**ДЕРЖАВНИЙ УНІВЕРСИТЕТ ІНТЕЛЕКТУАЛЬНИХ ТЕХНОЛОГІЙ І ЗВ’ЯЗКУ**

**Звіт**

**з дисципліни Проектний Практикум**

**Лабораторна робота №2**

**на тему: «Об’єктно-орієнтоване програмування»»**

Виконав: студент групи ІПЗ-3.04

Бухта М.М

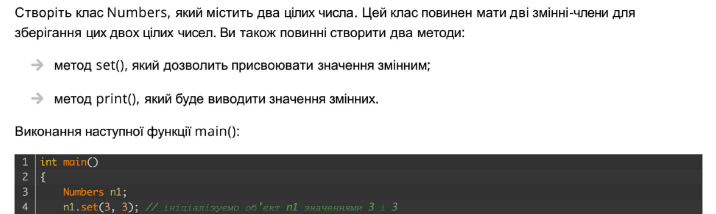
       \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

      Перевірив: Багачук Д.Г.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Одеса  2023**

**Завдання №1**

**Код програми:**

**Numbers.hpp**

*/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\**

*\* Laboratory work #2;              \**

*\* Student Bukhta Mykyta;           \**

*\* Grade: 3;                        \**

*\* Group Software Engineering 3.04; \**

*\* Task 1;                          \**

*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\**

*\*/*

#ifndef BUKHTAMYKYTA\_LAB\_2\_TASK\_1\_NUMBERS

#define BUKHTAMYKYTA\_LAB\_2\_TASK\_1\_NUMBERS

#include <inttypes.h>

namespace lab\_2 {

namespace task\_1 {

class Numbers {

public:

    Numbers(int32\_t val\_1 = 0, int32\_t val\_2 = 0);

    virtual ~Numbers(void) = default;

    bool set(int32\_t val\_1, int32\_t val\_2);

    bool print(void);

private:

    int32\_t m\_val\_1;

    int32\_t m\_val\_2;

};

} *// !task\_1;*

} *// !lab\_2;*

#endif *// !BUKHTAMYKYTA\_LAB\_2\_TASK\_1\_NUMBERS;*

**Numbers.cpp**

#include "Numbers.hpp"

#include <iostream>

namespace lab\_2 {

namespace task\_1 {

Numbers::Numbers(int32\_t val\_1, int32\_t val\_2)

    : m\_val\_1{val\_1}, m\_val\_2{val\_2}

{

}

bool Numbers::set(int32\_t val\_1, int32\_t val\_2) {

    m\_val\_1 = val\_1;

    m\_val\_2 = val\_2;

    return true;

}

bool Numbers::print(void) {

    std::cout << "Numbers(" << m\_val\_1 << ", " << m\_val\_2 << ")\n";

    return true;

}

} *// !task\_1;*

} *// !lab\_2;*

**main.cpp**

*/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\**

*\* Laboratory work #2;              \**

*\* Student Bukhta Mykyta;           \**

*\* Grade: 3;                        \**

*\* Group Software Engineering 3.04; \**

*\* Task 1;                          \**

*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\**

*\*/*

#include "Numbers.hpp"

using namespace lab\_2::task\_1;

int main(int argc, char \*\*argv) {

    Numbers n1;

    n1.set(3, 3); *// Init the object n1 with values 3 : 3;*

    Numbers n2{4, 4};

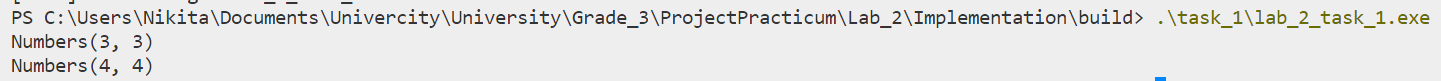
    n1.print();

    n2.print();

    return 0;

}

**Результат виконання програми:**



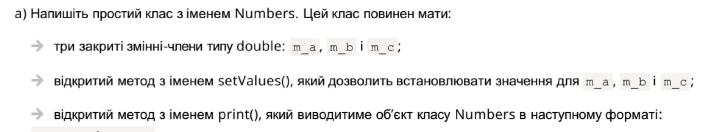
**Завдання №2**



**Відповідь:**

С++ це мова програмування, що ґрунтується на об’єктно-орієнтованих принципах програмування і проєктування (хоча можна на ній і використовувати інші парадигми програмування). Тим самим, у більшості випадків, використання класів для створення користувацького типу даних є найкращим рішенням. Структуру як користувацький тип даних можна використовувати в тих випадках, коли функціональних інтерфейсів типу даних у нас обмежено і немає необхідності інкапсулювати дані. Тобто, структура має на увазі під собою відкритий інтерфейс і доступ до метаданих без додаткового її захисту, що допомагає в деяких ситуціях спростити й оптимізувати програму. Іншими словами, це спосіб абстрагувати кілька метаданих в одну сутність: скомпонувати їх. Це можна також використовувати у випадку, коли у нас є контейнер сутностей.

**Завдання №3**

**Код програми:**

**Numbers.hpp**

*/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\**

*\* Laboratory work #2;              \**

*\* Student Bukhta Mykyta;           \**

*\* Grade: 3;                        \**

*\* Group Software Engineering 3.04; \**

*\* Task 3;                          \**

*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\**

*\*/*

#ifndef BUKHTAMYKYTA\_LAB\_2\_TASK\_3\_NUMBERS

#define BUKHTAMYKYTA\_LAB\_2\_TASK\_3\_NUMBERS

namespace lab\_2 {

namespace task\_3 {

class Numbers {

public:

    Numbers(void) = default;

    virtual ~Numbers(void) = default;

    bool setValues(double a, double b, double c) noexcept;

    bool print(void) const;

private:

    double m\_a;

    double m\_b;

    double m\_c;

};

} *// !task\_3;*

} *// !lab\_2;*

#endif *// !BUKHTAMYKYTA\_LAB\_2\_TASK\_3\_NUMBERS;*

**Numbers.cpp**

#include “Numbers.hpp”

#include <iostream>

namespace lab\_2 {

namespace task\_3 {

bool Numbers::setValues(double a, double b, double c) noexcept {

    m\_a = a;

    m\_b = b;

    m\_c = c;

    return true;

}

bool Numbers::print(void) const {

    std::cout << “<” << m\_a << “, “ << m\_b << “, “ << m\_c << “>\n”;

    return true;

}

} *// !task\_3;*

} *// !lab\_2;*

**main.cpp**

*/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\**

*\* Laboratory work #2;              \**

*\* Student Bukhta Mykyta;           \**

*\* Grade: 3;                        \**

*\* Group Software Engineering 3.04; \**

*\* Task 3;                          \**

*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\**

*\*/*

#include "Numbers.hpp"

using namespace lab\_2::task\_3;

int main(int argc, char \*\*argv) {

    Numbers point;

    point.setValues(3.0, 4.0, 5.0);

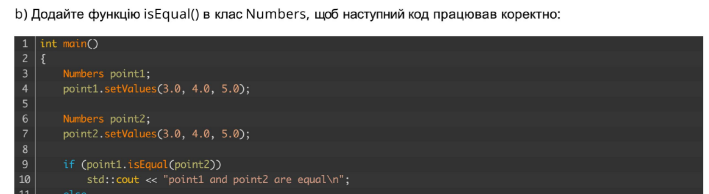
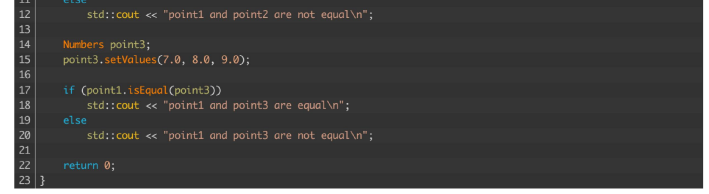
    point.print();

    return 0;

}

**Результат виконання програми:**



**Завдання №4** 

**Код програми:**

**Numbers.hpp**

*/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\**

*\* Laboratory work #2;              \**

*\* Student Bukhta Mykyta;           \**

*\* Grade: 3;                        \**

*\* Group Software Engineering 3.04; \**

*\* Task 4;                          \**

*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\**

*\*/*

#ifndef BUKHTAMYKYTA\_LAB\_2\_TASK\_4\_NUMBERS

#define BUKHTAMYKYTA\_LAB\_2\_TASK\_4\_NUMBERS

namespace lab\_2 {

namespace task\_4 {

class Numbers {

public:

    Numbers(void) = default;

    virtual ~Numbers(void) = default;

    bool isEqual(const Numbers &other) const;

    bool setValues(double a, double b, double c) noexcept;

    bool print(void) const;

private:

    double m\_a;

    double m\_b;

    double m\_c;

};

} *// !task\_4;*

} *// !lab\_2;*

#endif *// !BUKHTAMYKYTA\_LAB\_2\_TASK\_4\_NUMBERS;*

**Numbers.cpp**

#include "Numbers.hpp"

#include <iostream>

namespace lab\_2 {

namespace task\_4 {

bool Numbers::isEqual(const Numbers &other) const {

    return this->m\_a == other.m\_a && this->m\_b == other.m\_b

        && this->m\_c == other.m\_c;

}

bool Numbers::setValues(double a, double b, double c) noexcept{

    m\_a = a;

    m\_b = b;

    m\_c = c;

    return true;

}

bool Numbers::print(void) const {

    std::cout << "<" << m\_a << ", " << m\_b << ", " << m\_c << ">\n";

    return true;

}

} *// !task\_4;*

} *// !lab\_2;*

**main.cpp**

*/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\**

*\* Laboratory work #2;              \**

*\* Student Bukhta Mykyta;           \**

*\* Grade: 3;                        \**

*\* Group Software Engineering 3.04; \**

*\* Task 4;                          \**

*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\**

*\*/*

#include "Numbers.hpp"

#include <iostream>

using namespace lab\_2::task\_4;

int main(int argc, char \*\*argv) {

    Numbers point1;

    point1.setValues(3.0, 4.0, 5.0);

    Numbers point2;

    point2.setValues(3.0, 4.0, 5.0);

    if (point1.isEqual(point2))

        std::cout << "point1 and point2 are equals\n";

    else

        std::cout << "point1 and point2 are not equals\n";

    Numbers point3;

    point3.setValues(7.0, 8.0, 9.0);

    if (point1.isEqual(point3))

        std::cout << "point1 and point3 are equals\n";

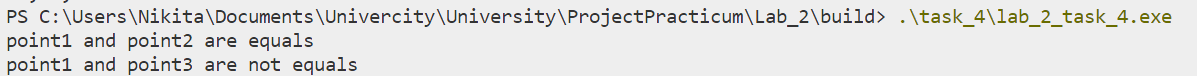
    else

        std::cout << "point1 and point3 are not equals\n";

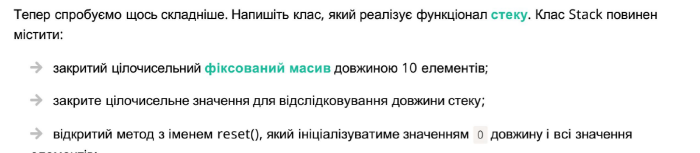
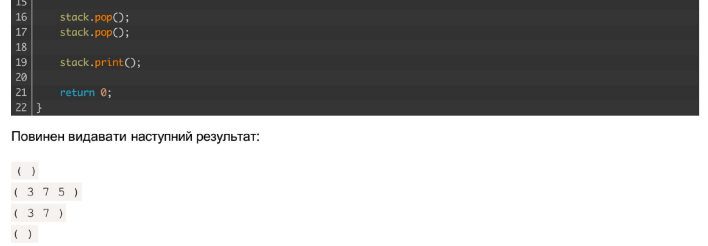
    return 0;

}

**Результат виконання програми:**



**Завдання №5**

**Код програми:**

**Stack.hpp**

*/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\**

*\* Laboratory work #2;              \**

*\* Student Bukhta Mykyta;           \**

*\* Grade: 3;                        \**

*\* Group Software Engineering 3.04; \**

*\* Task 5;                          \**

*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\**

*\*/*

#ifndef BUKHTAMYKYTA\_LAB\_2\_TASK\_5\_STACK

#define BUKHTAMYKYTA\_LAB\_2\_TASK\_5\_STACK

#include <inttypes.h>

namespace lab\_2 {

namespace task\_5 {

class Stack {

public:

    Stack(void);

    virtual ~Stack(void);

    bool reset(void);

    bool push(int32\_t val) noexcept;

    bool pop(void) noexcept;

    bool print(void) const;

private:

    const int32\_t m\_MAX\_SIZE;

    int32\_t \*m\_root;

    int32\_t \*m\_head;

};

} *// !task\_5;*

} *// !lab\_2;*

#endif *// !BUKHTAMYKYTA\_LAB\_2\_TASK\_5\_STACK;*

**Stack.cpp**

#include "Stack.hpp"

#include <iostream>

#include <assert.h>

namespace {

    const int32\_t MAX\_STACK\_SIZE = 10;

}

namespace lab\_2 {

namespace task\_5 {

Stack::Stack(void) : m\_MAX\_SIZE{MAX\_STACK\_SIZE}, m\_head{nullptr} {

    m\_root = new int32\_t[10];

}

Stack::~Stack(void) {

    delete[] m\_root;

}

bool Stack::reset(void) {

    m\_head = nullptr;

    return true;

}

bool Stack::push(int32\_t val) noexcept {

    if (!m\_head) {

        m\_head = m\_root;

    } else if ((m\_head - m\_root) < m\_MAX\_SIZE) {

        ++m\_head;

    } else {

        std::cout << "nothing" << std::endl;

        return false;

    }

    \*m\_head = val;

    return true;

}

bool Stack::pop(void) noexcept {

    assert((m\_head - m\_root) < m\_MAX\_SIZE && m\_head);

    if (--m\_head < m\_root) {

        m\_head = nullptr;

    }

    return true;

}

bool Stack::print(void) const {

    std::cout << "( ";

    for (int32\_t \*current = m\_root; current <= m\_head && m\_head; ++current) {

        std::cout << \*current << " ";

    }

    std::cout << ")\n";

    return true;

}

} *// !task\_5;*

} *// !lab\_2;*

**main.cpp**

*/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\**

*\* Laboratory work #2;              \**

*\* Student Bukhta Mykyta;           \**

*\* Grade: 3;                        \**

*\* Group Software Engineering 3.04; \**

*\* Task 5;                          \**

*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\**

*\*/*

#include "Stack.hpp"

#include <iostream>

using namespace lab\_2::task\_5;

int main(int argc, char \*\*argv) {

    Stack stack;

    stack.reset();

    stack.print();

    stack.push(3);

    stack.push(5);

    stack.push(7);

    stack.print();

    stack.pop();

    stack.print();

    stack.pop();

    stack.pop();

    stack.print();

    return 0;

}

**Результат виконання програми:**

