

---

10.05.2022

## 1 Задача първа

### 1.1 Какво ще изведе кода? Какво трябва да напишем на мястото на коментарите за да работят функциите?

```
class Base {
protected:
    int data;
public:
    Base() {
        data = 0;
    }
};

class Derive : public Base {
protected:
    int data;
public:
    Derive() {
        data = 1;
    }

    void print() const {
        std::cout << data << std::endl;
    }

    void setData(int newData) {
        // ...
    }

    void setParentData(int newParentData) {
        // ...
    }
};

int main() {
    Derive d;

    d.print();
}
```

## 1.2 Какво ще изведе програмата?

```
class Base {
protected:
    int data;
public:

    Base(int x) {
        data = x;
    }
};

class Derive : public Base {
public:
    int idx;
public:
    Derive() {
        idx = pow(log10(0x1e23f), log2(0x12e1a)) * (0x33e2 << 2);
    }
};

int main() {
    Derive d;
    std::cout << d.idx;
}
```

## 2 Задача втора

Да се създаде клас Natural който описва числата от  $\mathbb{N}$ . За съхранение на числото може да използвате unsigned.

Да се създаде клас Integer който наследява Natural и описва числата от  $\mathbb{Z}$ . За определяне на знака може да използвате булева променлива.

Да се създаде клас Real, който наследява Integer и описва числата от  $\mathbb{R}$ . За частта след десетичната запетая може да използвате unsigned.

Да се създаде клас, който наследява Real и описва числата от  $\mathbb{C}$ . За имажинерната част можете да използвате double или Real.

За всеки от тези класове да се дефинират:

1. Подходящи конструктори
2. Оператори за математически действия  $+$ ,  $-$ ,  $*$ ,  $/$
3. Оператор за сравнение  $==$ ,  $>$ ,  $<$ ,  $<=$ ,  $>=$
4. Оператор за извеждане на стандартния изход.

Да се дефинира клас Pair, който представлява наредена двойка от Integer числа. Нека първото число наричаме  $a$  а второто  $b$ . Да се дефинира функция  $\log$  на Pair, която пресмята  $\log_a b$ . Функцията връща Real.

Да се дефинират функции `floor` и `ceil` за класа `Real`. Те връщат число от тип `Integer` което е закръглено съответно надолу или нагоре.

**Пример:**

`floor(3,4) = 3`

`ceil(3,4) = 4`

Трябва ли да предефинираме `floor` и `ceil` за класа `Complex`? Ако трябва да ги предефинираме приемерте, че

`floor(a + bi) = floor(a) + floor(b)i`

`ceil(a + bi) = ceil(a) + ceil(b)i`

### 3 Бонус задача за домашно:

Една много добра и полезна задача:

<https://github.com/ivanahristova/oop-labs-fini/tree/main/11-inheritance>