***ЛАБОРАТОРНА РОБОТА №7***

**Тема:** Шаблони функцій і класів  
**Мета:** Навчитись створювати і використовувати шаблонні функції і класи

***ПОРЯДОК ВИКОНАННЯ РОБОТИ***

**Завдання 1.** 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.

**Завдання 2.** Створіть функцію аmaх(), 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.

**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.

**Завдання 4.** Створити шаблонний клас – одновимірний масив. Тип елементів масиву визначається параметром шаблона. Передбачити функції для виконання таких перевантажених операцій: [] (з контролем діапазону індексу), = , <>.

Код програми:

#include <QCoreApplication>

#include <iostream>

using namespace std;

template<class T>

T middleSum(T \*array, int count)

{

T sum = 0;

for(int i = 0; i < count; i++)

sum += array[i];

return sum/count;

}

template<class T>

T amax(T \*array, int count)

{

if(count < 1) return 0;

T max = array[0];

for(int i = 0; i < count; i++)

if(array[i] > max) max = array[i];

return max;

}

template<class T>

class arrayWork

{

T \*mas;

int size;

public:

arrayWork(int size){this->size = size; mas = new T[size];}

void writeElement(int index, T val){if(index < size) mas[index] = val;}

void Show(){for(int i= 0; i < size; i++) cout << mas[i] << " "; cout << endl;}

T middleSum()

{

T sum = 0;

for(int i = 0; i < size; i++)

sum += mas[i];

return sum/size;

}

T amax()

{

if(size < 1) return 0;

T max = mas[0];

for(int i = 0; i < size; i++)

if(mas[i] > max) max = mas[i];

return max;

}

};

template<class T>

class Array1D

{

T \*array;

int size;

public:

Array1D(int size){

this->size = size;

if(this->size < 1) this->size = 1;

array = new T[this->size]();

}

T& operator[](int indx){return array[indx%size];}

Array1D& operator=(Array1D& array){

int sizeN = size;

if(sizeN > array.size) sizeN = array.size;

for(int i = 0; i < sizeN; i++)

array[i] = array[i];

return this;

}

T GetSize(){return size;}

template<class F> friend ostream& operator<<(ostream& outStr, Array1D<F>& array);

template<class F> friend istream& operator>>(istream& outStr, Array1D<F>& array);

};

template<class T>

ostream& operator<<(ostream& outStr, Array1D<T>& array)

{

for(int i = 0; i < array.size; i++)

outStr << array[i] << " ";

outStr << endl;

return outStr;

}

template<class T>

istream& operator>>(istream& inStr, Array1D<T>& array)

{

for(int i = 0; i < array.size; i++)

inStr >> array[i];

return inStr;

}

int main(int argc, char \*argv[])

{

QCoreApplication a(argc, argv);

//-----------1----------------

int mas1[]{1,4,66,3};

float mas2[]{1.4,4.6,6.6,35.3};

cout << "Mas1: " << middleSum(mas1, 4) << " Mas2: " << middleSum(mas2, 4) << endl << endl;

//-----------2----------------

cout << "Mas1: " << amax(mas1, 4) << " Mas2: " << amax(mas2, 4) << endl << endl;

//-----------3-----------------

arrayWork<int> array1(4);

for(int i = 0; i < 4; i++)

array1.writeElement(i, mas1[i]);

array1.Show();

cout << array1.middleSum() << endl;

arrayWork<float> array2(4);

for(int i = 0; i < 4; i++)

array2.writeElement(i, mas2[i]);

array2.Show();

cout << array2.amax() << endl << endl;

//------------4-----------------

Array1D<int> arr(4);

cout << "Write mas(" << arr.GetSize() << "): " << endl;

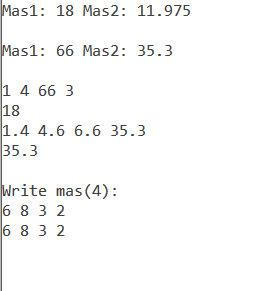
cin >> arr;

cout << arr;

return a.exec();

}

Вивід програми:



**Висновок:** Навчився створювати і використовувати шаблонні функції і класи.