**1) [memory\_example\_3](https://github.com/AntonC9018/uniCourse_dataStructuresAndAlgorithms/blob/master/en/05_programming_fundamentals/memory_example_3) объяснить программу.**

**#include <iostream>**

**int a = 0; // объявляется глобальная перемнная и иницилизируется 0**

**void staticMemory() // функция, которая выводит значения a и добавляет 1.**

**{**

**cout << "a: " << a << endl; a += 1; }**

**int main() {**

**staticMemory(); // prints 0 // вызов 1 и каждый вызов +1**

**staticMemory(); // prints 1 // вызов 2**

**staticMemory(); // prints 2 // вызов 3**

**return 0;**

**}**

Программа объявляет глобальную переменную a и инициализирует её значением 0. Затем она определяет функцию staticMemory(), которая выводит значение a и увеличивает его на 1. Функция main() вызывает функцию staticMemory() три раза, и увеличивается на 1 каждый раз.

2)**Чем переменная a в static\_1 отличается от переменной из static\_2? Как сделать пример static\_2 эквивалентным первому ?**

В первой программе переменная a объявлена вне функции staticMemory(). Это означает, что она является локальной переменной функции main(). Поэтому в первой программе переменная a будет иметь значение 0 при первом вызове функции staticMemory(), значение 1 при втором вызове и значение 2 при третьем вызове.

Во второй программе переменная a объявлена внутри функции staticMemory(). Это означает, что она является статической переменной функции staticMemory(). Поэтому во второй программе переменная a будет иметь значение 0 при первом вызове функции staticMemory(), значение 1 при втором вызове и значение 2 при третьем вызове.