

Лабораторна робота №7

Тема: Шаблони функцій і класів.

Мета: Навчитись створювати і використовувати шаблонні функції і класи

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.

Код програми:

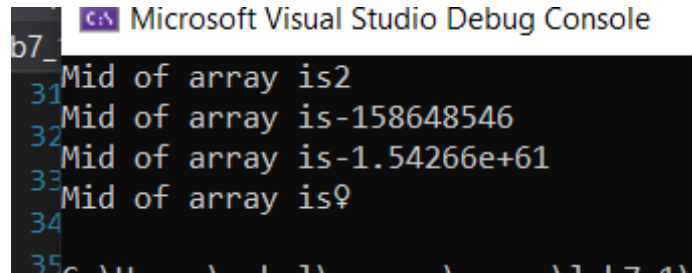
```
#include <iostream>

template <typename T>
T getMiddle (T *a, int size)
{
    T sum = 0;
    for( int i = 0; i < size; i++ )
        sum += a[i];
    return sum / size;
}

int main ()
{
    int array[ ] = { 1,2,3,4,5, 1};
    std::cout << "Mid of array is" << getMiddle (array, 6) << std::endl;
    long arra[ ] = { 1111111111111111,222222222222, 3 };
    std::cout << "Mid of array is" << getMiddle (arra, 3) << std::endl;
    double array8[ ] = { 1.2,3.0,4.3,1.0, 1.3 };
    std::cout << "Mid of array is" << getMiddle (array8, 6) << std::endl;
    char arra1[ ] = { 'a','b', 'c' };
    std::cout << "Mid of array is" << getMiddle (arra1, 3) << std::endl;
```

}

Результат виконання програми:



```
Microsoft Visual Studio Debug Console
b7_
31 Mid of array is2
32 Mid of array is-158648546
33 Mid of array is-1.54266e+61
34 Mid of array is0
35
```

Завдання 2. 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>
```

```
template <typename T>
```

```
T getMax (T *a, int size)
```

```
{
```

```
    T max = a[0];
```

```
    for( int i = 1; i < size; i++ ) {
```

```
        if( a[i] > max )
```

```
            max = a[i];
```

```
    }
```

```
    return max;
```

```
}
```

```
int main ()
```

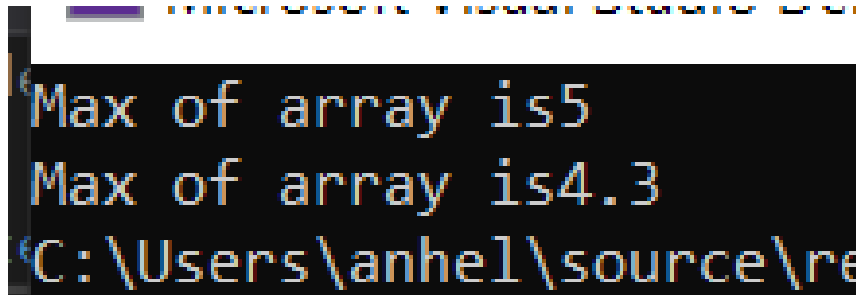
```
{
```

```
    int array[ ] = { 1,2,3,4,5, 1};
```

```
    std::cout << "Max of array is" << getMax (array, 6) << std::endl;
```

```
double array8[ ] = { 1.2,3.0,4.3,1.0, 1.3 };
std::cout << "Max of array is" << getMax (array8, 6);
}
```

Результат виконання програми:



```
Max of array is5
Max of array is4.3
C:\Users\anhel\source\re
```

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>
```

```
template<typename T>
```

```
class Num
```

```
{
```

```
private:
```

```
int m_lenght;
```

```
T *m_array;
```

```
public:
```

```
Num (int len = 0, T *a = nullptr) : m_lenght (len) {m_array = a; }
```

```
T getMax ();  
T getMid ();  
void getArray ();  
void setArray(int );  
~Num ();  
};
```

```
template<typename T>  
T Num<T>::getMax ()  
{  
    T max = m_array[0];  
    for( int i = 1; i < m_lenght; i++ ) {  
  
        if( m_array[i] > max )  
  
            max = m_array[i];  
  
    }  
  
    return max;  
  
}
```

```
template<typename T>  
T Num<T>::getMid ()  
{  
    T sum = 0;
```

```
for( int i = 0; i < m_lenght; i++ )  
    sum += m_array[i];
```

```
if( m_lenght != 0 )  
    sum = sum / m_lenght;  
else  
    std::cout << "Uncorrect!";  
return sum;
```

```
}
```

```
template<typename T>  
void Num<T>::getArray ()  
{  
    for( int i = 0; i < m_lenght; i++ )  
        std::cout << m_array[i];  
  
}
```

```
template<typename T>  
void Num<T>::setArray (int len)  
{  
    T n;  
    m_lenght = len;  
    std::cout << "Enter array:";  
    for( int i = 0; i < m_lenght; i++ ) {  
        std::cin >>n;  
        m_array[i]=n;
```

```
}  
}
```

```
template<typename T>  
Num<T>::~~Num ()  
{  
  
}
```

```
int main ()  
{
```

```
double arr[5] = { 1.0, 1.3, 1.99, 2.00, 0 };
```

```
Num<double> b(5, arr);  
std::cout << "B is ";  
b.isArray ();  
std::cout << "\nB max is" << b.getMax();  
std::cout << "\nB mid is " << b.getMid ();  
b.setArray (5);  
std::cout << "\nB is ";  
b.isArray ();  
  
}
```

Результат:

```
Microsoft Visual Studio Debug Console
Num< B is 11.31.9920
B max is 2
n; B mid is 1.258Enter array:1
_ler2
td::3
or( 4
s5
n
B is 12345
C:\Users\anhel\source\repos\lab7_1\Debug\lab7_1.exe
To automatically close the console when debugging stops.
ate<le when debugging stops.
>::Press any key to close this window . . .
```

Завдання 4:

Створити шаблонний клас – числовий масив. Тип елементів масиву визначається параметром шаблону (одновимірний або двовимірний). Передбачити функції для виконання таких операцій: введення елементів у масив, обчислення норми (для одновимірного масиву – корінь квадратний із суми квадратів елементів, для двовимірного – максимальне значення за модулем), виведення елементів масиву на екран.

Код програми:

```
#include <iostream>
```

```
template <typename type, unsigned int m_rows, unsigned int m_column = 1 >
```

```
class Arr
```

```
{
```

```
private:
```

```
    type **m_arr;
```

```
public:
```

```
    Arr() {
```

```
        std::cout << "Enter elements of array:" << std::endl;
```

```

    m_arr = new type*[m_rows];
for( int i = 0; i < m_rows;i++ ) {
    m_arr[i] = new type[m_column];
}
for( int i = 0; i < m_rows; i++ )
{
    for( int j = 0; j < m_column; j++ ) {
        std::cin >> m_arr[i][j];
    }
}
};

void getArr ()
{
    std::cout << "Array is ";
    for( int i = 0; i < m_rows; i++ ) {
        for( int j = 0; j < m_column; j++ ) {
            std::cout << m_arr[i][j] << " ";
        }
        std::cout << std::endl;
    }
}

type calculate ()
{
    if( m_column == 1 ) {
        int sum = 0;

```



```

    for( int i = 0; i < m_rows; i++ ) {
        for( int j = 0; j < m_column; j++ ) {
            sum += m_arr[i][j];
        }
    }
    return sqrt(sum);
}
else
{
    type max = abs(m_arr[0][0]);
    for( int i = 0; i < m_rows; i++ ) {
        for( int j = 0; j < m_rows; j++ ) {
            if( max < abs (m_arr[i][j]) )
                max = abs (m_arr[i][j]);
        }
    }
    return max;
}
};

```

```

int main()
{
    Arr<int, 2, 2> a;
    a.getArr();
    std::cout << "Norm is " << a.calculate();
}

```

Висновок: Я ознайомилась з шаблонами функцій і класів.