**Лабораторная работа №1**

**Цель:**

Определить закономерность размещения переменных в памяти компьютера.

**Описание:**

**1)** Глобальные переменные и функции находятся в основном разделе памяти, записываются по мере объявления(снизу верх);

**2)** Переменные могут определяться внутри функции(локальные), вне функции и в определении параметров функции;

**3)** Локальные переменные и параметры функции записываются в стек, в разном порядке: локальные переменные записываются по мере объявления, а параметры функций – начиная с последнего объявленного.

**4)** Существует разница между адресами объявленных подряд переменных, которая соответствует размеру этой переменной(количеству байт, выделяемых под данный тип).

**Результат:** обобщим результат работы написанной программы в одну таблицу.

|  |  |
| --- | --- |
| **адрес p1** | **0xbfeb0950** |
| **адрес p2** | **0xbfeb0954** |
| **адрес p3** | **0xbfeb0958** |
| **адрес глобальной переменной** | **0x804a024** |
| **адрес функции func** | **0x804844c** |
| **адрес функции main** | **0x8048444** |
| **адрес l1** | **0xbfdaf17c** |
| **адрес l2** | **0xbfdaf178** |
| **адрес l3** | **0xbfdaf174** |

Программа выводит на экран таблицу с адресами переменных и функций. Можно увидеть, что функции и глобальные переменные находятся в основном разделе памяти, по мере объявления адреса увеличиваются. Наибольший адрес имеют локальные переменные функции main. Вначале мы вызываем параметры функции(p1,p2,p3), из 3-х объявленных параметров, последний имеет наибольший адрес. При объявлении локальных переменных (l1,l2,l3) можно заметить, что чем позже мы объявляем переменную, тем меньше адрес переменная имеет. Так же можно заметить разницу между параметрами функции, объявленными подряд, которая везде составляет 4 байта. Между глобальными переменными и параметром функции имеется небольшое отступление.

**Вывод:**

**1)** Глобальные переменные и функции находятся в основном разделе памяти(на это указывает порядок записи и адреса).

**2)** Наибольшим адресам соответствует стек.

**3)** Разница между адресами подряд объявленных переменных **действительно** соответствует размеру переменной.