

Лабораторна №9

Тема: Вивчення механізмів оброблення виняткових ситуацій.

Мета: навчитись обробляти ситуації появлення виняткових ситуацій, вивчити механізми їх оброблення.

Завдання 1. Додайте клас винятків до програми ARROVER таким чином, щоб індекси, що виходять за межі масиву, викликали генерацію винятку.

Блокпастка catch може виводити користувачеві повідомлення про помилку.

```
/////program ARROVER///// #include using namespace std; #include //for exit()
const int LIMIT = 100; //array size //////////////////////////////////////
class safearray { private: int arr[LIMIT]; public: int& operator [] (int n) { if( n< 0 ||
n>=LIMIT ) { cout << "\nIndex out of bounds"; exit(1); } return arr[n]; } };
//////////////////////////////////// int main() { safearray sa1; for(int j=0;
j<< "Element " << j << " is " << temp << endl; } return 0; }
```

Завдання 2. Модифікуйте програму з попереднього завдання таким чином, щоб в повідомлення про помилку входила інформація про значення індексу, який призвів до збою. 4. Модифікуйте текст програми, розробленої згідно індивідуального завдання лабораторної роботи №3, додавши до нього (в місця можливого виникнення помилок) процедури оброблення виняткових ситуацій, які будуть генерувати об'єкти класу, який міститиме такі атрибути: місце виникнення помилки; значення, яке призвело до помилки; параметризований конструктор; перевизначену операції виводу, яка виводитиме значення на екран (або зберігатиме виняткову ситуацію у файл).

Код програми:

```
#include <QCoreApplication>

// #include <iostream>
// using namespace std;
// class asd {
// public:
//     int asd1;
//     asd() {
//         asd1 = 1;
//     }
//     asd(int a2) {
```

```

//      try
//      {
//          asd1 = a2;
//      }
//      catch (const std::exception&)
//      {
//          cout << "Введене значення не є числом";
//      }
//
//  }
//  void operator + (int a1) {
//      try
//      {
//          this->asd1 = a1 + this->asd1;
//      }
//      catch (const std::exception&)
//      {
//          cout << "Якийсь із доданків не є числом";
//      }
//  }
//  void operator = (asd a1) {
//      try
//      {
//          this->asd1 = a1.asd1;
//      }
//      catch (const std::exception&)
//      {
//          cout << "Якесь значення не є числом";
//      }
//  }
//  void Output() {
//      std::cout << asd1;
//  }
//};
//
//int main()
//{
//    try
//    {
//        asd* mas1 = new asd(3);
//        asd* rez = new asd(0);
//        *mas1 + 2;
//        rez = mas1;
//        rez->Output();
//    }
//    catch (const std::exception&)
//    {
//        cout << "Проблеми при використанні конструктора";
//    }
//
//
//}
/////program ARROVER/////
#include <iostream>
using namespace std;

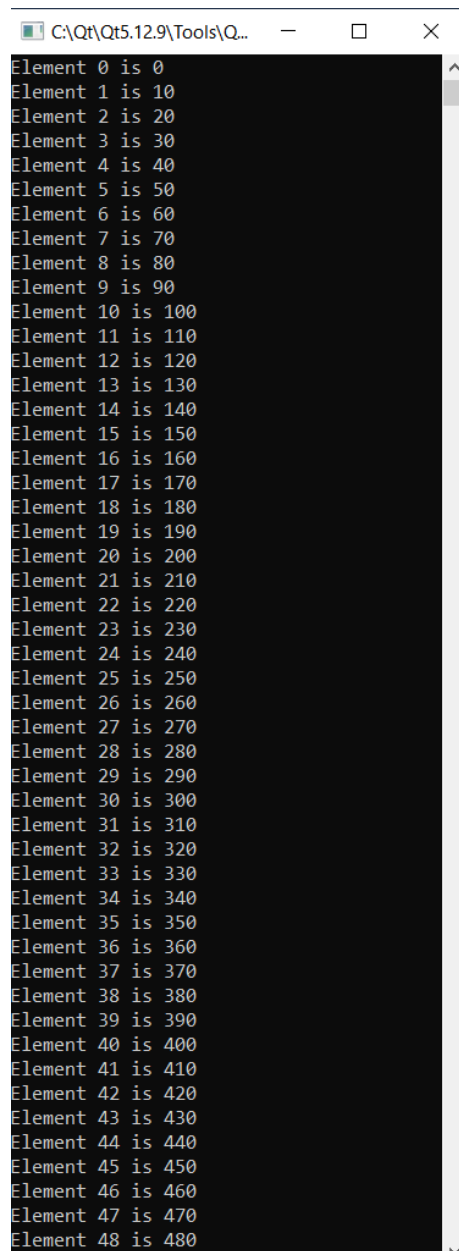
```

```

#include <process.h> //for exit()
const int LIMIT = 100; //array size
////////////////////////////////////
class safearray
{
private:
    int arr[LIMIT];
public:
    int& operator [] (int n)
    {
        if (n < 0 || n >= LIMIT)
        {
            cout << "\nIndex out of bounds"; exit(1);
        }
        return arr[n];
    }
};
////////////////////////////////////
int main()
{
    try
    {
        safearray sal;
        for (int j = 0; j < LIMIT; j++)
            sal[j] = j * 10; /*left* side of equal sign
        for (int j = 0; j < LIMIT; j++) //display elements
        {
            int temp = sal[j]; /*right* side of equal sign
            cout << "Element " << j << " is " << temp << endl;
        }
        return 0;
    }
    catch (int j)//const std::exception&
    {
        cout << "\nIndex out of bounds"; exit(1);
    }
}

```

Результат:



```
C:\Qt\Qt5.12.9\Tools\Q...
Element 0 is 0
Element 1 is 10
Element 2 is 20
Element 3 is 30
Element 4 is 40
Element 5 is 50
Element 6 is 60
Element 7 is 70
Element 8 is 80
Element 9 is 90
Element 10 is 100
Element 11 is 110
Element 12 is 120
Element 13 is 130
Element 14 is 140
Element 15 is 150
Element 16 is 160
Element 17 is 170
Element 18 is 180
Element 19 is 190
Element 20 is 200
Element 21 is 210
Element 22 is 220
Element 23 is 230
Element 24 is 240
Element 25 is 250
Element 26 is 260
Element 27 is 270
Element 28 is 280
Element 29 is 290
Element 30 is 300
Element 31 is 310
Element 32 is 320
Element 33 is 330
Element 34 is 340
Element 35 is 350
Element 36 is 360
Element 37 is 370
Element 38 is 380
Element 39 is 390
Element 40 is 400
Element 41 is 410
Element 42 is 420
Element 43 is 430
Element 44 is 440
Element 45 is 450
Element 46 is 460
Element 47 is 470
Element 48 is 480
```

Висновок: навчилась обробляти ситуації появи виняткових ситуацій, вивчила механізми їх оброблення.