# Практическое задание № 1

**Тема 1.2. Динамические структуры данных: «линейные списки», «стек», «очередь», «дерево»**

**Цель работы:** ознакомится с различными динамическими структурами данных и методами работы с ними; научиться решать задачи с использованием динамических структур данных.

**Формулировка задания № 1**

1. Сформировать ***динамическую структуру данных.***

Описать структуру с именем STUDENT, содержащую следующие поля: фамилия и инициалы; номер группы; успеваемость (массив из пяти элементов). Написать программу, выполняющую следующие действия:

* ввод с клавиатуры данных в массив, состоящий из десяти структур типа STUDENT;
* записи должны быть упорядочены по возрастанию номера группы;
* вывод на экран фамилий и номеров групп для всех студентов, включенных в массив, если средний балл студента выше 4.0;
* если таких студентов нет, вывести соответствующее сообщение.

2. Информацию, составляющую содержание структуры и введенную с клавиатуры, организовать в виде ***стека***, односвязного или двусвязного ***списка*** или в виде ***бинарного дерева.***

3. Обработку информации в соответствии с заданием реализовать в динамической структуре данных.

**Указания к выполнению работы**

При выполнении задания необходимо написать программу на языке C++, в которой производятся формирование структуры и её полей в виде стека, односвязного или двусвязного списка в соответствии с постановкой задачи, ввод элементов структуры с учетом типа данных, их обработка и вывод на экран в указанном формате. Для хранения данных списков следует использовать ресурсы динамической памяти. Ввод данных осуществляется с клавиатуры с учетом требований ко входным данным, содержащимся в постановке задачи. Ограничениями на входные данные являются максимальный размер строковых данных, диапазоны числовых типов полей структуры и допустимый размер области динамической памяти в языке C++.

Задание нужно решать с опорой на изученные методы создания, вывода и обработки динамических данных в языке C++.

Выполнение задания необходимо проводить в следующей последовательности:

* выбрать метод решения задачи;
* разработать графическую схему алгоритма;
* записать разработанный алгоритм на языке C++;
* разработать контрольный тест к программе;
* отладить программу;
* представить отчет по работе.

**Требования к отчету**

Структура отчета должна соответствовать приведённой выше последовательности и содержать:

* титульный лист?
* алгоритм решения задачи (схема алгоритма выполняется по ЕСПД (ГОСТ 19.003-80 и ГОСТ 19.002-80));
* листинг программы;
* контрольный тест;
* выводы.