

**ЗАДАНИЕ НА КУРСОВОЕ ПРОЕКТИРОВАНИЕ**  
**по дисциплине «Программирование»**  
**(2-ой семестр)**

**Тема: разработка базы данных «Студенческий поток»**

Общая часть задания

Студент потока характеризуется следующими данными:

- ФИО (до 50 символов);
- номер группы;
- набор из пяти оценок за последнюю сессию (без указания предметов);
- размер стипендии.
- дополнительная информация (выбирается индивидуально каждым студентом, например, телефон, эл. почта, год рождения и т.п.)

Необходимо:

- 1) разработать (и программно реализовать) динамические структуры данных и алгоритмы их обработки, позволяющие поддерживать выполнение следующих функций:
  - консольный ввод/вывод данных о всех студентах потока;
  - файловый ввод/вывод данных о потоке;
  - редактирование данных о студентах и группах потока, включающее операции добавления/удаления групп и студентов;
- 2) разработать и программно реализовать алгоритмы обработки базы данных, предусмотренные персональным заданием

Примеры персональных заданий

Вариант	Задание
1	Вывести в алфавитном порядке фамилии тех студентов, чей личный средний балл за сессию ниже среднего балла по потоку
2	Вывести в порядке возрастания номера тех групп, в которых процент хорошистов и отличников превышает заданное значение
3	Вывести в алфавитном порядке фамилии всех «круглых» отличников потока, начинающиеся с заданного сочетания букв
4	Вывести в порядке возрастания номера групп, в которых соотношение

	«суммарный объем стипендии» / «общее количество баллов» превышает заданное значение
5	Вывести перечень номеров групп, упорядоченный по значению показателя «число двоечников» / «списочная численность группы»
6	Рассчитать для каждой из групп потока процент отличников
7	Вывести на экран номера всех групп, не имеющих двоечников, в порядке убывания среднего (по группе) балла
8	Вывести в алфавитном порядке фамилии N (вводится с клавиатуры) первых по среднему баллу студентов потока
9	Вывести перечень групп потока с указанием (для каждой группы) процента суммарной стипендии группы по отношению к суммарной стипендии потока
10	Привести отсортированный по фамилиям и номерам групп список всех студентов потока, получающих стипендию (с указанием ее размера)
11	Вывести на экран список студентов группы, которая содержит максимальное (в пределах потока) количество хорошистов и отличников
12	Вывести список всех студентов потока, отсортированный по размеру стипендий и алфавиту
13	Вывести (с разбивкой по группам) фамилии всех студентов, не получающих стипендии
14	Вывести на экран в порядке возрастания номера групп, в которых процент троечников и двоечников превышает заданное с клавиатуры значение
15	Рассчитать и вывести в табличном виде распределение количества студентов по набранным ими суммам баллов
16	Вывести перечень всех фамилий студентов потока, сгруппированный по размеру стипендии
17	Вывести в алфавитном порядке фамилии студентов, которые в своих группах имеют максимальное значение показателя «средний балл» / «объем стипендии»
18	Вывести в порядке роста номеров перечень всех групп потока, в которых количество студентов, не получающих стипендию, превышает заданное число
19	Вывести в алфавитном порядке список той группы, в которой средний балл является максимальным по потоку
20	Рассчитать и вывести на экран (для каждой группы) количество студентов, у

	которых нет двоек, и большинство оценок – пятерки
21	Вывести в алфавитном порядке (с указанием номера группы) фамилии всех студентов, имеющих задолженности по сессии
22	Вывести перечень групп, упорядоченный по значению показателя «процент должников» (с указанием значения показателя)
23	Вывести распределение количества студентов по группам (в порядке убывания численности)
24	Вывести в алфавитном порядке фамилии студентов заданной группы, имеющих размер стипендии ниже средней по потоку
25	Вывести в порядке следования номеров групп фамилии тех студентов, у которых личный средний балл не превышает заданного значения

### Общие требования к программам

1) Программа должна поддерживать систему меню, пункты которых соответствуют выполнению функций, предусмотренных общей частью задания.

2) Предлагаемые структуры данных должны учитывать изначальную неопределенность возможного количества групп и студентов в группах, а также обеспечивать максимальную скорость процессов обработки данных, предусмотренных заданием.

3) Тексты программ должны содержать комментарии, объясняющие назначение основных функций, типов и объектов данных, функциональных блоков и т.п.

4) Представляемые тексты программ должны обеспечивать возможность их компиляции и построения в среде MS Visual Studio

Курсовая работа (документ MS WORD) оформляется по общим правилам (формления текста, рисунков, формул, таблиц) в виде пояснительной записки (например, [https://etu.ru/assets/files/3004\\_3\\_ShABLON-kursovika.doc](https://etu.ru/assets/files/3004_3_ShABLON-kursovika.doc)) и включает: титульный лист, описание задания, алгоритм и программы решения задачи, описание процесса тестирования и полученных результатов, список использованной литературы. Полный текст программы приводится в приложении «А». Вводимые при тестировании данные должны быть оформлены в виде отдельных файлов и также включаются в приложение «В».

См. Документы для обеспечения учебного процесса

<https://etu.ru/ru/obrazovatel'naya-deyatelnost/dokumenty-dlya-obespecheniya-uchebnogo-processa>

и

[Шаблон оформления курсовой работы \(курсового проекта\)](#)

[https://etu.ru/assets/files/3004\\_3\\_ShABLON-kursovika.doc](https://etu.ru/assets/files/3004_3_ShABLON-kursovika.doc)