|  |
| --- |
| Пензенский государственный университет  Кафедра «Вычислительная техника» |
| Отчёт |
| По лабораторной работе №1  по дисциплине: «Логика и основы алгоритмизации в ИЗ»  на тему: «Простые структуры данных». |
|  |

**Выполнил студент группы 21ВВ3:В**

Димитренко Н.С.

**Приняли:**

Митрохин М.А.

Юрова О.В.

**Пенза 2022**

**Задание 1:** написать программу, вычисляющую разницу между максимальным и минимальным элементами массива.

**Задание 2:** написать программу, реализующую инициализацию массива

случайными числами.

**Задание 3:** написать программу, реализующую создание массива произвольного размера, вводимого с клавиатуры.

**Задание 4:** написать программу, вычисляющую сумму значений в каждом столбце (или строке) двумерного массива.

**Задание 5:** написать программу, осуществляющую поиск среди структур student структуру с заданными параметрами (фамилией, именем и т.д.).

**Теоретический материал:**

Динамический массив

Для использования функций динамического распределения памяти необходимо

подключение библиотеки malloc.h; или stdlib.h (в зависимости от компилятора и

используемого стандарта языка).

Функции динамического выделения памяти

void\* malloc(РазмерМассиваВБайтах);

void\* calloc(ЧислоЭлементов, РазмерЭлементаВБайтах);

выделяют блок памяти, размером РазмерМассиваВБайтах или

ЧислоЭлементов\*РазмерЭлементаВБайтах байт, и возвращает указатель на начало

блока.

Элементами массива могут быть как данные простых типов, так и составных типов

(например, структуры), поэтому для точного определения размера элемента массива в

общем случае используется функция

int sizeof(ТипЭлемента);

Эта функция возвращает количество байт, занимаемое элементом указанного типа.

Память, динамически выделенная с использованием функций calloc(), malloc(),

должна быть освобождена после окончания использования функцией

free(указатель);

Для генерации случайных чисел в языке Си используется функция rand(), которая

содержится в библиотеке stdlib.h.

Функция возвращает случайное целое число в диапазоне от нуля до RAND\_MAX.

Глобальная константа RAND\_MAX является максимальным значением, которое может

возвращаться функцией rand, это значение определено в заголовочном файле stdlib.h.

Чтобы ограничить сверху случайные числа, можно воспользоваться операцией

получения остатка от деления. Остаток от деления на число всегда меньше этого числа.

Поэтому для ограничения сверху случайных чисел возьмём остаток от деления на

граничное значение.

int k;

k = rand()%100;

В приведенном примере значение k не превысит 99. Т.е. для rand()%n возвращается

любое число в диапазоне [0, n). Если необходимо получить числа в диапазоне [0, n], то

необходимо изменить выражение на rand()%(n+1).

Для получения чисел из некоторого отрезка [min, max] необходимо найти остаток

от деления на длину этого отрезка и выполнить сдвиг возвращаемого остатка на нижнюю

границу отрезка

int k;

k = min + rand()%(max-min+1);

При повторных запусках программы функция rand() будет возвращать одинаковые

последовательности чисел. Для получения уникальных последовательностей необходимо

инициализировать генератор случайных чисел различными значениями. Для этого

используется функция srand(), аргументом которой является инициализирующее целое

число.

В качестве уникального для каждого запуска программы числа чаще всего

используется текущее время, возвращаемое функцией time(), прототип которой описан в

библиотеке time.h. Передав в time() в качестве параметра NULL, мы получим целое число,

которое можно передать в srand(), установив, таким образом уникальное

инициализирующее значение генератора

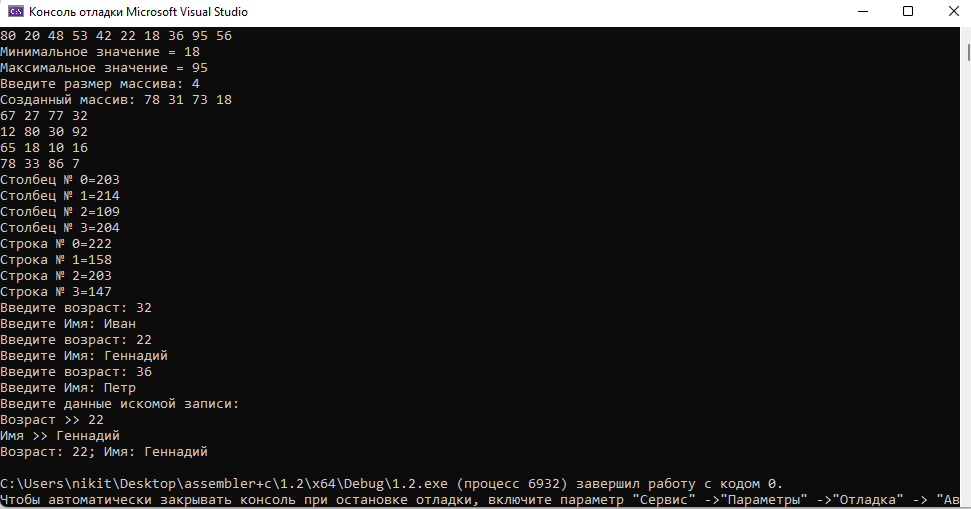
srand(time(NULL));

int k;

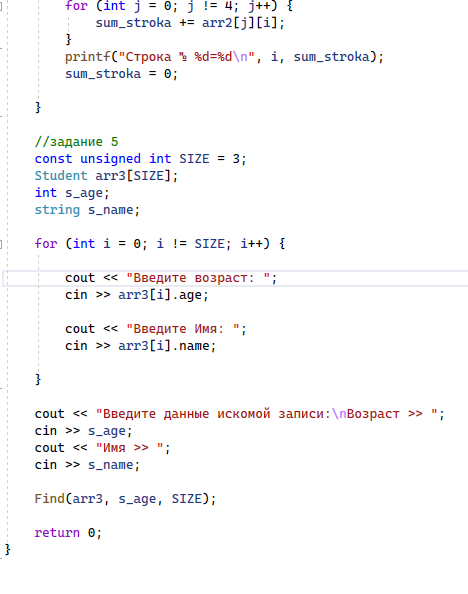
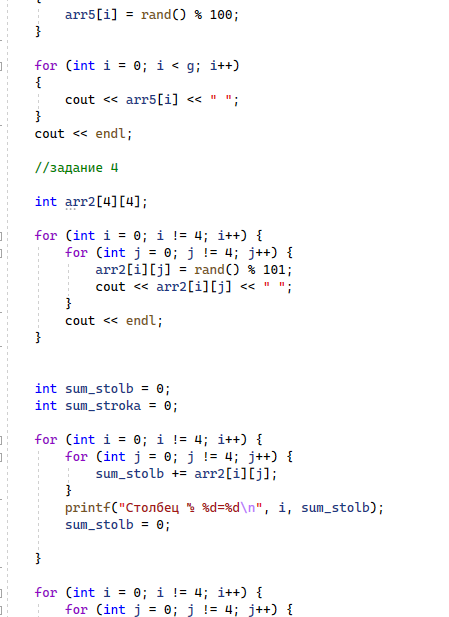
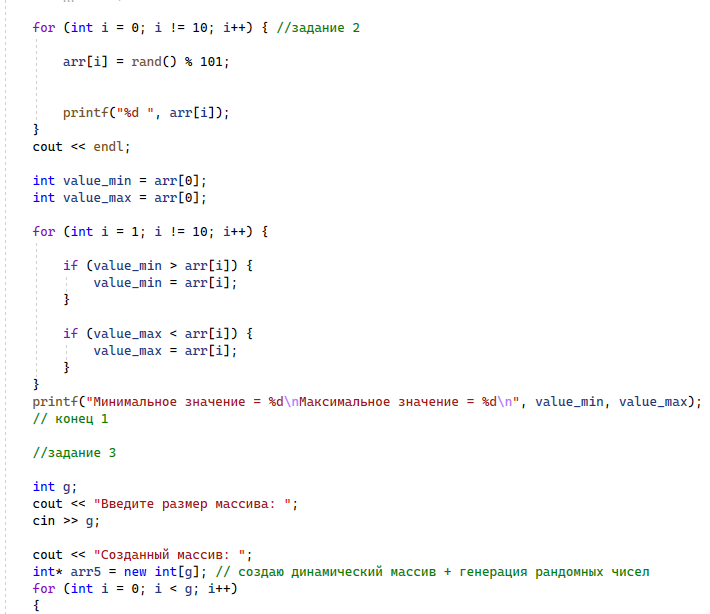
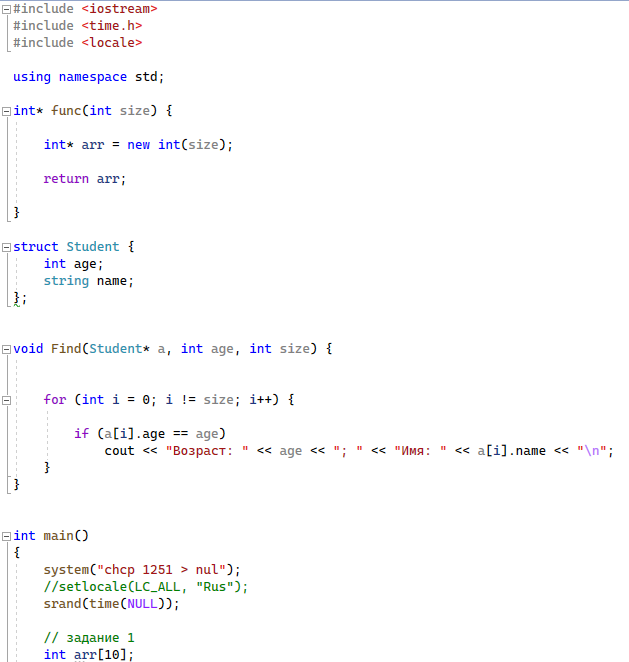
k = rand();

теперь генерируемая последовательность чисел k будет уникальной.

**Результат работы программы:**



**Листинг:**



**Вывод:** Освоил основы структур данных, научился создавать динамические массивы произвольного размера, пользоваться многомерными массивами.