

Mini Projet C++

06/10/2016

Zixuan FENG

Nalla Adebayo

1D

Table des matières

I- INTRODUCTION.....	3
II- HYPOTHESE DE TRAVAIL ET DIFFICULTES RECONTRES.....	3
HYPOTHESE DE TRAVAIL.....	3
DIFFICULTES RENCONTRES.....	3
III- TEST DU PROGRAMME (JEU D’ESSAIS)	4
IV- BILAN DU TRAVAIL	5
V- ANNEXE.....	5

I- INTRODUCTION

Le but de notre programme est de créer une simulation du jeu FizzBuzz. Ce jeu consiste à compter d'une valeur à une autre. Lors d'une partie de FizzBuzz, nous affrontons l'ordinateur et nous avons 3 vies. Le but est simple : dire Fizz lorsque l'on tombe sur un multiple de 3, dire Buzz lorsque l'on tombe sur un multiple de 5 et dire FizzBuzz lorsque ce nombre est un multiple de 3 et de 5 également. Le jeu s'arrête lorsque nous n'avons plus de vie où lorsque l'on atteint la valeur maximale.

II- HYPOTHESE DE TRAVAIL ET DIFFICULTES RECONTRES

HYPOTHESE DE TRAVAIL

Nous avons utilisé les boucles `do.while` et `if` pour simplifier le programme car dans le cas d'une boucle `for`, il faut connaître le nombre de fois où elle se répète, dans ce cas nous ne le connaissons pas. Alors que quand on utilise une boucle `while`, on utilise que deux fois la valeur d'initialisation. Nous avons distingué la valeur de « commence » car sinon on trouve le même message pour les deux cas, c'est-à-dire le cas où commence est égal à 0 et commence est égal à 1. Nous avons également préféré utiliser `getline` pour permettre aux personnes ayant un nom composé de pouvoir l'écrire correctement.

DIFFICULTES RENCONTRES

1. Nous avons eu du mal à comprendre le terme « concaténation » dans le sens de la programmation à la question.
2. Nous avons eu un problème car nous avons mis `const int = 999999999` et notre programme ne voulait compiler, pourtant il n'y avait aucune erreur. Nous avons donc mis `const int = 1000` pour remédier au problème.
3. Quelques fois, l'ordinateur affichait `max+1` au lieu du `max`.
4. Certaines fois, le programme ne fonctionnait pas correctement. Lorsque l'on tapait 3, nous n'avions pas de Fizz. C'est parce que nous n'avions pas mis les variables `mult3` et `mult5` en `bool`.

III- TEST DU PROGRAMME (JEU D'ESSAIS)

BUT DU TEST	VALEUR D'ENTREE	RESULTAT PREVU
Nous allons commencer une partie de 1 à 10, nous allons rentrer 0 pour voir si c'est vraiment l'ordinateur qui commence	0 pour que l'ordinateur débute	Ordinateur > 1
Nous allons commencer une partie de 1 à 10, nous allons rentrer 1 pour voir si c'est vraiment le joueur qui commence	1 pour que nous débutions	« NOM DE FAMILLE » > 1
Nous allons commencer de 1 à 10 et dire Fizz alors que 1 n'est pas un multiple de 3. Normalement on devrait avoir un message d'erreur.	1	"Vous vous êtes trompé,vous perdez un point de vie"
Nous allons répéter cette erreur 3 fois afin de voir si le nombre de vie fonctionne.	1	«
Nous allons tester la valeur 3, la bonne réponse devrait être Fizz, nous allons donc écrire Fizz	Fizz	On passe au rang suivant
Nous allons tester la valeur 5, la bonne réponse devrait être Buzz, nous allons donc écrire Buzz	Buzz	On passe au rang suivant
Lors d'une partie compris dans l'intervalle [2 ;10] nous allons atteindre 10 pour voir si la partie s'arrete	2	La partie s'arrête
Lorsque la partie se fini, nous allons tester d'appuyer sur 1, normalement nous allons recommencer une nouvelle partie	1	On recommence une partie.

IV- BILAN DU TRAVAIL

Ce travail a été très dur a réalisé mais nous avons quand même réussi à le terminer. Ce projet nous as permis de mieux gérer le passage d'un int vers un string et à mieux gérer le choix entre les différents types de boucles. Il nous a également appris à nous écouter les uns les autres car dans certains cas, nous n'étions pas d'accord sur le type de boucle qu'il fallait utiliser.

V- ANNEXE

//Zixuan FENG & Adebayo NALLA

//Groupe 1D

#include <iostream>

#include <string> //pour pouvoir utiliser getline

#include <cstdio> //pour pouvoir utiliser sprintf

using namespace std;

const int MAXT=1000; //(une constante) la taille du tableau

int main()

{

 //définir des variables

 int vie, valj, valo, nbmin, nbmax, commence, recommence; // valj=valeur du joueur , valo=valeur de l'ordinateur

 string nom, valjs, valjse, valos; //valjs=valeur du joueur au genre de string , valjse=le string pour examiner si 'valjs' est correct

 bool mult3, mult5; //les booléens pour vérifier si le valeur est le multiple de 3 ou 5

 char tabj[MAXT], tabo[MAXT]; //pour la conversion entre int et string

 //la boucle pour recommencer le jeu

 do

 {

 vie=3;

 //saisir le nom du joueur

 cout << "Donnez-moi votre nom de famille , SVP." << endl;

 cin.sync();

 getline(cin, nom);

 //saisir l'intervalle

 cout << nom << " , choisissez deux nombres s'il vous plait." << endl;

 cout << "(Les deux valeur doivent etre strictement positives.)" << endl;

 cin >> nbmin >> nbmax ;

 //examiner si les deux valeurs respectent les contraintes , sinon redemander des nombres

 while (nbmin<=0 || nbmax<=0 || nbmin>=nbmax)

 {

 cout << "Les nombres que vous avez donne ne respectent pas les regles." << endl;

```

        cout <<"Veuillez me redonner d'autres nombres SVP."<<endl;
        cout <<"( Les deux valeur doivent être strictement positives. )" <<endl;
        cin >> nbmin >> nbmax ;
    }

    //demander qui commence le jeu
    cout <<"Bienvenue dans notre simulation du FizzBuzz, voulez-vous
commencer?"<<endl;
    cout <<"Appuyez sur 1 pour debuter, 0 pour que l'ordinateur debute."<<endl;
    cin >>commence;
    //vérifier si la valeur saisie est correcte ( 0 ou 1 ) sinon redemander le nombre
    while ( commence != 0 && commence != 1 )
    {
        cout <<"Le nombre que vous avez importe ne fonctionne pas."<<endl;
        cout <<"Tapez '0' pour que l'ordinateur démarre et tapez '1' pour
demarrez."<<endl;
        cin >> commence;
    }

    //le début du jeu , recommander les règles du jeu
    cout <<"Preparez-vous,le jeu va bientôt commencer."<<endl;
    cout <<"Comptez de 1 en 1 entre "<<nbmin<<" et "<<nbmax<<"."<<endl;
    cout <<"Dites Fizz lorsqu'il y a un multiple de 3, Buzz lorsqu'il y a une multiple de 5
et FizzBuzz lorsque le nombre est un multiple de 3 et 5. Bonne chance !" <<endl;

    //Vérifier qui va commencer le jeu
    // 1° cas --- l'ordinateur commence le jeu
    if ( commence==0 )
    {
        //saisir l'initialisation
        valo=nbmin;
        //conversion de int vers string
        sprintf( tabo,"%d",valo );
        valos=tabo;
        //vérifier si 'valo' est le multiple de 3 ou 5
        mult3 = (valo%3==0) ;
        mult5 = (valo%5==0) ;
        //si 'valo' est le multiple de 3 ou 5 , on change 'valos'(le message affiché) à
Fizz/Buzz/FizzBuzz
        //utilisant la concaténation pour générer le mot FizzBuzz
        if ( mult3 )
        {
            valos="Fizz";
        }
        if ( mult5 )
        {
            if ( mult3 )
            {
                valos="Fizz";
            }
            else
            {

```

```

        valos.clear();
    }
    valos=valos+"Buzz";
}
//afficher le message (un nombre/Fizz/Buzz/FizzBuzz)
cout <<"Ordinateur > "<<valos<<endl;
valj=valo+1;
}

// 2° cas --- le joueur commence le jeu
//saisir l'initialisation
else
{
    valj=nbmin;
}

//la boucle pour jouer le jeu
do
{
    //conversion de int vers string
    sprintf( tabj,"%d",valj );
    valjse=tabj;
    //vérifier si 'valj' est le multiple de 3 ou 5
    mult3 = (valj%3==0) ;
    mult5 = (valj%5==0) ;
    //si 'valj' est le multiple de 3 ou 5 , on change 'valjse'(le message affiché) à
Fizz/Buzz/FizzBuzz
    //utilisant la concaténation pour générer le mot FizzBuzz
    if ( mult3 )
    {
        valjse="Fizz";
    }
    if ( mult5 )
    {
        if ( mult3 )
        {
            valjse="Fizz";
        }
        else
        {
            valjse.clear();
        }
        valjse=valjse+"Buzz";
    }
    //demander et saisir la réponse du joueur
    cout <<nom<< "("<<vie<<" vies)"<<" > ";
    cin >>valjs;

    //vérifier si la réponse est correcte
    while ( valjs!=valjse )
    {
        //si la réponse est incorrecte , le joueur perd un point de vie

```

```

        vie=vie-1;
        cout <<"Vous vous etes trompe,vous perdez un point de vie"<<endl;
        cout <<"La bonne valeur est "<<valjse<<" ."<<endl;
        //vérifier si le joueur a encore de la vie pour continuer le jeu
        if ( vie==0 )
        {
                cout <<"Vous avez perdu !"<<endl;
                break;
        }
        //redemander la réponse du joueur
        cout <<nom<<"("<<vie<<" vies)"<<" > ";
        cin >>valjs;
}

```

```

//vérifier si le joueur a encore de la vie pour continuer le jeu
if ( vie==0 )
{
        break;
}

```

```

//donner la nouvelle valuation à 'valo'
valo=valj+1;

```

```

//comparer 'valo' et 'nbmax' pour confirmer si ce 'valo' peut être afficher
if ( valo>nbmax )
{
        break;
}

```

```

//conversion de int vers string
sprintf( tabo,"%d",valo );
valos=tabo;
//vérifier si 'valo' est le multiple de 3 ou 5
mult3 = (valo%3==0) ;
mult5 = (valo%5==0) ;
//si 'valo' est le multiple de 3 ou 5 , on change 'valos'(le message affiché) à

```

Fizz/Buzz/FizzBuzz

```

//utilisant la concaténation pour générer le mot FizzBuzz
if ( mult3 )
{
        valos="Fizz";
}
if ( mult5 )
{
        if ( mult3 )
        {
                valos="Fizz";
        }
        else
        {
                valos.clear();
        }
}

```



```

        valos=valos+"Buzz";
    }
    //afficher le message (un nombre/Fizz/Buzz/FizzBuzz)
    cout <<"Ordinateur > "<<valos<<endl;
    //donner la nouvelle valuation à 'vqlj'
    valj=valo+1;
}
while( valj<=nbmax );

//quand le joueur réussi , on affiche le message "Victoire"
if ( valj>=nbmax )
{
    cout <<"Victoire."<<endl;
}
//proposer au joueur de recommencer ou de quitter
cout <<"Vous voulez recommencer une partie ou quitter le jeu?"<<endl;
cout <<"Tapez '0' pour recommencer et tapez '1' pour quitter."<<endl;
cin >>recommence;
}
while ( recommence==0 );

return 0 ;
}

```