Вариант 1

В отдельной библиотеке классов определить абстрактный класс Subject:  
**abstract class Subject**  
**public Subject(string lecturerName)** – конструктор, принимающей фамилию лектора по предмету  
автосвойство **string LecturerName** с закрытым **set** – фамилия лектора  
абстрактный метод **double GetMark(double nakop, double exam)** – возвращает оценку студента по его накопу и оценке за экзамен  
переопределить метод **ToString()** таким образом, чтобы он возвращал строку в формате «{Имя типа}: LecturerName = {LecturerName}»

Определить классы DiscreteMaths и Programming, производные от класса Subject, в них переопределить метод GetMark(double nakop, double exam) в соответствии со следующими формулами:  
**0.7 \* exam + 0.3 \* nakop** для класса **Programming**  
**nakop >= 3.5 ? nakop : exam** для класса **DiscreteMaths**

В основной программе на ввод принять число n – количество студентов на этих дисциплинах. Далее в n строк следуют оценки каждого студента, считать, что оценки по дискретной математике и программированию совпадают. Создать экземпляры классов Programming и DiscreteMaths, вывести на экран их строковое представление. Для каждого студента вывести его оценки по программированию и дискретной математике. В одной строке сначала вводится накоп, за ним оценка за экзамен. Выводить сначала оценку за программирование, затем за ДМ.

Пример входных данных:  
2  
10 10  
0 10  
Вывод:  
Programming: LecturerName = Dudarev  
DiscreteMaths: LecturerName = Avdoshin  
10 10  
7 10