

ЛАБОРАТОРНА РОБОТА №7

Тема: Створення власних класів. Наслідування і віртуальні функції.

Виконав: студент групи ALK-43
Вівчар Вадим Вікторович

Мета роботи

Навчитися створювати власні класи у мові C++,
використовувати наслідування, віртуальні функції,
інкапсуляцію, перевантаження операторів і роботу з
массивами об'єктів.

Завдання

1. Знайти всі пари дружніх чисел на відрізку [a; b].
2. Створити клас Student — сортування за середнім балом і
відбір студентів з оцінками 4 або 5.
3. Створити клас Complex — виконати операції +, -, *, /,
спряження.

Задача 1. Дружні числа

```
#include <bits/stdc++.h>
using namespace std;
typedef long long int64;

int64 sumDiv(int64 n) {
    if (n <= 1) return 0;
    int64 s = 1;
    for (int64 d = 2; d * d <= n; ++d)
        if (n % d == 0) {
            s += d;
            if (d * d != n) s += n / d;
        }
    return s;
}

int main() {
    long long a, b;
    cin >> a >> b;
    if (a > b) swap(a, b);
    for (long long m = a; m <= b; ++m) {
        long long n = sumDiv(m);
```

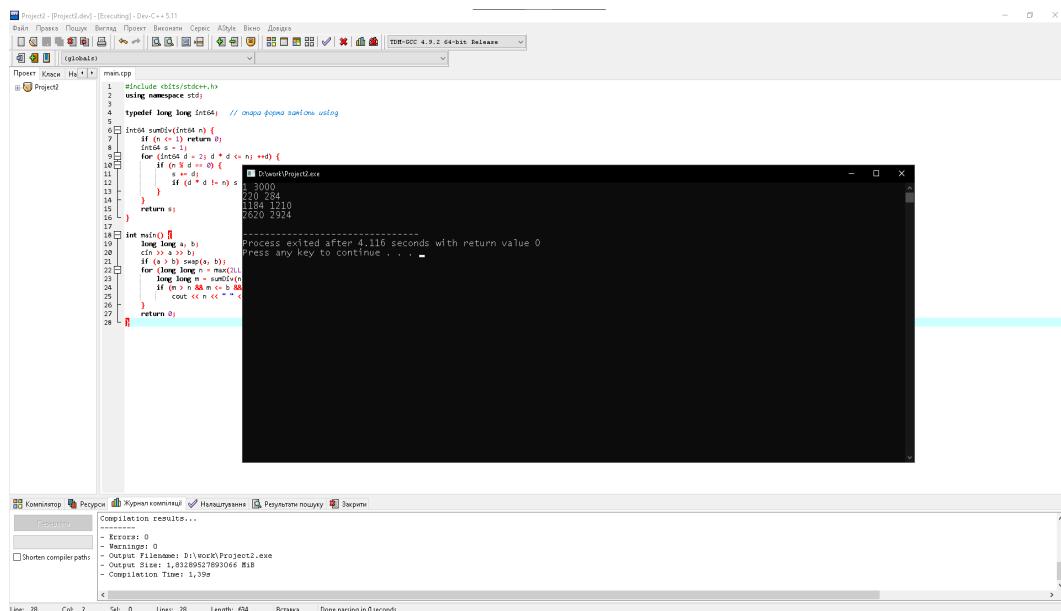
```

        if (m < n && n <= b && sumDiv(n) == m)
            cout << m << " " << n << "\n";
    }
    return 0;
}

```

Результати виконання програми:

220 284
1184 1210
2620 2924



Задача 2. Клас Student

```

#include <bits/stdc++.h>
using namespace std;

struct Student {
    string name, group;
    int marks[5];

    double avg() const {
        int s = 0;
        for (int i = 0; i < 5; i++) s += marks[i];
        return s / 5.0;
    }

    bool only4or5() const {
        for (int i = 0; i < 5; i++)
            if (marks[i] != 4 && marks[i] != 5) return false;
        return true;
    }
};

```

```

int main() {
    Student v[4] = {
        {"Ivanov I.I.", "ALK-43", {5,4,5,5,4}},
        {"Petrenko P.P.", "ALK-42", {3,4,5,4,5}},
        {"Sydorenko S.S.", "ALK-41", {4,4,4,4,4}},
        {"Melnyk M.M.", "ALK-43", {5,5,5,5,5}}
    };

    int n = 4;
    for (int i = 0; i < n - 1; i++)
        for (int j = i + 1; j < n; j++)
            if (v[i].avg() > v[j].avg())
                swap(v[i], v[j]);

    cout << "Sorting by average grade:\n";
    for (int i = 0; i < n; i++)
        cout << v[i].name << " " << v[i].group << " avg=" << v[i].avg() << "\n";

    cout << "\nStudents with only 4 or 5 marks:\n";
    for (int i = 0; i < n; i++)
        if (v[i].only4or5())
            cout << v[i].name << " " << v[i].group << "\n";

    return 0;
}

```

Результати виконання програми:

Sorting by average grade:

Sydorenko S.S. ALK-41 avg=4

Petrenko P.P. ALK-42 avg=4.2

Ivanov I.I. ALK-43 avg=4.6

Melnyk M.M. ALK-43 avg=5

Students with only 4 or 5 marks:

Sydorenko S.S. ALK-41

Ivanov I.I. ALK-43

Melnyk M.M. ALK-43

```
D:\work\Project2.exe
Sorting by average grade:
Sydorenko S.S. ALK-41 avg=4
Petrenko P.P. ALK-42 avg=4.2
Ivanov I.I. ALK-43 avg=4.6
Melnik M.M. ALK-43 avg=5

Students with only 4 or 5 marks:
Sydorenko S.S. ALK-41
Ivanov I.I. ALK-43
Melnik M.M. ALK-43

Process exited after 0.1073 seconds with return value 0
Press any key to continue . . .
```

Задача 3. Клас Complex

```
#include <bits/stdc++.h>
using namespace std;

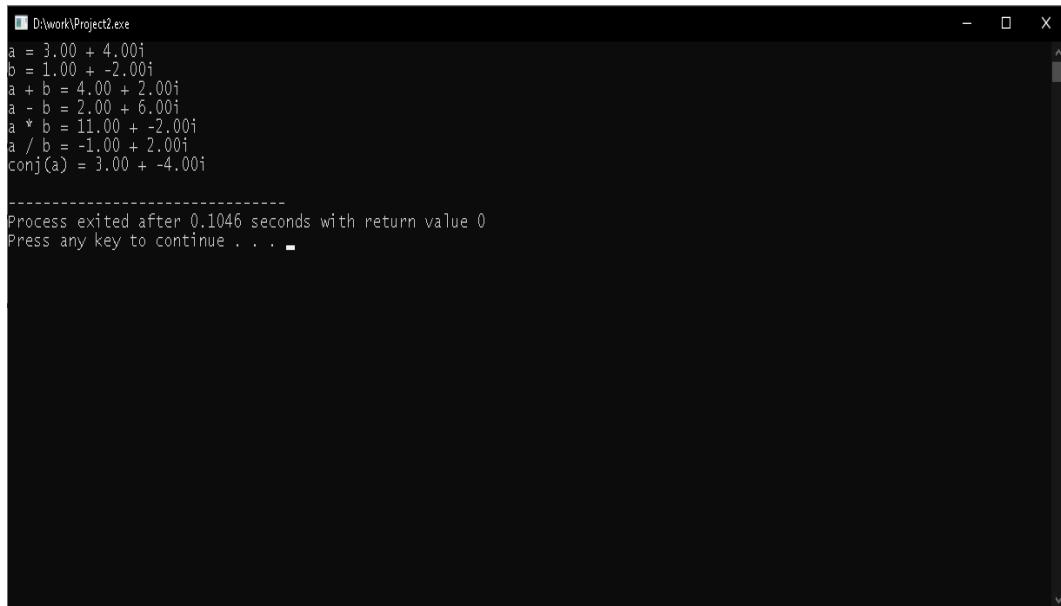
struct Complex {
    double re, im;
    Complex(double r = 0, double i = 0) { re = r; im = i; }

    Complex operator+(Complex z) const { return Complex(re + z.re, im + z.im); }
    Complex operator-(Complex z) const { return Complex(re - z.re, im - z.im); }
    Complex operator*(Complex z) const { return Complex(re*z.re - im*z.im, re*z.i);
    Complex operator/(Complex z) const {
        double d = z.re*z.re + z.im*z.im;
        return Complex((re*z.re + im*z.im)/d, (im*z.re - re*z.im)/d);
    }
    Complex conj() const { return Complex(re, -im); }
};

int main() {
    Complex a(3, 4), b(1, -2);
    cout << fixed << setprecision(2);
    cout << "a = " << a.re << " + " << a.im << "i\n";
    cout << "b = " << b.re << " + " << b.im << "i\n";
    cout << "a + b = " << (a + b).re << " + " << (a + b).im << "i\n";
    cout << "a - b = " << (a - b).re << " + " << (a - b).im << "i\n";
    cout << "a * b = " << (a * b).re << " + " << (a * b).im << "i\n";
    cout << "a / b = " << (a / b).re << " + " << (a / b).im << "i\n";
    cout << "conj(a) = " << a.conj().re << " + " << a.conj().im << "i\n";
    return 0;
}
```

Результати виконання програми:

```
a = 3.00 + 4.00i
b = 1.00 + -2.00i
a + b = 4.00 + 2.00i
a - b = 2.00 + 6.00i
a * b = 11.00 + -2.00i
a / b = -1.00 + 2.00i
conj(a) = 3.00 + -4.00i
```



```
D:\work\Project2.exe
a = 3.00 + 4.00i
b = 1.00 + -2.00i
a + b = 4.00 + 2.00i
a - b = 2.00 + 6.00i
a * b = 11.00 + -2.00i
a / b = -1.00 + 2.00i
conj(a) = 3.00 + -4.00i

-----
Process exited after 0.1046 seconds with return value 0
Press any key to continue . . .
```

Висновок

У ході лабораторної роботи створено три програми, які демонструють роботу з класами, успадкуванням і перевантаженням операторів у C++. Отримано практичні навички реалізації інкапсуляції, наслідування та поліморфізму.