

Домашнее задание на 30.08

27.1 Non type parameters

Реализовать шаблонный класс Array2d, входными параметрами шаблона должны быть:

T- тип данных массива и 2 нон-тайп параметра (N,M) --- число строк и столбцов массива.

```
template<class T, size_t N, size_t M>
class Array2d{
private:
    T array2d[N][M];
public:
```

Реализовать методы :

1. Переопределить конструктор по умолчанию(он должен заполнять массив нулями)

и конструктор копирования (необходимо избежать поверхностного копирования, о котором я рассказывал неоднократно на занятиях)

2. operator<< --вывод в консоль

3. double average() --- должен выводить среднее значение в массиве.

4. Operator+ and operator- сложение и вычитание (о соответствии размеров можно не беспокоиться, по умолчанию массивы одинаковые)

5. Также, реализуйте необходимый метод(методы) для того чтобы можно было к объекту типа array2d использовать [], т.е.

```
Array2d<int,5,6> dArray;
dArray[0][1]=14;
```

произвести тесты на типах данных int и double , отразить их в main.

27.2. Optionally

Необходимо продемонстрировать работу реализованного в 27.1 класса на типе данных, который должен был быть реализован на прошлом занятии — `ComplexNumber`. Т.е. как должна выглядеть инициализация:

```
int main()
{
    // Array2d<ComplexNumber<int>,5,6> dArray;
```

Note : в `ComplexNumber` должен быть реализован конструктор такого типа:

```
ComplexNumber(T real) :m_real(real), m_im(0)
{
    // std::cout << "\nConstructor executed\n";
}
```

Вероятно какие-то методы придется доопределять, имейте в виду.

27.3 template classes inheritance

Реализовать самостоятельно шаблонный класс `TypeSize`, который я реализовывал на занятии, отнаследовать от него класс `TypeHash`, который должен, пользуясь методом библиотеки `<typeinfo>`: `typeid(/* переменная`

`).hash_code()` выводить hash код переданного класса.

Произвести эксперименты на `ComplexNumber` и над `Array2d`.