

Блиц тест № 3 - ООП практикум
03.04.2024

Зад 1. За всяка от променливите посочете нейната видимост:

<pre>class A{ int a1; private: int a2; int a3; public: int a4; private: };</pre>	<pre>struct B{ int b1; private: int b2; int b3; public: int b4; private: };</pre>
--	---

a1		b1	
a2		b2	
a3		b3	
a4		b4	

Зад 2. Ще даде ли грешка компилаторът, ако дефинираме оператора за присвояване на Test така? Обосновете се.

```
class Test{
    int var = 0;
public:
    Test() = default;
    bool operator=(const Test& other){
        if(this == &other){
            return false;
        }
        var = other.var;
        return true;
    }

    int getVar() const {
        return var;
    }
    void setVar(int var) {
        this->var = var;
    }
};
```

Ще възникне ли грешка и при изпълнението изпълнението на main() функцията? Обосновете се.

```
int main(){
    Test t;
    Test t3(14);
}
```



```

    Test t4(7);
    t3 = (t4 = t);
}

```

Зад 3. Отговорете с истина или лъжа

Вярно ли е, че клас Test може да има конструктор от типа Test(Test other)?	
Възможно ли е в един клас да се дефинират няколко различни версии на оператора за присвояване?	
Задължително трябва да разпишем голяма 4-ка, ако имаме указател към функция като член-данна на класа	
Вярно ли е, че клас Test може да има конструктор от типа Test(Test& other)?	

Зад 4. За кои от следните класове е необходимо да се дефинира конструктор за копиране?

- A) `class A {private: std::string s; //...};`
- B) `class A {private: std::string* s; //...};`
- C) `class A {private: int a[100]; char string[255]; //...};`
- D) `class A {private: int* a[100]; //...};`
- E) `class A {private: int a[100]; //...};`
- F) `class A {private: int a[100]; char* string; //...};`

Зад 5. Какво ще се случи, ако деструкторът е `private`?

```

class A{
private:
~A();
};
class X{
    A a;
    int var;
};

```

Зад 6. Какво ще се отпечата на екрана? Подсказка: не забравяйте за return value optimizations!

```

class X{
    int var = 0;
public:

```



```

X(int _var): var(_var){
    std::cout<< "X(int) " <<var<<std::endl;
}
X(){
    std::cout<< "X()" <<std::endl;
}
X(const X& other){
    var = other.var;
    std::cout << "X(X) " << var <<std::endl;
}
~X(){
    std::cout << "~X() " << var <<std::endl;
}
};

```

```

X foo(int param){
    X x1(4);
    X x2;
    if(param >= 5){
        return x1;
    }
    return x2;
}

int main() {
    X x1 = foo(6);
}

```

Зад 7. Има ли Memory Leak? Ако да, защо? Ако не, защо?

```
constexpr int MAX_SIZE = 1024;
```

```

char* readLineFromConsole()
{
    char buff[MAX_SIZE];
    std::cin.getline(buff, MAX_SIZE, '\n');

    return buff;
}

int main()
{
    char* line = readLineFromConsole();
    std::cout << line;
}

```



Зад 8. Има ли Memory Leak? Ако да, защо? Ако не, защо?

```
constexpr int MAX_SIZE = 1024;

struct Test
{
    char* content = nullptr;
};

size_t getFileSize(std::ifstream& ifs)
{
    size_t currPos = ifs.tellg();
    ifs.seekg(0, std::ios::end);

    size_t size = ifs.tellg();
    ifs.seekg(currPos);

    return size;
}

void readFromFile(const char* fileName, Test& t)
{
    std::ifstream ifs(fileName);
    char* data = new char[getFileSize(ifs)];

    if (!ifs.is_open())
    {
        std::cout << "Error!" << std::endl;
        delete[] data;

        return;
    }

    int index = 0;
    while (true)
    {
        char ch = ifs.get();

        if (ifs.eof())
        {
            break;
        }

        data[index] = ch;
        index++;
    }
    data[index] = '\0';

    t.content = new char[strlen(data)];
    strcpy(t.content, data);
}

int main()
{
    Test t;
    readFromFile("file.txt", t);
}
```

