

Вам дан класс Foo:

```
struct Foo {  
    void say() const { std::cout << "Foo says: " << msg <<  
        "\n"; }  
protected:  
    Foo(const char *msg) : msg(msg) { }  
private:  
    const char *msg;  
};
```

Как видно, создатель класса не хотел чтобы его использовали и "спрятал" конструктор класса. Но вам очень нужен объект этого класса, чтобы передать его в функцию `foo_says`:

```
void foo_says(const Foo& foo) { foo.say(); }
```

В этом задании вам нужно реализовать функцию `get_foo` (сигнатура которой намерено не приводится в задании полностью, вам нужно подумать и вывести ее самостоятельно) так, чтобы следующий код компилировался и работал:

```
foo_says(get_foo(msg));
```

Где `msg` — произвольная C-style строка.

**Требования к реализации:** при выполнении задания вам разрешено вводить любые вспомогательные функции и классы. Запрещено изменять определение класса `Foo` или функции `foo_says`. Вводить или выводить что-либо не нужно. Реализовывать функцию `main` не нужно.

---

Нажмите, чтобы начать решать ☐