

Corrigé de TP N°2 Algorithmique et programmation C++ (DEV11)

Exercice 1. Écrire un programme qui demande la saisie d'un entier n et affiche la lettre C écrite avec des étoiles, n étoiles sur les deux lignes horizontales et n étoiles sur la ligne verticale. Voir ci-dessous un exemple d'exécution.

Donner un entier n : 10

```
*****
*
*
*
*
*
*
*
*****
```

Corrigé:

```
#include <iostream>
using namespace std;
int main() {
    int n;
    cout<<"Veuillez saisir un entier : ";
    cin>>n;
    for(int i=0;i<n;i++){
        cout<<"* ";
    }
    cout<<endl;
    for(int i=0;i<n-2;i++){
        cout<<"*"<<endl;
    }
    for(int i=0;i<n;i++){
        cout<<"* ";
    }
    return 0;
}
```

Exercice 2. Nous rappelons que le développement de Taylor de la fonction e^x est écrit comme suit.

$$e^x = 1 + \frac{x}{1!} + \frac{x^2}{2!} + \frac{x^3}{3!} + \dots, \quad -\infty < x < \infty$$

Écrire un programme qui demande la saisie d'un nombre réel x et affiche une approximation de e^x en considérant les 20 premiers termes de la série ci-dessus.

Corrigé:

```
#include <iostream>
#include <cmath>
using namespace std;
int fact (int n){
    if(n==0 || n==1)
        return 1;
    else return n*fact(n-1);
}
int main(){
    float x,s;
    cout<<" saisir un entier : "<<endl;
    cin>>x;
    s=0;
    for(int i =1;i<20;i++){
        s=s+pow(x,i)/fact(i);
    }
    cout<<"e à la puissance"<<x<<"="<<s<<endl;
    return 0;
}
```

Exercice 3. Écrire un programme qui génère un nombre entier aléatoire N dans l'intervalle [10,200], et il demande à l'utilisateur de le deviner.

Le programme affiche à l'utilisateur une information sur le positionnement du nombre saisi par rapport au nombre à deviner en affichant « Le nombre recherché est plus petit, ou le nombre

recherché est plus grand ». Le programme s'arrête quand l'utilisateur trouve le nombre N ou quand il dépasse 10 essais.

Corrigé:

```
#include <iostream>
#include <stdlib.h>
#include <time.h>
using namespace std;
int main() {
    srand(time(NULL));
    int a= rand()%191+10;
    int n;
    for(int i=0; i<10; i++){
        cout<<"saisir un nombre :"<< endl;
        cin>> n;
        if(n<a)
            cout<<"le nombre est plus petit"<< endl;
        else if(n>a)
            cout<<"le nombre est plus grand"<< endl;
        else{
            cout<<"le nombre est égal:Bravo";
            break;
        }
    }
    return 0;
}
```

Exercice 4. Ecrire un programme C++ qui demande à l'utilisateur un nombre entier et l'affiche inversé, par exemple si l'entrée est 12345 on doit afficher l'entier 54321.

Corrigé:

Méthode 1:

```

#include <iostream>
using namespace std;
int main() {
    int n;
    cout<<"Donner un nombre : ";
    cin>>n;

    int i = 0;
    while(n != 0) {
        i = i*10 + n%10;
        n = n / 10;
    }
    cout<<"l'entier inverse est : "<< i <<endl;
    return 0;
}

```

Méthode 2:

```

#include <iostream>
using namespace std;
int main() {
    string numero;
    string Resultat;
    cout<<"entre votre numero "<<endl;
    cin>> numero;
    for(int i= numero.length(); i>=0 ;i--){
        Resultat += numero[i];
    }
    cout << Resultat <<endl;
    return 0;
}

```

Exercice 5. Un entier est dit premier s'il a exactement deux diviseurs différents 1 et lui-même, Ecrire un programme C++ qui lit un entier puis détermine s'il est premier ou non.

Corrigé:

```
#include <iostream>
using namespace std;
int main() {
    int n;
    cout<<"Donner un entier : ";
    cin>>n;
    while(n <= 0){
        cout<<"Donner un entier : ";
        cin>>n;
    }
    if(n == 1 || n == -1){
        cout<< n <<" n est un nombre premier "<<endl;
        return 0;
    }
    else{
        for(int i=3; i<n; i++){
            if(n%i == 0 ){
                cout<< n <<" n'est pas un nombre premier "<<endl;
                return 0;
            }
        }
        cout<< n <<" est un nombre premier ";
        return 0;
    }
    return 0;
}
```