

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ "ЛЬВІВСЬКА ПОЛІТЕХНІКА"

Інститут КНІТ  
Кафедра ПЗ

**ЗВІТ**

До лабораторної роботи № 10

На тему: *"Шаблони класів"*

З дисципліни: "Об'єктно-орієнтоване програмування"

**Лектор:**

доцент кафедри ПЗ  
Коротєєва Т.О.

**Виконав:**

студент групи ПЗ-16  
Коваленко Д.М.

**Прийняв:**

доцент кафедри ПЗ  
Яцишин С.І.

«\_\_\_\_\_» \_\_\_\_\_ 2022 р.  
 $\Sigma$  = \_\_\_\_\_

**Тема.** Шаблони класів.

**Мета.** Навчитись створювати шаблони класу та екземпляри шаблонів.

## Індивідуальне завдання

Створити шаблон класу `Array`, який містить однотипні елементи. Шаблон класу повинен давати можливість вивести всі елементи на екран, знайти суму всіх елементів, а також максимальний з елементів. Продемонструвати функціонал шаблону на створеному користувацькому типі `Vector2D` – вектор на площині. Більшим вважати той з векторів, який має більшу довжину.

## Код програми

```
#ifndef ARRAY_H
#define ARRAY_H

#include "QVector"
#include "QLabel"

template<class T>
class Array {
private:
    QVector<T> array;
public:
    Array(int size) {
        array = QVector<T>(size);
    };

    T sum() {
        T sum = 0;
        for (T i : array) sum += i;
        return sum;
    };
    T max() {
        T max = INT32_MIN;
        for (T i : array)
            if (i > max) max = i;
        return max;
    };
    void print(QLabel * label) {
        label->clear();
        for (T i : array) {
            label->setText(label->text() + QString::number(i) + " ");
        }
    };
    T get(int i) { return array[i]; };
    void set(int i, T v) { array[i] = v; };
};

#endif // ARRAY_H
```

```
#ifndef MAINWINDOW_H
#define MAINWINDOW_H

#include <QMainWindow>
```

```

QT_BEGIN_NAMESPACE
namespace Ui { class MainWindow; }
QT_END_NAMESPACE

class MainWindow : public QMainWindow
{
    Q_OBJECT

public:
    MainWindow(QWidget *parent = nullptr);
    ~MainWindow();

private slots:
    void on_set1Button_clicked();

    void on_set2Button_clicked();

    void on_compareButton_clicked();

    void on_print1Button_clicked();

    void on_print2Button_clicked();

private:
    Ui::MainWindow *ui;
};
#endif // MAINWINDOW_H

```

```

#ifndef VECTOR2D_H
#define VECTOR2D_H

#include "array.h"
#include "math.h"

class Vector2D {
private:
    Array<double> vector;
public:
    Vector2D(double x, double y):
        vector(Array<double>(2))
    {
        vector.set(0, x);
        vector.set(1, y);
    };
    double getX() { return vector.get(0); };
    double getY() { return vector.get(1); };
    double length() { return sqrt(getX()*getX() + getY()*getY()); };
    Array<double> * getVector() { return &vector; };
};

#endif // VECTOR2D_H

```

```

#include "mainwindow.h"
#include "ui_mainwindow.h"

#include "array.h"

```

```

#include "vector2d.h"

Vector2D * vec1;
Vector2D * vec2;

MainWindow::MainWindow(QWidget *parent)
: QMainWindow(parent)
, ui(new Ui::MainWindow) {
    ui->setupUi(this);
}

MainWindow::~MainWindow() {
    delete ui;
    delete vec1;
    delete vec2;
}

void MainWindow::on_set1Button_clicked() {
    double x = ui->vec1xEdit->text().toDouble();
    double y = ui->vec1yEdit->text().toDouble();
    vec1 = new Vector2D(x, y);
}

void MainWindow::on_set2Button_clicked() {
    double x = ui->vec2xEdit->text().toDouble();
    double y = ui->vec2yEdit->text().toDouble();
    vec2 = new Vector2D(x, y);
}

void MainWindow::on_compareButton_clicked() {
    if (vec1->length() > vec2->length())
        ui->compareButton->setText(QString::fromStdString(">"));
    else if (vec1->length() < vec2->length())
        ui->compareButton->setText(QString::fromStdString("<"));
    else if (vec1->length() == vec2->length())
        ui->compareButton->setText(QString::fromStdString("=="));
}

void MainWindow::on_print1Button_clicked() {
    vec1->getVector()->print(ui->array1Label);
}

void MainWindow::on_print2Button_clicked() {
    vec2->getVector()->print(ui->array2Label);
}

```

```

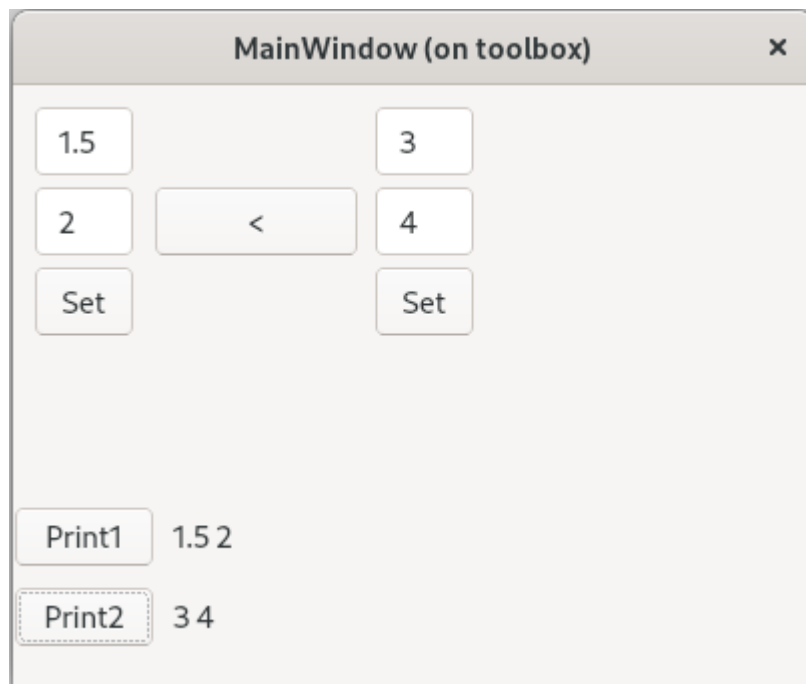
#include "mainwindow.h"

#include <QApplication>

int main(int argc, char *argv[])
{
    QApplication a(argc, argv);
    MainWindow w;
    w.show();
    return a.exec();
}

```

## Робота програми



## Висновок

Під час виконання лабораторної роботи я навчивсь створювати шаблони класу та екземпляри шаблонів