

Les fonctions en C++

Exercice 1 :

```
#include <iostream>
using namespace std ;

int fct (int p) {
    return 2*p ;
}

int main() {
    int n, p=5;
    n=fct(p);
    cout<<"p="<<p<<"n="<<n;
}
```

On remplace les variables **r** par les variables **p**.
On en profite également pour remplacer la fonction **fct** au-dessus du **main**

Exercice 2 :

```
#include <iostream>
using namespace std;

void f1() {
    cout<<"Bonjour";
}

void f2() {
    for (int x=0 ; x<5 ; x++) {
        cout<<"Bonjour";
    }
}

int f3() {
    for (int x=0 ; x<5 ; x++) {
        cout<<"Bonjour";
    }
    return 0;
}

int main() {
    f1();
    cout<<endl;
    f2();
    cout<<endl;
    f3();
    cout<<endl;
}
```

Création des trois fonctions

Main (cœur du programme)
Appel des 3 fonctions

Exercice 3 :

Ce programme renverra le message suivant :

B : dans fct , n = 10 p = 5 q = 20

A : dans main, n = 20 p = 5 q = 2

C : dans f , n = 10 p = 20 q = 2

Exercices 4 et 5 :

Ces deux exercices étant similaire, je les ai regroupés en un programme.

```
#include <iostream>
using namespace std;
```

```
char c;
```

Variable globale

```
void calcul(float a, float b) {
    switch (c) {
        case '+':
            cout<<a<<"+"<<b<<"="<<a+b<<endl;
            break;
        case '-':
            cout<<a<<"-"<<b<<"="<<a-b<<endl;
            break;
        case '*':
            cout<<a<<"*"<<b<<"="<<a*b<<endl;
            break;
        case '/':
            cout<<a<<"/"<<b<<"="<<a/b<<endl;
            break;
        default:
            cout<<"Erreur de saisie";
    }
}
```

La fonction **calcul** reçoit en argument les deux variables **float**

Affichage et calcul

```
int main() {
    cout<<"Saisir l'opération à effectuer (+, -, *, /) : ";
    cin>>c;
    calcul(10,2);
}
```

Dans le **main**, on demande à l'utilisateur de saisir l'opération à effectuer

Exercice 6 :

```
#include <iostream>
using namespace std;
```

```
static int n=10, i=1;
```

On créer deux variables, i correspond à la valeur de départ et n à la valeur finale

```
void nombreFois() {
    cout<<"Appel numéro "<<i<<endl;
}
```

Fonction à appeler n fois

```
int main() {
    for (i ; i<=n ; i++) {
        nombreFois();
    }
}
```

Ainsi, on appel (dans notre cas) 10 fois la fonction **nombreFois**

RESULTAT :

Appel numéro 1
Appel numéro 2
Appel numéro 3
Appel numéro 4
Appel numéro 5
Appel numéro 6
Appel numéro 7
Appel numéro 8
Appel numéro 9
Appel numéro 10

Exercice 7 :

```
using namespace std;
```

```
static int nombre;
```

Création d'une variable globale **nombre**

```
int nombrePair(int nombre) {  
    if (nombre%2==0) {  
        cout<<"il est pair"<<endl;  
    }  
    return 0;  
}
```

Si le nombre%2=0 (% calcul le reste de la division), alors le nombre est pair

```
int division(int nombre) {  
    if (nombre%3==0) {  
        cout<<"il est divisible par 3"<<endl;  
    }  
    if (nombre%6==0) {  
        cout<<"il est divisible par 6"<<endl;  
    }  
    return 0;  
}
```

Si le nombre%3=0 alors nombre est divisible par 3
Si le nombre%6=0 alors nombre est divisible par 6

```
int main() {  
    cout<<"entrez un nombre entier : ";  
    cin>>nombre;  
    cout<<endl;  
    nombrePair(nombre);  
    division(nombre);  
}
```

On demande à l'utilisateur de saisir le nombre qu'il souhaite tester.
Puis, on affiche le résultat.

Exercice 8 :

```
#include <iostream>  
using namespace std;
```

```
int appel(int p) {  
    return 2*p+1;  
}
```

Variable fournie en argument

```
int main() {  
    static int n=0, p=4;  
    n=appel(p);  
    cout<<n<<endl;  
}
```

Valeur fournie en argument

Exercice 9 :

- a. Appel correcte (avec la fonction 1)
- b. Appel correcte (avec la fonction 2)
- c. Appel correcte (avec la fonction 3)
- d. Appel correcte (avec la fonction 4)
- e. Appel incorrecte car aucune des fonctions appellent une variable char
- f. Appel incorrecte car aucune des fonctions appellent deux variables int
- g. Appel incorrecte car aucune des fonctions appellent une variable int et char
- h. Appel incorrecte car aucune des fonctions appellent une variable int et double
- i. Appel incorrecte car aucune des fonctions appellent deux variables double

Exercice 10 :

```
#include <iostream>
using namespace std ;
int main() {
    int fct(char, int);
    int n=150, p;
    char c='s';
    p=fct(c,n);
    cout<<"fct(\"<c<<"\", "<n<<" ) vaut : "<p<<endl;
}
inline int fct(char c, int n) {
    int res;
    if (c=='a') {res=n+c;}
    else if (c=='s') {res=n-c;}
    else {res=n*c;}
    return res;
}
```

La fonction **fct** devient une fonction en ligne grâce à « **inline** »

Si on souhaite que la fonction **fct** soit compilée séparément, il faut retirer « **inline** » lors de la déclaration de la fonction.