

Міністерство освіти і науки України
Національний університет «Львівська Політехніка»
Кафедра ЕОМ



ЗВІТ

з лабораторної роботи № 4

З дисципліни: «Об'єктно-орієнтовне програмування»

КЛАСИ ТА ОБ'ЄКТИ

Оцінка:

Виконав:
ст. групи КІ-103
Кобзев Р.Д

Прийняв
аспірант Науличний В.В

Львів-2024

Мета роботи: познайомитися із класами та об'єктами.

Завдання: Клас CDice (Гральні кості). Клас зберігає 3 гральні кості та загальну кількість очків після останнього кидання. Повинні бути передбачені методи, що реалізують ініціалізацію та кидання гральних костей, вивід на екран поточного стану об'єкта.

Пояснення:

dice.h:

Оголошує клас Dice з приватними змінними

Містить конструктори, гетери та сетери для доступу до приватних змінних.

Оголошує методи для “кидка кубика”

dice.cpp:

Реалізує методи класу Dice, які були оголошені в файлі заголовка dice.h.

Визначає конструктори за замовчуванням та з параметрами, методи для отримання значень та методи для управління кубиком.

Dice_main.cpp:

Використовується для демонстрації функціоналу класу Dice.

Створюється об'єкт dice класу Dice з вектором каналів, де викликаються різні методи для керування кубиками.

Виводиться стан гральних кубиків після кожного кидка.

Отже, цей код дозволяє створити об'єкт з трьома кубиками, а також бачити їхню суму чи результат кидка.

Код:

Файл dice.h:

```
#include <ctime>
```

```
#include <cstdlib>
```

```
#include <iostream>
```

```
class Dice
```

```
{
```

```
    private:
```

```

    int all_dices[3];
    int totalPoints;
public:
    Dice(){
        srand(time(0));
        for(int i=0;i<3;i++)
        {
            all_dices[i]=0;
        }
        totalPoints=0;
    }
    virtual ~Dice();
    void rollDice();
    void displayState();
};

```

Файл dice.cpp:

```

#include "dice.h"
using namespace std;
Dice::~Dice(){
}
void Dice::rollDice(){
    totalPoints = 0;
    for(int i=0;i<3;i++){
        all_dices[i] = rand()% 6 + 1;
        totalPoints +=all_dices[i];
    }
}

```

```

void Dice::displayState(){
    cout<<"Dice values: ";
    for(int i=0;i<3;i++){
        cout<<all_dices[i]<<' ';
    }
    cout<<"Total points: "<<totalPoints<<endl;
}

```

Файл Dice_main.cpp:

```
#include <iostream>
```

```
#include "dice.h"
```

```
using namespace std;
```

```
int main()
```

```
{
```

```
    int choice;
```

```
    Dice dice;
```

```
    while(true){
```

```
        cout<<"Choose ur option: "<<endl;
```

```
        cout<<"1.Roll dices"<<endl;
```

```
        cout<<"2.Show results"<<endl;
```

```
        cin>>choice;
```

```
        if(choice==1){
```

```
            dice.rollDice();
```

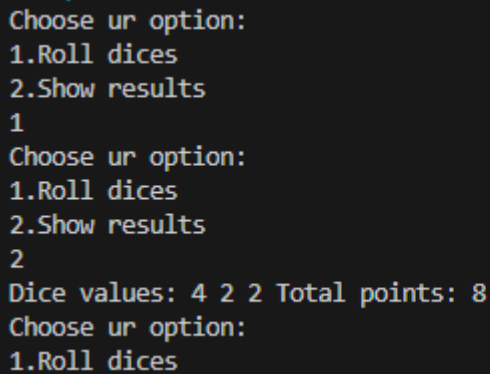
```
        }else if(choice==2){
```

```
            dice.displayState();
```

```
        }else{
```

```
        cout<<"Invalid input!";  
    }  
}  
}
```

Результати виконання програми:



```
Choose ur option:  
1.Roll dices  
2.Show results  
1  
Choose ur option:  
1.Roll dices  
2.Show results  
2  
Dice values: 4 2 2 Total points: 8  
Choose ur option:  
1.Roll dices
```

Висновок

В рамках виконання даного завдання було розроблено клас CDice, що моделює процес кидання трьох гральних костей та підрахунок загальної кількості очок. Основна функціональність класу включає ініціалізацію костей, здійснення кидків та виведення поточного стану об'єкта на екран.

Розроблений клас CDice успішно виконує свої функції по моделюванню процесу кидання гральних костей та обчисленню загальної кількості очок. Реалізовані методи дозволяють легко ініціалізувати, кидати та виводити стан костей, що робить цей клас корисним для програм, пов'язаних з іграми та симуляціями.