

Міністерство освіти і науки України
Національний університет «Львівська політехніка»

Кафедра ЕОМ



Звіт
до лабораторної роботи № 5
з дисципліни: «Програмування, частина 2 (ООП)»
на тему: «Перевантаження операторів »
Варіант № 8

Підготував:
студент групи КІ-103
Кобзєв Роман
Перевірив:
асист. каф. ЕОМ
Науличний В.В.

Львів 2024

Мета роботи: познайомитися із переваженням операторів.

Завдання

Розширити функціональність розроблених у 4 лабораторній роботі класів за допомогою операторів, що задані варіантом та оператора присвоєння. Конкретні функції операторів реалізувати на власний розсуд (крім оператора присвоєння). Організувати виведення та введення даних за допомогою класів-потоків `cin`, `cout` та переважених операторів вводу/виводу. Написати програму, яка демонструє роботу з об'єктами цього класу.

№	Завдання
8	++, +=, >, <

Текст програми

Програма реалізована як проєкт з 3-ох файлів: `dice.h` (прототип класу та його методів), `dice.cpp` (реалізація методів класу), `Dice_main.cpp` (головна функція).

`dice.h` (прототип класу та його методів):

```
#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();
    Dice operator++();
    Dice& operator+=(int q);
    bool operator>(const Dice& other);
    bool operator<(const Dice& other);
    Dice operator=(const Dice& other);
    friend ostream& operator<<(ostream& os, const Dice& obj);
};
```

`dice.cpp` (реалізація методів класу):

```
#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;
}
Dice Dice::operator++()
{
    for(int i=0 ; i<3 ; i++){
        all_dices[i]++;
    }
    return *this;
}
```

`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!";
        }
    }
}
```

}

```
Choose your option:
1. Roll dices
2. Show results
3. Exit
1
Choose your option:
1. Roll dices
2. Show results
3. Exit
2
Dice values: 3 5 2 Total points: 10
Choose your option:
1. Roll dices
2. Show results
3. Exit
```

Висновок

У ході виконання даного завдання було розширено функціональність розроблених у попередній лабораторній роботі класів за допомогою перевантаження операторів, що дозволило забезпечити більш зручну та інтуїтивно зрозумілу взаємодію з об'єктами цих класів. Оператори було реалізовано на власний розсуд, а саме: оператор присвоювання, оператори збільшення (++), оператори порівняння (>, <), а також оператори додавання та додавання з присвоєнням (+, +=).