | racine carree : r | valeur absolue : a ) :
```

## En pseudo code :

```
PROGRAMME calculatrice

VAR
    nb1, nb2 : entier
    op, fin = 1: caractère

Debut

    Afficher "Ce programme effectue l'operation des deux nombres saisis par l'utilisateur"

    TantQue fin <> '0' Faire
        Afficher "Saisir le 1er nombre, l'operation voulue puis le 2eme nombre ( addition : + | soustraction : - | multiplication : * | division : /
        | puissance : ^ | modulo : % | racine carree : r | valeur absolue : a ) : "
        Saisir nb1, op, nb2

        Si op = '+' Alors
            Afficher "La somme de" nb1 "+" nb2 "est" nb1+nb2
        Sinon Si op = "-" Alors
            Afficher "La difference de" nb1 "-" nb2 "est" nb1-nb2
        Sinon Si op = "*" Alors
            Afficher "Le produit de" nb1 "x" nb2 "est" nb1*nb2
        Sinon Si op = "/" Alors
            Afficher "Le quotient de" nb1 "/" nb2 "est" nb1/nb2
        Sinon Si op = "%" Alors
            Afficher "Le resultat de" nb1 "modulo" nb2 "est" nb1%nb2
        Sinon Si op = "^" Alors
            a, cpt : entier
            a = nb1
            cpt = 1
            TantQue cpt<nb2
                nb1 = nb1*a
                cpt = cpt+1
            FinTantQue
            Afficher "La puissance de" a "^" nb2 "est" nb1
        Sinon Si op = "r" Alors
            Afficher "La racine carree de" nb1 "est" racineCarrée(nb1) "et de" nb2 "est" racineCarrée(nb2)
        Sinon Si op = "a" Alors
            Si nb1<0 et nb2<0 Alors
                Afficher "la valeur absolue de" nb1 "est" nb1*(-1) "et de" nb2 "est" nb2*(-1)
            Sinon si nb1<0 et nb2>0 Alors
                Afficher "la valeur absolue de" nb1 "est" nb1*(-1) "et de" nb2 "est" nb2
            Sinon si nb>0 et nb2<0 Alors
                Afficher "la valeur absolue de" nb1 "est" nb1 "et de" nb2 "est" nb2*(-1)
            Sinon
                Afficher "la valeur absolue de" nb1 "est" nb1 "et de" nb2 "est" nb2
            Finsi
        Sinon
            Afficher "ERREUR"
        Finsi

        Afficher "Saisir 0 si vous voulez arreter ou autre chose pour continuer"
        Saisir fin
        Afficher "Fin"
    Fin

Fin
```

## Avec des switch (case) :

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

int main()
{
    int nb1, nb2;
    char op, fin = 1;

    printf ("Ce programme effectue l'operation des deux nombres saisi par l'utilisateur\n");

    while (fin != '0')
    {
        printf ("\nSaisir le 1er nombre, l operation voulue puis le 2eme nombre\n( addition : + | soustraction : - | multiplication : * |  
division : / |\n puissance : ^ | modulo : % | racine carre : r | valeur absolue : a ) :");
        scanf ("%d %c%d",&nb1,&op,&nb2);    //fflush(stdin);

        switch(op)
        {
            case'+':printf ("La somme de %d + %d est %d \n", nb1, nb2, nb1+nb2);break;
            case'-':printf ("La difference de %d - %d est %d \n", nb1, nb2, nb1-nb2);break;
            case'*':printf ("La produit de %d x %d est %d \n", nb1, nb2, nb1*nb2);break;
            case'/':printf ("La quotient de %d / %d est %d \n", nb1, nb2, nb1/nb2);break;
            case'%':printf ("Le resultat de %d modulo %d est %d \n", nb1, nb2, nb1%nb2);break;
            case'^':printf ("La puissance de %d ^ %d est %.0f \n", nb1, nb2, pow(nb1,nb2));break;
            case'r':printf ("La racine carre de %d est %.2lf et de %d est %.2lf \n", nb1, sqrt(nb1), nb2, sqrt(nb2));break;
            case'a':printf ("La valeur absolue de %d est %d et de %d est %d \n", nb1, abs(nb1), nb2, abs(nb2));break;
            default:printf ("ERREUR");
        }

        printf ("\nSaisir 0 si vous voulez arreter ou taper autre chose pour continuer\n");
        scanf ("%c", &fin);    //fflush(stdin);
    }

    printf ("Fin\n");
    return 0;
}
```