

Разработать приложение, позволяющее организовать работу по учёту студентов некоторого подразделения института. Информация об учащемся, хранится в некотором описателе учащегося.

Описатель младшекурсника содержит следующую информацию: фамилия и инициалы студента, "индекс группы"; номер профилирующей кафедры; оценки за прошедшую сессию (максимум 5 чисел).

Описатель старшекурсника содержит следующую информацию: фамилия и инициалы студента, "индекс группы"; номер профилирующей кафедры; оценки за прошедшую сессию (максимум 4 числа); направление темы УИР (учебно-исследовательской работы); место выполнения; оценки руководителя и комиссии.

Описатель выпускника содержит следующую информацию: фамилия и инициалы студента, "индекс группы"; номер профилирующей кафедры; направление темы ДП (дипломного проекта); место выполнения; оценки руководителя, рецензента и ГЭК (государственной экзаменационной комиссии).

Каждый студент имеет уникальную характеристику – шифр студента (число). Информация обо всех студентах сведена в таблицу, каждый элемент которой содержит шифр и указатель на его описатель. Элементы таблицы упорядочены по возрастанию значений шифра.

Обеспечить выполнение следующих операций.

- ❖ Для таблицы:
 - включить новый элемент, не нарушая упорядоченности;
 - найти элемент по заданному шифру;
 - удалить элемент, заданный шифром;
 - показать содержимое таблицы.
- ❖ Для любого студента:
 - вывести информацию о студенте;
 - получить (вернуть в качестве результата) категорию студента;
 - получить (вернуть в качестве результата) информацию об оценках в соответствии с категорией студента; записать оценки соответствующего семестра;
 - изменить индекс учебной группы;
 - перевести студента в другую категорию учащихся.
- ❖ Для старшекурсников и выпускников:
 - получить (вернуть в качестве результата) информацию о теме индивидуальной работы (УИР или ДП) студента; изменить информацию о теме индивидуальной работы;
 - получить информацию о месте выполнения работы; изменить информацию о месте выполнения работы (УИР или ДП).
- ❖ Для приложения:
 - оформить поступление студента в институт;
 - оформить перевод студента на новый семестр (добавление новой записи с возможным изменением статуса);
 - отчислить студента из института (исключение записи из таблицы);
 - (*) получить отчёт обо всех студентах в краткой форме с указанием категории учащегося, используя класс-итератор.

Порядок выполнения работы

1. На основе описания задачи определить состав классов.
2. Разработать иерархию классов и схему их взаимодействия.
3. Для каждого класса определить его состояние и необходимые методы.
4. Разработать и отладить все классы. Приложение реализовать в виде простой диалоговой программы.
5. (*) Разработать и отладить класс, реализующий работу приложения. Предусмотреть в классе возможность сохранения таблицы в файле и ее восстановления.