

Модульний контроль №1

Виконав: : Шалівський Віталій

Група: ALK-43

Київ2025

Ієрархія класів «Університет – Факультет – Кафедра»

Структура проекта

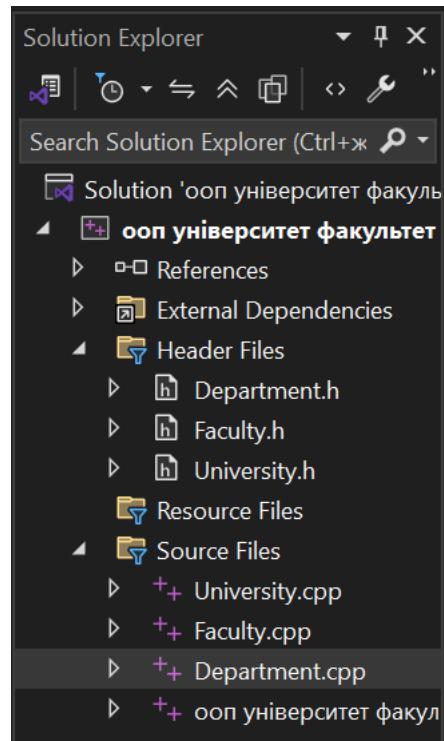


Рис. 1-Структура проекту у Visual Studio

Код програми:

University.h

```
#pragma once
#include <string>
#include <iostream>
class University {
protected:
    std::string name;
    std::string city;
    int foundedYear;
public:
    University(std::string n, std::string c, int y);
    virtual void show() const; // поліморфізм
    virtual ~University(); // віртуальний деструктор
};
```

Рис. 2-код файлу University.h

University.cpp

```
1 #include "University.h"
2
3 University::University(std::string n, std::string c, int y) {
4     : name(n), city(c), foundedYear(y)
5 }
6
7 void University::show() const {
8     std::cout << "Університет\n";
9     std::cout << "Назва: " << name << std::endl;
10    std::cout << "Місто: " << city << std::endl;
11    std::cout << "Рік заснування: " << foundedYear << std::endl;
12 }
13
14 University::~University() {}
```

Рис. 3-код файлу

University.cpp

Faculty.h

```
1 #pragma once
2 #include "University.h"
3
4 class Faculty : public University {
5 protected:
6     std::string facultyName;
7     std::string dean;
8     int students;
9
10 public:
11     Faculty(std::string n, std::string c, int y,
12             std::string f, std::string d, int s);
13
14     void show() const override;
15     ~Faculty() override;
16 }
```

Рис. 4-код файлу Faculty.h

Faculty.cpp

```
1 #include "Faculty.h"
2
3 Faculty::Faculty(std::string n, std::string c, int y,
4                   std::string f, std::string d, int s)
5     : University(n, c, y), facultyName(f), dean(d), students(s) {}
6
7
8 void Faculty::show() const {
9     std::cout << "Факультет\n";
10    std::cout << "Університет: " << name << std::endl;
11    std::cout << "Місто: " << city << std::endl;
12    std::cout << "Рік заснування: " << foundedYear << std::endl;
13    std::cout << "Факультет: " << facultyName << std::endl;
14    std::cout << "Декан: " << dean << std::endl;
15    std::cout << "К-сть студентів: " << students << std::endl;
16 }
17
18 Faculty::~Faculty() {}
```

Рис. 5-код файлу Faculty.cpp

Department.h

```
1 #pragma once
2 #include "Faculty.h"
3
4 class Department : public Faculty {
5 private:
6     std::string departmentName;
7     std::string head;
8     int teachers;
9
10 public:
11     Department(std::string n, std::string c, int y,
12                 std::string f, std::string d, int s,
13                 std::string dep, std::string h, int t);
14
15     void show() const override;
16     ~Department() override;
17 }
```

Рис. 6-код файлу Department.h

Department.cpp

```
1 #include "Department.h"
2
3 Department::Department(std::string n, std::string c, int y,
4                         std::string f, std::string d, int s,
5                         std::string dep, std::string h, int t)
6 : Faculty(n, c, y, f, d, s),
7  departmentName(dep), head(h), teachers(t) {
8 }
9
10 void Department::show() const {
11     std::cout << "Кафедра\n";
12     std::cout << "Університет: " << name << std::endl;
13     std::cout << "Місто: " << city << std::endl;
14     std::cout << "Рік заснування: " << foundedYear << std::endl;
15     std::cout << "Факультет: " << facultyName << std::endl;
16     std::cout << "Декан: " << dean << std::endl;
17     std::cout << "Кількість студентів: " << students << std::endl;
18     std::cout << "Кафедра: " << departmentName << std::endl;
19     std::cout << "Завідувач: " << head << std::endl;
20     std::cout << "Кількість викладачів: " << teachers << std::endl;
21 }
22
23 Department::~Department() {}
```

Рис. 7-код файлу Department.cpp

main.cpp

```
1 #include <iostream>
2 #include <locale>
3
4 #include "University.h"
5 #include "Faculty.h"
6 #include "Department.h"
7
8 using namespace std;
9
10 int main() {
11     setlocale(LC_ALL, "");
12
13     University* arr[3];
14
15     arr[0] = new University(
16         "Київський національний університет імені Тараса Шевченка",
17         "Київ",
18         1834
19     );
20
21     arr[1] = new Faculty(
22         "Київський національний університет імені Тараса Шевченка",
23         "Київ",
24         1834,
25         "Факультет інформаційних технологій",
26         "Петренко О.О.",
27         3200
28     );
29 }
```

```
29     arr[2] = new Department(
30         "Київський національний університет імені Тараса Шевченка",
31         "Київ",
32         1834,
33         "Факультет інформаційних технологій",
34         "Петренко О.О.",
35         3200,
36         "Кафедра програмних систем і технологій",
37         "Сидоренко М.В.",
38         35
39     );
40
41     cout << "==== Ієархія класів: Університет – Факультет – Кафедра ===\n\n";
42
43     for (int i = 0; i < 3; i++) {
44         arr[i] -> show();
45         cout << "-----\n";
46     }
47
48     for (int i = 0; i < 3; i++)
49         delete arr[i];
50
51     return 0;
52 }
```

Рис. 8-код файлу main.cpp

Результат роботи програми

```
Microsoft Visual Studio Debug X + v
==== Ієархія класів: Університет – Факультет – Кафедра ===

Університет
Назва: Київський національний університет імені Тараса Шевченка
Місто: Київ
Рік заснування: 1834
-----
Факультет
Університет: Київський національний університет імені Тараса Шевченка
Місто: Київ
Рік заснування: 1834
Факультет: Факультет інформаційних технологій
Декан: Петренко О.О.
Кількість студентів: 3200
-----
Кафедра
Університет: Київський національний університет імені Тараса Шевченка
Місто: Київ
Рік заснування: 1834
Факультет: Факультет інформаційних технологій
Декан: Петренко О.О.
Кількість студентів: 3200
Кафедра: Кафедра програмних систем і технологій
Завідувач: Сидоренко М.В.
Кількість викладачів: 35
```

Ієархія класів "Фігури обертання" – "Конус".

Структура проекта

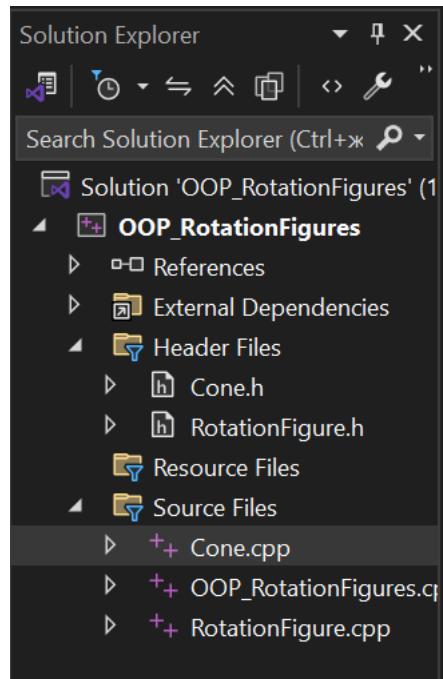


Рис. 1-Структура проекту у Visual Studio

Код програми

RotationFigure.h

```
#pragma once
#include <string>
class RotationFigure {
protected:
    std::string name;
public:
    RotationFigure(const std::string& n);
    virtual double volume() const = 0;
    virtual void show() const = 0;
    virtual ~RotationFigure() {} // ВАМПИВО: inline
};
```

Рис. 2-код файлу RotationFigure.h

RotationFigure.cpp

```
#include "RotationFigure.h"
RotationFigure::RotationFigure(const std::string& n)
    : name(n)
{}
```

Рис. 3-код файлу RotationFigure.cpp

Cone.h

```
1 #pragma once
2 #include "RotationFigure.h"
3
4 class Cone : public RotationFigure {
5 private:
6     double radius;
7     double height;
8
9 public:
10    Cone(double r, double h);
11    double volume() const override;
12    void show() const override;
13    ~Cone() override; // ! ОБОВ'ЯЗКОВО
14};
```

Рис. 4-код файлу Cone.h

Cone.cpp

```
1 #include "Cone.h"
2 #include <iostream>
3
4 Cone::Cone(double r, double h)
5     : RotationFigure("Конус"), radius(r), height(h) {
6 }
7
8 double Cone::volume() const {
9     return 3.141592653589793 * radius * radius * height / 3.0;
10}
11
12 void Cone::show() const {
13     std::cout << "Фигура обертання: " << name << std::endl;
14     std::cout << "Радіус основи: " << radius << std::endl;
15     std::cout << "Висота: " << height << std::endl;
16     std::cout << "Об'єм: " << volume() << std::endl;
17 }
18
19 Cone::~Cone() {
20 }
```

Рис. 5-код файлу Cone.cpp

RotationFigures.cpp (main)

```
1 #include <iostream>
2 #include <locale>
3 #include "Cone.h"
4
5 int main() {
6     setlocale(LC_ALL, "");
7
8     RotationFigure* figures[3];
9
10    figures[0] = new Cone(3.0, 5.0);
11    figures[1] = new Cone(2.5, 4.0);
12    figures[2] = new Cone(4.0, 6.5);
13
14    std::cout << "==== Фігури обертання ===\n\n";
15
16    for (int i = 0; i < 3; i++) {
17        figures[i]->show();
18        std::cout << "-----\n";
19    }
20
21    for (int i = 0; i < 3; i++) {
22        delete figures[i];
23    }
24
25    return 0;
26 }
```

Рис. 5-код файлу RotationFigures.cpp (main)

Результат роботи програми

```
Microsoft Visual Studio Debug + - X
==== Ф?гури обертання ===
Ф?гура обертання: Конус
Рад?ус основи: 3
Висота: 5
Об'єм: 47.1239
-----
Ф?гура обертання: Конус
Рад?ус основи: 2.5
Висота: 4
Об'єм: 26.1799
-----
Ф?гура обертання: Конус
Рад?ус основи: 4
Висота: 6.5
Об'єм: 108.909
-----
D:\Desktop\студент\00П\На 27 12 25\OOP_RotationFigures\x64\Debug\OOP_RotationFigures.exe (process 11336) exited with code 0 (0x0).
Press any key to close this window . . .
```

шаблонна функція пошуку елементів у діапазоні

Структура проекта

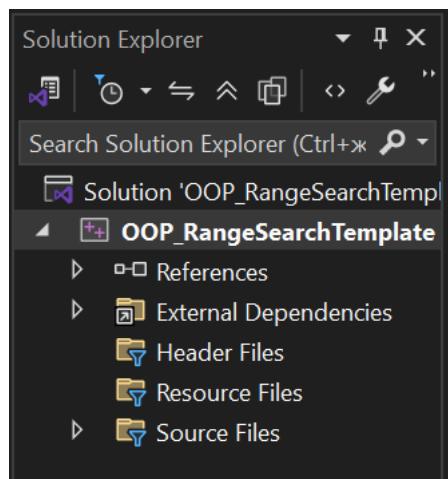


Рис. 1-Структура проекту у Visual Studio

Код програми RangeSearchTemplate

```

OOP.RangeSearchTemplate.cpp  What's New? (Global Scope) main()
1 #include <iostream>
2 #include <cmath>
3
4 using namespace std;
5
6 // Шаблонна функція пошуку елементів у діапазоні [low; high]
7 template <typename T>
8 int findInRange(const T arr[], int n, T low, T high) {
9     int count = 0;
10
11     cout << "Діапазон [" << low << "; " << high << "]\n";
12     cout << "Елементи у діапазоні: ";
13
14     for (int i = 0; i < n; i++) {
15         if (arr[i] >= low && arr[i] <= high) {
16             cout << arr[i] << " ";
17             count++;
18         }
19     }
20
21     if (count == 0) {
22         cout << "(немає)";
23     }
24
25     cout << "\nКількість знайдених: " << count << "\n";
26     return count;
27 }
28
29 // Друк масиву
30 template <typename T>
31 void printArray(const T arr[], int n) {
32     cout << "{ ";
33     for (int i = 0; i < n; i++) {
34         cout << arr[i];
35         if (i != n - 1) cout << ", ";
36     }
37     cout << " }";
38 }
39
40 int main() {
41     setlocale(LC_ALL, "Ukrainian");
42
43     // int
44     int a[] = { 1, 5, 9, -2, 7, 3, 10, 4 };
45     int n1 = sizeof(a) / sizeof(a[0]);
46
47     cout << "==== Перевірка для типу int ===\n";
48     cout << "Масив: ";
49     printArray(a, n1);
50     cout << "\n";
51     cout << "findInRange(a, n1, 3, 8);";
52     cout << "-----\n\n";
53
54     // double
55     double b[] = { 1.2, 5.5, 9.1, -2.3, 7.0, 3.14, 10.8, 4.4 };
56     int n2 = sizeof(b) / sizeof(b[0]);
57
58     cout << "==== Перевірка для типу double ===\n";
59     cout << "Масив: ";
60     printArray(b, n2);
61     cout << "\n";
62     cout << "findInRange(b, n2, 3.0, 7.5);";
63     cout << "-----\n\n";
64
65     // char
66     char c[] = { 'a', 'f', 'z', 'b', 'm', 'c', 'x' };
67     int n3 = sizeof(c) / sizeof(c[0]);
68
69     cout << "==== Перевірка для типу char ===\n";
70     cout << "Масив: ";
71     printArray(c, n3);
72     cout << "\n";
73     cout << "findInRange(c, n3, 'c', 'm');";
74     cout << "-----\n\n";
75
76     return 0;
77 }

```

Рис. 2-код файлу RangeSearchTemplate

Результат роботи програми

```
Microsoft Visual Studio Debug < + >

==== Перевірка для типу int ===
Масив: { 1, 5, 9, -2, 7, 3, 10, 4 }
Діапазон [3; 8]
Елементи у діапазоні: 5 7 3 4
Кількість знайдених: 4
-----
==== Перевірка для типу double ===
Масив: { 1.2, 5.5, 9.1, -2.3, 7, 3.14, 10.8, 4.4 }
Діапазон [3; 7.5]
Елементи у діапазоні: 5.5 7 3.14 4.4
Кількість знайдених: 4
-----
==== Перевірка для типу char ===
Масив: { a, f, z, b, m, c, x }
Діапазон [c; m]
Елементи у діапазоні: f m c
Кількість знайдених: 3
-----
D:\Desktop\студент\00П\На 27 12 25\00P_RangeSearchTemplate\x64\Debug\00P_RangeSearchTemplate.exe (process 1940) exited with code 0 (0x0).
Press any key to close this window . . .
```

Створити шаблон класу для зберігання пари чисел різних типів.

Структура проекта

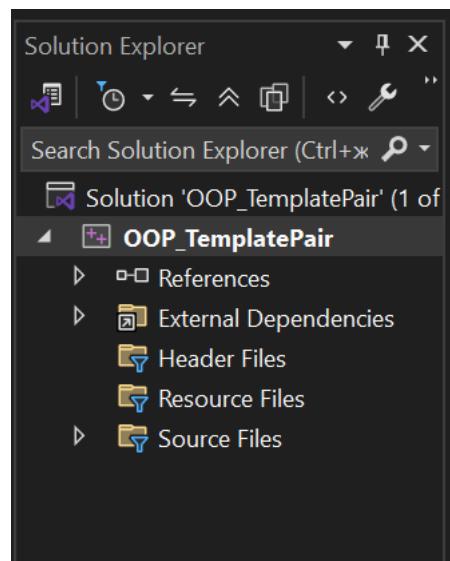


Рис. 1-Структура проекту у Visual Studio

Код програми TemplatePair

```

1 //include <iostream>
2 //include <cmath>
3 //include <string>
4
5 using namespace std;
6
7 // === Шаблон класу Pair ===
8 // Зберігає два значення різних типів
9 template <typename T1, typename T2>
10 class Pair {
11 private:
12     T1 first;
13     T2 second;
14
15 public:
16     Pair(T1 f, T2 s) : first(f), second(s) {}
17
18     T1 getFirst() const {
19         return first;
20     }
21
22     T2 getSecond() const {
23         return second;
24     }
25
26     void show() const {
27         cout << "Перше значення: " << first << endl;
28         cout << "Друге значення: " << second << endl;
29     }
30 };

```

```

31
32 int main() {
33     // Установлення коректної української мови
34     setlocale(LC_ALL, "");
35     system("chcp 65001 > nul");
36
37     cout << "==== Шаблон класу Pair (пара чисел) ====\n\n";
38
39     // 1) int + double
40     Pair<int, double> p1(10, 3.14);
41     cout << "Пара 1 (int + double):\n";
42     p1.show();
43     cout << endl;
44
45     // 2) double + int
46     Pair<double, int> p2(5.75, 20);
47     cout << "Пара 2 (double + int):\n";
48     p2.show();
49     cout << endl;
50
51     // 3) long + float
52     Pair<long, float> p3(1000000L, 2.5f);
53     cout << "Пара 3 (long + float):\n";
54     p3.show();
55     cout << endl;
56
57     // 4) string + int (не лише числа)
58     Pair<string, int> p4("Значення", 55);
59     cout << "Пара 4 (string + int):\n";
60     p4.show();
61     cout << endl;
62
63     return 0;
64 }

```

Рис. 2-код файлу TemplatePair

Результат роботи програми

```

Microsoft Visual Studio Debug

==== Шаблон класу Pair (пара чисел) ====
Пара 1 (int + double):
Перше значення: 10
Друге значення: 3.14

Пара 2 (double + int):
Перше значення: 5.75
Друге значення: 20

Пара 3 (long + float):
Перше значення: 1000000
Друге значення: 2.5

Пара 4 (string + int):
Перше значення: Значення
Друге значення: 55

D:\Desktop\студент\00P\На 27 12 25\OOP_TemplatePair\x64\Debug\OOP_TemplatePair.exe (process 13752) exited with code 0 (0x0).
Press any key to close this window . . .

```