

Лабораторна робота №7

Тема: Шаблони функцій і класів.

Мета: Навчитись створювати і використовувати шаблонні функції і класи

Завдання 1-2.

Напишіть шаблон функції, що повертає середнє арифметичне всіх елементів масиву. Аргументами функції повинні бути ім'я і розмір масиву (типу `int`). У функції `main()` перевірте роботу шаблонної функції з масивами типу `int`, `long`, `double` і `char`.

Створіть функцію `amax()`, що повертає значення максимального елемента масиву. Аргументами функції повинні бути адреса і розмір масиву. Зробіть з функції шаблон, щоб вона могла працювати з масивом будь-якого числового типу. Напишіть функцію `main()`, в якій перевірите роботу функції з різними типами масивів.

Код програми

```
#include <iostream>

using namespace std;

//Завдання №1
template<typename T>
T func(T *arr,int n)
{
    arr=new T[n];

    for (int i = 0; i < n; ++i)
    {
        cout<<"Enter arr"<<i<<"]=";
        cin>>arr[i];
    }

    T t=0;
    for (int i = 0; i < n; ++i)
    {
        t+=arr[i];
    }
    return t/n;
}

//Завдання №2
template<typename T1>
T1 aMax(T1 *arr,int n)
{
    arr=new T1[n];

    for (int i = 0; i < n; ++i)
    {
```

```

        cout<<"Enter arr"<<i<<"]=";
        cin>>arr[i];
    }

    T1 max=0;

    for(int i=0;i<n;i++)
    {
        if(arr[i]>max) {max=arr[i];}
    }

    return max;
}

int main()
{
    int n=3;

    int *a = nullptr;
    int h;
    h=func<int>(a,n);
    cout<<h<<endl;
    long *b = nullptr;
    long j;
    j=func<long>(b,n);
    cout<<j<<endl;
    double *c = nullptr;
    double g;
    g=func<double>(c,n);
    cout<<g<<endl;
    char *d = nullptr;
    char f;
    f=func<char>(d,n);
    cout<<f<<endl;

    //////////////////////////////////////
    //Завдання №2
    int z;
    h=aMax<int>(&z,n);
    cout<<h<<endl;
    long v;
    j=aMax<long>(&v,n);
    cout<<j<<endl;
    double x;
    g=aMax<double>(&x,n);
    cout<<g<<endl;
    char m;
    f=aMax<char>(&m,n);
    cout<<f<<endl;

```

Результат

Завдання 1

```
C:\Qt\Qt5.12.9\Tools\QtCreator\bin\qtcreator_process_stub.exe
Enter arr0]=1
Enter arr1]=2
Enter arr2]=3
2
Enter arr0]=25
Enter arr1]=54
Enter arr2]=458
179
Enter arr0]=1.5
Enter arr1]=4.6
Enter arr2]=7.5
4.53333
Enter arr0]=q
Enter arr1]=w
Enter arr2]=r
[]
```

Завдання 2

```
C:\Qt\Qt5.12.9\Tools\QtCreator\bin\qtcreator_process_stub.exe
Enter arr0]=2
Enter arr1]=4
Enter arr2]=5
5
Enter arr0]=445
Enter arr1]=558
Enter arr2]=55
558
Enter arr0]=1.5
Enter arr1]=0.6
Enter arr2]=88.2
88.2
Enter arr0]=t
Enter arr1]=r
Enter arr2]=w
w
```

Завдання 3.

Створіть шаблонний клас, який міститиме як атрибут - масив будь-якого числового типу. Розмір масиву необхідно визначати параметром конструктора класу. З допомогою методів класу потрібно:

- заповнювати масив;
- виводити значення масиву на екран;
- визначати і вивести середнє арифметичне всіх елементів масиву;
- визначати і вивести максимальний елемент масиву.

Напишіть функцію `main()`, в якій перевірите роботу класу з різними вбудованими типами даних.

Код програми

```
#include <iostream>
#include <typeinfo>
using namespace std;

template<typename T>
class Tckass
{
private:
    int size = 0;
    T* arr;
public:
    Tckass(int size)
    {
        this->size = size;
        arr = new T[size];
    }

public:
    void fillArray()
    {
        cout << "Enter Size array"<< size << endl;
        for (int i = 0; i<size; i++ )
        {
            cout<<"Enter Element["<<i<<"]=";
            cin >> arr[i];
        }
    }

    void printArray()
    {
        cout << endl << "Element Array " << endl;
        for (int i =0; i<size; i++)
        {
            cout << endl << "Element Array [" << i<<"]="<< arr[i] << endl;
        }
        cout << endl;
    }

    T ArithmeticMean()
    {
        T suma = 0;
        for (int i =0; i<size; i++)
        {
            suma += arr[i];
        }
        cout << "Arithmetic mean = ";
        return suma / size;
    }

    T MaxElement()
    {
        T maxElement = arr[0];
        for (int i =0; i<size; i++)
        {
            if (arr[i] > maxElement)
            {
                maxElement = arr[i];
            }
        }
        cout << "Max Element = ";
    }
};
```

```

        return maxElement;
    }
};

int main()
{
    cout << "Enter size=";
    int size = 0;
    cin >> size;

    Tckass<int>Arr(size);
    Arr.fillArray();
    Arr.printArray();
    cout << endl;
    cout << Arr.ArithmeticMean() << endl;
    cout << Arr.MaxElement() << endl;

    cout << endl << "-----" << endl << endl;

    Tckass<double>Arr2(size);
    Arr2.fillArray();
    Arr2.printArray();
    cout << endl;
    cout << Arr2.ArithmeticMean() << endl;
    cout << Arr2.MaxElement() << endl;
    cout << "-----" << endl;
    return 0;
}

```

Результат



```

C:\Qt\Qt5.12.9\Tools\QtCreator\bin\qtcreator_process_stub.exe
Element Array [0]=1
Element Array [1]=2
Element Array [2]=3
Arithmetic mean = 2
Max Element = 3
-----
Enter Size array3
Enter Element[0]=1.
Enter Element[1]=1.5
Enter Element[2]=2.2
Element Array
Element Array [0]=1
Element Array [1]=1.5
Element Array [2]=2.2
Arithmetic mean = 1.56667
Max Element = 2.2
-----

```

Завдання 4.

Створити шаблонний клас – одновимірний масив. Тип елементів масиву визначається параметром шаблону. Передбачити функції для виконання таких перевантажених операцій: [] (з контролем діапазону індексу), =

Код програми

```

// c++11/14/17
#include <iostream>
#include <string>
#include <conio.h>

using namespace std;

template<class T>
class MyArr
{
    T*arr;
    int len;
    int position;
public:
    MyArr()
    {
        len = 1;
        arr = new T[len]{};
    }

    MyArr(int size)
    {
        len = size;
        arr = new T[len]{};
    }

    ~MyArr()
    {
        if (arr != nullptr)
        {
            delete[] arr;
        }
    }

    void set()
    {
        for (int i = 0; i < len; i++)
        {
            cout<<"Enter element " <<i+1<<" = ";
            cin>>arr[i];
        }
    }

    void print() const
    {
        for (int i = 0; i < len; i++)
        {
            cout << arr[i] << "\t";
        }
        cout << endl;
    }

    T& operator[](int i)
    {
        if(i<0||i>len){
            cout<<"Invalide data";
        }
        else
        {
            position=i;
            return arr[i];
        }
    }

```

```

    }

MyArr& operator=( T &arrInt2 ) //перегрузка оператора присваивания
{
    this->arr[position]=arrInt2;
    return *this;
}
friend ostream& operator<< (std::ostream &out, const MyArr<T> &point)
{
    out << point.arr;

    return out;
}
friend istream& operator>> (istream &in, MyArr<T> &point)
{
    in >> point.set();
    return in;
}

};

int main()
{
    char ch;
    int n,size=0;
    MyArr<int> a(10);
    a.set();
    a.print();
point1:

    cout<<"Enter element array Y/N\n";
    ch=getch();

    if(ch=='N' || ch=='n') exit(0);
    else if(ch=='Y' || ch=='y')
    {

point2:
        cout<<"Enter position=";
        cin>>n;
        if(n<1){cout<<"Incorrect button\n";
            goto point2;}
        a[n-1];
        cout<<"Enter new values=";
        cin>>size;
        a[n-1]=size;

        cout<<"\n";
        a.print();

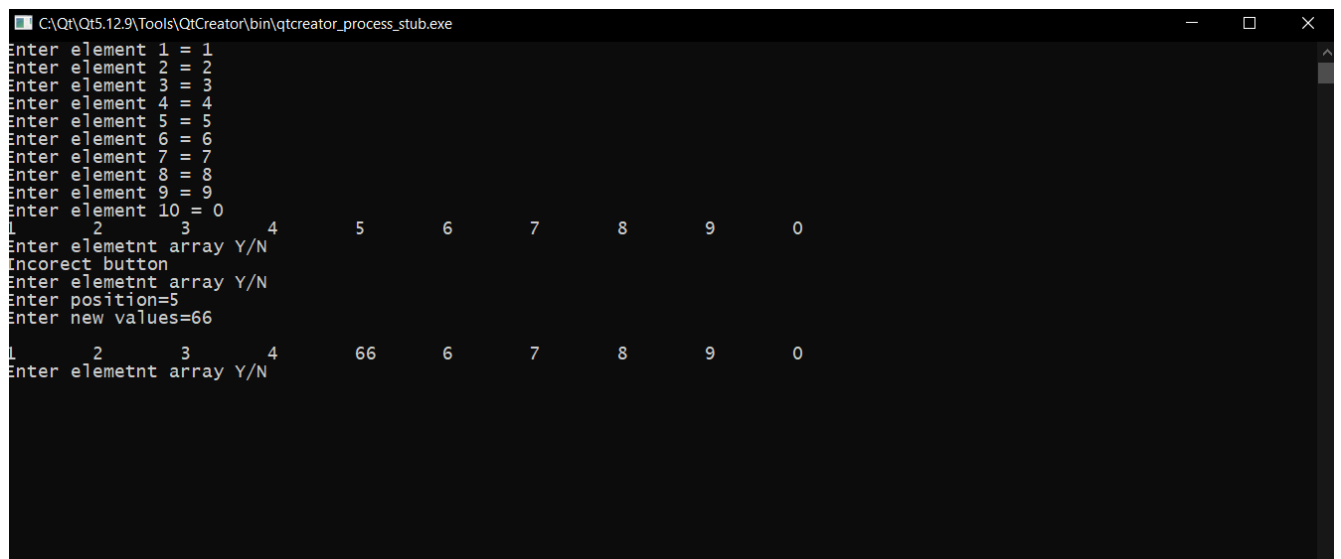
        goto point1;

    }
    else {cout<<"Incorrect button\n";
        goto point1;}

}

```

Результат



```
C:\Qt\Qt5.12.9\Tools\QtCreator\bin\qtcreator_process_stub.exe
Enter element 1 = 1
Enter element 2 = 2
Enter element 3 = 3
Enter element 4 = 4
Enter element 5 = 5
Enter element 6 = 6
Enter element 7 = 7
Enter element 8 = 8
Enter element 9 = 9
Enter element 10 = 0
Enter element array Y/N
Incorrect button
Enter element array Y/N
Enter position=5
Enter new values=66
Enter element array Y/N
```

1	2	3	4	5	6	7	8	9	0
---	---	---	---	---	---	---	---	---	---

1	2	3	4	66	6	7	8	9	0
---	---	---	---	----	---	---	---	---	---

Висновок:

Навчився створювати і використовувати шаблонні функції і класи