

КОЛЛЕКЦИИ И ПЕРЕГРУЗКА ОПЕРАТОРОВ

ЛАБОРАТОРНАЯ РАБОТА №4

Задания 1-5 решаются отдельно от заданий 6 и 7 (в отдельном проекте). Решение задач 1-5 оформить в виде одного класса со статическими методами, решающими поставленные задачи. В классе могут присутствовать методы со спецификатором доступа `private` вспомогательного характера.

В **задании 5** исходные данные хранятся в текстовом файле, заполнение исходного файла необходимо организовать отдельным методом.

Задания 6 и 7 решаются в одном проекте отдельно от заданий 1-5 (задание 7 продолжает задание 6).

В **задании 6** необходимо реализовать определение класса (поля, свойства, конструкторы, перегрузка метода `ToString()` для вывода полей, заданный метод согласно варианту). Протестировать все методы, включая конструкторы (при тестировании методов классов в заданиях не забывайте проверять вводимые пользователем данные на корректность).

В **задании 7** добавить к реализованному в задании 6 классу указанные в варианте перегруженные операции. Протестировать все методы.

Необходимо решить по 1 задаче из каждого задания согласно вашему варианту. Задания 1,2,3,4 и 6 оцениваются по 1 баллу, задания 5 и 7 – 1,5 балла. Максимально за лабораторную работу можно получить 10 баллов (8 баллов за решение задач + 2 балла за оформление отчета).

Задание 1. List

1	Составить программу, которая удаляет из списка L все элементы E, если такие есть.
2	Составить программу, которая в списке L заменяет первое вхождение списка L1 (если такое есть) на список L2.
3	Даны упорядоченные списки L1 и L2. Вставить элементы списка L2 в список L1, не нарушая его упорядоченности.
4	Составить программу, которая удаляет из списка L за каждым вхождением элемента E один элемент, если такой есть, и он отличен от E.
5	Составить программу, которая удаляет из списка L все элементы с указанным значением.
6	Составить программу, которая проверяет, есть ли в списке L хотя бы два одинаковых элемента.
7	Составить программу, которая переносит в конец непустого списка L его первый элемент.
8	Составить программу, которая вставляет в список L за первым вхождением элемента E все элементы списка L, если E входит в L.
9	Составить программу, которая переворачивает список L, т.е. изменяет ссылки в этом списке так, чтобы его элементы оказались расположенными в обратном порядке.
10	Составить программу, которая в списке L из каждой группы подряд идущих одинаковых элементов оставляет только один.
11	Составить программу, которая формирует список L, включив в него по одному разу элементы, которые входят одновременно в оба списка L1 и L2.
12	Составить программу, которая формирует список L, включив в него по одному разу элементы, которые входят в список L1, но не входят в список L2.
13	Составить программу, которая формирует список L включив в него по одному разу элементы, которые входят в один из списков L1 и L2, но в то же время не входит в другой из них.
14	Оставить в списке только первые вхождения одинаковых элементов.
15	Определить количество различных элементов списка.
16	Повторить дважды каждое вхождение элемента E в список L.

Задание 2. LinkedList

1	Напечатать в обратном порядке элементы непустого списка L.
2	Сортировать элементов списка по возрастанию.
3	Подсчитать количество элементов списка L, у которых равные «соседи».
4	Определить, есть ли в списке L хотя бы один элемент, который равен следующему за ним (по кругу) элементу (первый элемент считать следующим для последнего).
5	В списке L переставить в обратном порядке все элементы между первым и последним вхождениями элемента E, если E входит в L не менее двух раз.

6	Удалить из списка L первое вхождение заданного элемента, если такой есть.
7	Из списка L, содержащего не менее двух элементов, удалить все элементы, у которых одинаковые «соседи» (первый и последний элементы считать соседями).
8	Добавить в начало и конец списка L новый элемент E.
9	В списке L справа и слева от элемента E вставить элемент F.
10	Построить двунаправленный список L по однонаправленному списку L1.
11	В конец непустого списка L добавить все его элементы, располагая их в обратном порядке (например, по списку из элементов 1, 2, 3 требуется построить список из элементов 1, 2, 3, 3, 2, 1).
12	Проверить симметричность участка списка с i-го по j-й элемент ($i < j$).
13	Удалить все элементы между минимальным и максимальным элементами.
14	Если у элемента со значением E "соседи" не равны, поменять их местами.
15	Если у элемента со значением E "соседи" не равны, заменить их на элемент F.
16	Поменять местами фрагменты списка между минимальным и максимальным элементами.

Задание 3. HashSet

1	В городе имеется n общеобразовательных учебных заведений, которые могут закупать компьютерную технику и оргтехнику. Есть перечень компьютерных фирм, где может производиться закупка. Ответить на следующие вопросы: 1) в каких фирмах закупка производилась каждым из заведений? 2) в каких фирмах закупка производилась хотя бы одним из заведений? 3) в каких фирмах ни одно из заведений не закупало компьютеры?
2	Есть перечень компьютерных игр. Студенты группы играют в какие-либо из этих игр. Известно для каждого студента, в какие игры он играет. Определить: <ul style="list-style-type: none"> • в какие игры из перечня играют все студенты группы; • в какие игры из перечня играют некоторые студенты группы; • в какие игры из перечня не играет ни один из студентов группы?
3	Задан некоторый набор блюд в кафе. Определить для каждого из блюд, какие из них заказывали все посетители, какие — некоторые из посетителей, и какие не заказывал никто.
4	Есть перечень стран, популярных у туристов. Определить для каждой страны, какие из них посетили все n туристов, какие — некоторые из туристов, и какие — никто из туристов.
5	Есть перечень фильмов. Определить для каждого фильма, какие из них посмотрели все n зрителей, какие — некоторые из зрителей, и какие — никто из зрителей.
6	Есть перечень музыкальных произведений. Определить для каждого произведения, какие из них нравятся всем n меломанам, какие — некоторым из меломанов, и какие — никому из меломанов.
7	Есть перечень мебельных фабрик, продукция которых представлена в мебельном магазине. Известно, мебель каких фабрик приобреталась n покупателями. Определить для каждой фабрики, мебель каких из них приобреталась всеми покупателями, каких — некоторыми из покупателей, и каких — никем из покупателей.
8	Есть перечень названий книг. Определить для каждой книги, какие из них прочли все из n читателей, какие — некоторые из читателей, и какие — никто из читателей.
9	Есть перечень дискотек города. Студенты группы любят посещать дискотеки. Известно для каждого студента, в каких дискотеках он побывал. Определить: <ul style="list-style-type: none"> • в какие дискотеки из перечня ходили все студенты группы; • в какие дискотеки из перечня ходили некоторые студенты группы; • в какие дискотеки из перечня не ходил ни один из студентов группы?
10	Есть перечень факультативов в учебном заведении. Студенты группы записались на некоторые из этих факультативов. Известно для каждого студента, на какие факультативы он записан. Определить: <ul style="list-style-type: none"> • на какие факультативы из перечня ходят все студенты группы; • на какие факультативы из перечня ходит хотя бы один студент группы; • на какие факультативы из перечня не ходит ни один из студентов группы?
11	Есть перечень торгово-развлекательных центров (ТРЦ) города. Студенты группы любят посещать ТРЦ. Известно для каждого студента, в каких ТРЦ он побывал. Определить: <ul style="list-style-type: none"> • в какие ТРЦ из перечня ходили все студенты группы; • в какие ТРЦ из перечня ходили некоторые студенты группы; • в какие ТРЦ из перечня не ходил ни один из студентов группы?
12	Есть перечень названий шоколада. Определить для каждого наименования шоколада, какие из них нравятся всем n сладкоежкам, какие — некоторым из сладкоежек, и какие — никому из сладкоежек.

13	Есть перечень названий телевизионных шоу. Определить для каждого наименования шоу, какие из них нравятся всем п телезрителям, какие — некоторым из телезрителей, и какие — никому из телезрителей.
14	Есть перечень иностранных языков. Работники фирмы могут знать некоторые из них. Для каждого работника известно, какие языки он знает. Определить для каждого языка, какие из них знает каждый из работников, какие — хотя бы один из работников, и какие — никто из работников.
15	В N сельскохозяйственных кооперативах выращивают некоторые сельскохозяйственные культуры из имеющегося перечня. Определить культуры: <ul style="list-style-type: none"> • возделываемые во всех сельскохозяйственных кооперативах; • возделываемые только в некоторых сельскохозяйственных кооперативах; • возделываемых ровно в одном сельскохозяйственном кооперативе.
16	Есть перечень политических партий в стране. Весь электорат (голосующих граждан) можно условно разделить на n групп. Проходит голосование на очередных выборах. Определить: <ul style="list-style-type: none"> • за какие партии проголосовали граждане из каждой группы; • за какие партии проголосовали граждане только из одной группы; • за какие партии не проголосовали?

Задание 4. HashSet

1	Файл содержит текст на русском языке. Напечатать в алфавитном порядке все звонкие согласные буквы, которые входят хотя бы в одно слово.
2	Файл содержит текст на русском языке. Напечатать в алфавитном порядке все глухие согласные буквы, которые не входят хотя бы в одно слово.
3	Файл содержит текст на русском языке. Напечатать в алфавитном порядке все согласные буквы, которые входят ровно в одно слово.
4	Файл содержит текст на русском языке. Напечатать в алфавитном порядке все глухие согласные буквы, которые не входят ровно в одно слово.
5	Файл содержит текст на русском языке. Напечатать в алфавитном порядке все звонкие согласные буквы, которые входят более чем в одно слово.
6	Файл содержит текст на русском языке. Напечатать в алфавитном порядке все гласные буквы, которые не входят более чем в одно слово.
7	Файл содержит текст на русском языке. Напечатать в алфавитном порядке все глухие согласные буквы, которые входят в каждое нечетное слово и не входят хотя бы одно четное слово.
8	Файл содержит текст на русском языке. Напечатать те символы, которых нет в первом слове, но они присутствуют в каждом из других.
9	Файл содержит текст на русском языке. Напечатать в алфавитном порядке символы, которые встречаются хотя бы однажды в словах с чётными номерами (нумерацию вести с 1).
10	Файл содержит текст на русском языке. Напечатать в алфавитном порядке символы, которые встречаются хотя бы однажды в словах с нечётными номерами (нумерацию вести с 1).
11	Файл содержит текст на русском языке. Сколько разных букв встречается в тексте?
12	Файл содержит текст на русском языке. Сколько букв русского алфавита не встречается в этом тексте?
13	Файл содержит текст на русском языке. Какие цифры встречаются в тексте?
14	Файл содержит текст на русском языке. С каких букв начинаются слова?
15	Файл содержит текст на русском языке. Какие символы есть в каждом слове?
16	Файл содержит текст на русском языке. Какие символы встречаются в одном и только в одном слове?

Задание 5. Dictionary/SortedList

1	<p>На вход программы подаются фамилии и имена учеников. Известно, что общее количество учеников не превосходит 100. В первой строке вводится количество учеников, принимавших участие в соревнованиях, N. Далее следуют N строк, имеющих следующий формат:</p> <p><Фамилия><Имя></p> <p>Здесь <Фамилия> – строка, состоящая не более чем из 20 символов; <Имя>– строка, состоящая не более чем из 15 символов. При этом <Фамилия> и <Имя> разделены одним пробелом. Примеры входных строк:</p> <p>Иванова Мария Петров Сергей</p> <p>Требуется написать программу, которая формирует и печатает уникальный логин для каждого ученика по следующему правилу: если фамилия встречается первый раз, то логин – это данная фамилия, если фамилия встречается второй раз, то логин – это фамилия, в конец которой приписывается число 2 и т.д. Например, для входной последовательности:</p> <p>Иванова Мария Петров Сергей Бойцова Екатерина Петров Иван Иванова Наташа</p> <p>будут сформированы следующие логины:</p> <p>Иванова Петров Бойцова Петров2 Иванова2</p>
2	<p>На городской олимпиаде по информатике участникам было предложено выполнить 3 задания, каждое из которых оценивалось по 25-балльной шкале. Известно, что общее количество участников первого тура олимпиады не превосходит 250 человек. На вход программы подаются сведения о результатах олимпиады. В первой строке вводится количество участников N. Далее следуют N строк, имеющих следующий формат:</p> <p><Фамилия><Имя><Баллы></p> <p>Здесь<Фамилия>– строка, состоящая не более чем из 20 символов;<Имя>– строка, состоящая не более чем из 15 символов;<Баллы>– строка, содержащая три целых числа, разделенных пробелом, соответствующих баллам, полученным участником за каждое задание первого тура. При этом <Фамилия> и <Имя>, <Имя> и <Баллы> разделены одним пробелом. Примеры входных строк:</p> <p>Петрова Ольга 25 18 16 Калининченко Иван 14 19 15</p> <p>Напишите программу, которая будет выводить на экран фамилию и имя участника, набравшего максимальное количество баллов. Если среди остальных участников есть ученики, набравшие такое же количество баллов, то их фамилии и имена также следует вывести. При этом имена и фамилии можно выводить в произвольном порядке.</p>

3	<p>На вход программы подаются сведения о результатах соревнований по школьному многоборью. Многоборье состоит из соревнований по четырем видам спорта, участие в каждом из которых оценивается баллами от 0 до 10 (0 баллов получает ученик, не принимавший участия в соревнованиях по данному виду спорта). Победители определяются по наибольшей сумме набранных баллов. Известно, что общее количество участников соревнований не превосходит 100.</p> <p>В первой строке вводится количество учеников, принимавших участие в соревнованиях, N. Далее следуют N строк, имеющих следующий формат:</p> <p style="text-align: center;"><Фамилия><Имя><Баллы></p> <p>Здесь <Фамилия>– строка, состоящая не более чем из 20 символов; <Имя>– строка, состоящая не более чем из 15 символов; <Баллы> - строка, содержащая четыре целых числа, разделенных пробелом, соответствующих баллам, полученным на соревнованиях по каждому из четырех видов спорта. При этом <Фамилия> и <Имя>, <Имя> и <Баллы> разделены одним пробелом. Примеры входных строк:</p> <p style="text-align: center;">Иванова Мария 5 8 6 3 Петров Сергей 9 9 5 7</p> <p>Напишите программу, которая будет выводить на экран фамилии и имена трех лучших участников многоборья. Если среди остальных участников есть ученики, набравшие то же количество баллов, что и один из трех лучших, то их фамилии и имена также следует вывести. При этом имена и фамилии можно выводить в произвольном порядке.</p>
4	<p>В некотором вузе абитуриенты проходят предварительное тестирование, по результатам которого могут быть допущены к сдаче вступительных экзаменов в первом потоке. Тестирование проводится по двум предметам, по каждому предмету абитуриент может набрать от 0 до 100 баллов. При этом к сдаче экзаменов в первом потоке допускаются абитуриенты, набравшие по результатам тестирования не менее 30 баллов по каждому из двух предметов. На вход программы подаются сведения о результатах предварительного тестирования. Известно, что общее количество участников тестирования не превосходит 500.</p> <p>В первой строке вводится количество абитуриентов, принимавших участие в тестировании, N. Далее следуют N строк, имеющих следующий формат:</p> <p style="text-align: center;"><Фамилия><Имя><Баллы></p> <p>Здесь <Фамилия>– строка, состоящая не более чем из 20 символов; <Имя>– строка, состоящая не более чем из 15 символов; <Баллы>– строка, содержащая два целых числа, разделенных пробелом, соответствующих баллам, полученным на тестировании по каждому из двух предметов. При этом <Фамилия> и <Имя>, <Имя> и <Баллы> разделены одним пробелом. Примеры входных строк:</p> <p style="text-align: center;">Ветров Роман 68 59 Анисимова Екатерина 64 88</p> <p>Напишите программу, которая будет выводить на экран фамилии и имена абитуриентов, потерпевших неудачу, то есть не допущенных к сдаче экзаменов в первом потоке. При этом фамилии должны выводиться в алфавитном порядке.</p>
5	<p>На вход программе подаются сведения о телефонах всех сотрудников некоторого учреждения. В первой строке сообщается количество сотрудников N, каждая из следующих N строк имеет следующий формат:</p> <p style="text-align: center;"><Фамилия><Инициалы><телефон></p> <p>где <Фамилия> – строка, состоящая не более чем из 20 символов, <Инициалы> - строка, состоящая не более чем из 4-х символов (буква, точка, буква, точка), <телефон> – семизначный номер, 3-я и 4, я, а также 5-я и 6-я цифры которого разделены символом «-». <Фамилия> и <Инициалы>, а также <Инициалы> и <телефон> разделены одним пробелом. Пример входной строки:</p> <p style="text-align: center;">Иванов П.С. 555-66-77</p> <p>Сотрудники одного подразделения имеют один и тот же номер телефона. Номера телефонов в учреждении отличаются только двумя последними цифрами. Требуется написать эффективную программу, которая будет выводить на экран информацию, сколько в среднем сотрудников работает в одном подразделении данного учреждения.</p>

6	<p>На вход программе сначала подается число участников олимпиады N. В каждой из следующих N строк находится результат одного из участников олимпиады в следующем формате:</p> <p><Фамилия><Имя><класс><баллы></p> <p>где <Фамилия>– символьная строка (не более 20 символов), <Имя>– символьная строка (не более 15 символов), <класс>– число от 7 до 11, <баллы>– целое число набранных участником баллов. <Фамилия> и <Имя>, <Имя> и <класс>, а также <класс> и <баллы> разделены одним пробелом. Пример входной строки:</p> <p>Семенов Егор 11 225</p> <p>Победителем олимпиады становится участник, набравший наибольшее количество баллов, при условии, что он набрал более 200 баллов. Если такое количество баллов набрали несколько участников, то все они признаются победителями при выполнении условия, что их доля не превышает 20% от общего числа участников.</p> <p>Победителем олимпиады не признается никто, если нет участников, набравших больше 200 баллов, или больше 20% от общего числа участников набрали одинаковый наибольший балл.</p> <p>Напишите эффективную по времени работы и по используемой памяти программу, которая будет определять фамилию и имя лучшего участника, не ставшего победителем олимпиады. Если таких участников несколько, т.е. если следующий за баллом победителей один и тот же балл набрали несколько человек, или, если победителей нет, а лучших участников несколько (в этом случае именно они являются искомыми), то выдается только количество искомых участников. Гарантируется, что искомые участники (участник) имеются.</p> <p>Программа должна выводить через пробел фамилию и имя искомого участника или их количество. Пример выходных данных (один искомый участник):</p> <p>Семенов Егор</p> <p>Второй вариант выходных данных (несколько искомых участников):</p> <p>12</p>
7	<p>В молочных магазинах города X продается сметана с жирностью 15, 20 и 25 процентов. В городе X был проведен мониторинг цен на сметану. Напишите эффективную по времени работы и по используемой памяти программу, которая будет определять для каждого вида сметаны, сколько магазинов продают ее дешевле всего. На вход программе сначала подается число магазинов N. В каждой из следующих N строк находится информация в следующем формате:</p> <p><Фирма><Улица><Жирность><Цена></p> <p>где <Фирма> – строка, состоящая не более, чем из 20 символов без пробелов, <Улица> – строка, состоящая не более, чем из 20 символов без пробелов, <Жирность> – одно из чисел – 15, 20 или 25, <Цена> – целое число в диапазоне от 2000 до 5000, обозначающее стоимость одного литра сметаны в копейках. <Фирма> и <Улица>, <Улица> и <Жирность>, а также <Жирность> и <Цена> разделены ровно одним пробелом. Пример входной строки:</p> <p>Перекресток Короленко 25 3200</p> <p>Программа должна выводить через пробел 3 числа – количество магазинов, продающих дешевле всего сметану с жирностью 15, 20 и 25 процентов. Если какой-то вид сметаны нигде не продавался, то следует вывести 0.</p> <p>Пример выходных данных:</p> <p>12 10 0</p>

8	<p>Школьная олимпиада по информатике проводилась для учеников 7-11-х классов, участвующих в общем конкурсе. Каждый участник олимпиады мог набрать от 0 до 70 баллов. Для определения призеров олимпиады сначала отбираются 25% участников, показавших лучшие результаты. Если у последнего участника, входящего в 25%, оказывается такое же количество баллов, как и у следующих за ним в итоговой таблице, все они считаются призерами только тогда, когда набранные ими баллы больше половины максимально возможных; иначе все они не считаются призерами.</p> <p>Напишите эффективную по времени работы и по используемой памяти программу, которая по результатам олимпиады будет определять минимальный балл призера олимпиады, и количество призеров было в каждой параллели (среди 7-х, 8-х, 9-х, 10-х и 11-х классов отдельно). Гарантируется, что хотя бы одного призера по указанным правилам определить можно.</p> <p>На вход программе сначала подается число участников олимпиады N. В каждой из следующих N строк находится результат одного из участников олимпиады в следующем формате:</p> <p style="text-align: center;"><Фамилия><Имя><класс><баллы></p> <p>где <Фамилия> – строка, состоящая не более, чем из 30 символов, <Имя> – строка, состоящая не более, чем из 15 символов, <класс> – число от 7 до 11, <баллы> – целое число от 0 до 70 набранных участником баллов. <Фамилия> и <Имя>, <Имя> и <класс>, а также <класс> и <баллы> разделены одним пробелом. Пример входной строки:</p> <p style="text-align: center;">Семенов Сидор 11 66</p> <p>Программа должна выводить в первой строке минимальный балл призера, а в следующей – число призеров по всем параллелям отдельно.</p> <p>Пример выходных данных:</p> <p style="text-align: center;">63 1 5 8 12 22</p>
9	<p>В некотором вузе абитуриенты проходили предварительное тестирование, по результатам которого они могут быть допущены к сдаче вступительных экзаменов в первом потоке. Тестирование проводится по трём предметам, по каждому предмету абитуриент может набрать от 0 до 100 баллов. При этом к сдаче экзаменов в первом потоке допускаются абитуриенты, набравшие по результатам тестирования не менее 30 баллов по каждому из трёх предметов, причём сумма баллов должна быть не менее 140. На вход программы подаются сведения о результатах предварительного тестирования. Известно, что общее количество участников тестирования не превосходит 500.</p> <p>В первой строке вводится количество абитуриентов, принимавших участие в тестировании, N. Далее следуют N строк, имеющих следующий формат:</p> <p style="text-align: center;"><Фамилия><Имя><Баллы></p> <p>Здесь <Фамилия> – строка, состоящая не более чем из 20 символов; <Имя> – строка, состоящая не более чем из 15 символов, <Баллы> – строка, содержащая два целых числа, разделенных пробелом – баллы, полученные на тестировании по каждому из трёх предметов. При этом <Фамилия> и <Имя>, <Имя> и <Баллы> разделены одним пробелом. Пример входной строки:</p> <p style="text-align: center;">Романов Вельямин 48 39 55</p> <p>Напишите программу, которая будет выводить на экран фамилии и имена абитуриентов, допущенных к сдаче экзаменов в первом потоке. При этом фамилии должны выводиться в алфавитном порядке.</p>
10	<p>На автозаправочных станциях (АЗС) продается бензин с маркировкой 92, 95 и 98. В городе N был проведен мониторинг цены бензина на различных АЗС. Напишите программу, которая будет определять для каждого вида бензина, сколько АЗС продают его дешевле всего.</p> <p>На вход программе в первой строке подается число данных N о стоимости бензина. В каждой из последующих N строк находится информация в следующем формате:</p> <p style="text-align: center;"><Компания><Улица><Марка><Цена></p> <p>где <Компания> – строка, состоящая не более, чем из 20 символов без пробелов, <Улица> – строка, состоящая не более, чем из 20 символов без пробелов, <Марка> – одно из чисел – 92, 95 или 98, <Цена> – целое число в диапазоне от 1000 до 3000, обозначающее стоимость одного литра бензина в копейках.</p> <p><Компания> и <Улица>, <Улица> и <Марка>, а также <Марка> и <Цена> разделены ровно одним пробелом. Пример входной строки:</p> <p style="text-align: center;">Синойл Цветочная 95 2250</p> <p>Программа должна выводить через пробел 3 числа – количество АЗС, продающих дешевле всего 92-й, 95-й и 98-й бензин соответственно. Если бензин какой-то марки нигде не продавался, то следует вывести 0.</p> <p>Пример выходных данных:</p> <p style="text-align: center;">12 1 0</p>

11	<p>В соревнованиях по многоборью (из М видов спорта) участвуют N спортсменов (N< 1000) .На вход программе в первой строке подается число спортсменов N, во второй – число видов спорта М. В каждой из последующих N строк находится информация в следующем формате:</p> <p><Фамилия><Имя><Баллы></p> <p>где <Фамилия> – строка, состоящая не более, чем из 20 символов без пробелов, <Имя> – строка, состоящая не более, чем из 12 символов без пробелов, <Баллы> – Мцелых чисел, обозначающие количество баллов, набранных спортсменом в каждом из видов многоборья.</p> <p><Фамилия> и <Имя>, <Имя> и <Баллы>, а также отдельные числа в поле <Баллы>разделены ровно одним пробелом. Пример входных строк:</p> <p>3 4 Иванов Сергей 100 30 78 13 Петров Антон 90 16 98 14 Сидоров Юрий 100 70 30 21</p> <p>Программа должна выводить результирующую таблицу, содержащую список спортсменов, отсортированный по убыванию суммы баллов, набранные суммы и занятые места.</p> <p>В данном случае программа должна вывести</p> <p>Иванов Сергей 221 1 Сидоров Юрий 221 1 Петров Антон 218 2</p>
12	<p>На вход программе подаются сведения о пассажирах, сдавших свой багаж в камеру хранения. В первой строке задано текущее время: через двоеточие два целых числа, соответствующие часам (от 00 до 21, ровно 2 символа) и минутам (от 00 до 59, ровно 2 символа). Во второй строке задается количество пассажиров N, которое не меньше 10, но не превосходит 1000. В каждой из последующих N строк находится информация о пассажирах в следующем формате:</p> <p><Фамилия><Время освобождения ячейки></p> <p>где <Фамилия> – строка, состоящая не более, чем из 20 символов без пробелов, <Время освобождения ячейки> – через двоеточие два целых числа, соответствующие часам (от 00 до 21, ровно 2 символа) и минутам (от 00 до 59, ровно 2 символа). <Фамилия> и <Время освобождения ячейки> разделены ровно одним пробелом. Пример входных строк:</p> <p>10:00 3 Иванов 12:00 Петров 10:12 Сидоров 12:12</p> <p>Программа должна выводить список пассажиров, которые в ближайшие 2 часа должны освободить ячейки. Список должен быть отсортирован в хронологическом порядке освобождения ячеек. В данном случае программа должна вывести</p> <p>Петров Иванов</p>
13	<p>Имеется список людей с указанием их фамилии, имени и даты рождения. Напишите программу, которая будет определять самого старшего человека из этого списка и выводить его фамилию и имя, а если имеется несколько самых старших людей с одинаковой датой рождения, то определять их количество.</p> <p>На вход программе в первой строке подается количество людей в списке N. В каждой из последующих N строк находится информация в следующем формате:</p> <p><Фамилия><Имя><Дата рождения></p> <p>где <Фамилия> – строка, состоящая не более, чем из 20 символов без пробелов, <Имя>– строка, состоящая не более, чем из 20 символов без пробелов, <Дата рождения> – строка, имеющая вид ДД.ММ.ГГГГ, где ДД – двузначное число от 01 до 31, ММ – двузначное число от 01 до 12, ГГГГ – четырехзначное число от 1800 до 2100.</p> <p>Пример входной строки:</p> <p>Иванов Сергей 27.03.1993</p> <p>Программа должна вывести фамилию и имя самого старшего человека в списке.</p> <p>Пример выходных данных:</p> <p>Иванов Сергей</p> <p>Если таких людей, несколько, то программа должна вывести их количество. Пример вывода в этом случае:</p> <p>3</p>

14	<p>Имеется список учеников разных школ, сдававших экзамен по информатике, с указанием их фамилии, имени, школы и набранного балла. Напишите программу, которая будет определять двух учеников школы № 50, которые лучше всех сдали информатику, и выводить на экран их фамилии и имена.</p> <p>Если наибольший балл набрали более двух человек, нужно вывести только их количество. Если наибольший балл набрал один человек, а следующий балл набрало несколько человек, нужно вывести только фамилию и имя лучшего. Известно, что информатику сдавали не менее 5 учеников школы № 50.</p> <p>На вход программе в первой строке подается количество учеников списке N. В каждой из последующих N строк находится информация в следующем формате:</p> <p style="text-align: center;"><Фамилия><Имя><Школа><Балл></p> <p>где <Фамилия> – строка, состоящая не более, чем из 20 символов без пробелов, <Имя>– строка, состоящая не более, чем из 20 символов без пробелов, <Школа> – целое число от 1 до 99, <Балл> – целое число от 1 до 100.</p> <p>Пример входной строки:</p> <p style="text-align: center;">Иванов Сергей 50 87</p> <p>Пример выходных данных, когда найдено два лучших:</p> <p style="text-align: center;">Иванов Сергей Сергеев Иван</p> <p>Если больше двух учеников набрали высший балл, то программа должна вывести их количество.</p> <p>Пример вывода в этом случае:</p> <p style="text-align: center;">8</p> <p>Если высший балл набрал один человек, а следующий балл набрало несколько человек, то программа должна вывести только фамилию и имя лучшего. Пример вывода в этом случае:</p> <p style="text-align: center;">Иванов Сергей</p>
15	<p>Имеется список учеников разных школ, сдававших экзамен по информатике, с указанием их фамилии, имени, школы и набранного балла. Напишите программу, которая будет определять номера школ, в которых средний балл выше, чем средний по району. Если такая школа одна, нужно вывести и средний балл (в следующей строчке). Известно, что информатику сдавали не менее 5 учеников. Кроме того, школ с некоторыми номерами не существует.</p> <p>На вход программе в первой строке подается количество учеников списке N. В каждой из последующих N строк находится информация в следующем формате:</p> <p style="text-align: center;"><Фамилия><Имя><Школа><Балл></p> <p>где <Фамилия> – строка, состоящая не более, чем из 20 символов без пробелов, <Имя>– строка, состоящая не более, чем из 20 символов без пробелов, <Школа> – целое число от 1 до 99, <Балл> – целое число от 1 до 100.</p> <p>Пример входной строки:</p> <p style="text-align: center;">Иванов Сергей 50 87</p> <p>Пример выходных данных, когда найдено три школы:</p> <p style="text-align: center;">50 87 23</p> <p>Пример вывода в том случае, когда найдена одна школа:</p> <p style="text-align: center;">18</p> <p style="text-align: center;">Средний балл = 85</p>

16	<p>После единых выпускных экзаменов по информатике в район пришла информация о том, какой ученик какой школы сколько баллов набрал. По положению об экзамене каждый район сам определяет, за какой балл нужно поставить какую оценку.</p> <p>Районный методист решила, что оценку «отлично» должны получить 20% участников (целое число, с отбрасыванием дробной части). Для этого она должна определить, какой балл должен был набрать ученик, чтобы получить «отлично». Если невозможно определить такой балл, чтобы «отлично» получили ровно 20% участников, «отлично» должно получить меньше участников, чем 20%. Если таких участников не окажется (наибольший балл набрали больше 20% участников) — эти и только эти ученики должны получить «отлично».</p> <p>Напишите программу, которая должна вывести на экран наименьший балл, который набрали участники, получившие «отлично». Известно, что информатику сдавало больше 5-ти учеников. Также известно, что есть такое количество баллов, которое не получил ни один участник.</p> <p>На вход программе сначала подаётся число учеников, сдававших экзамен. В каждой из следующих N строк находится информация об учениках в формате:</p> <p style="text-align: center;"><Фамилия><Имя><Номер школы><Количество баллов></p> <p>где <Фамилия> — строка, состоящая не более чем из 30 символов без пробелов, <Имя> — строка, состоящая не более, чем из 20 символов без пробелов, <Номер школы> — целое число в диапазоне от 1 до 99, <Количество баллов> — целое число в диапазоне от 1 до 100. Эти данные записаны через пробел, причём ровно один между каждой парой (то есть, всего по три пробела в каждой строке).</p> <p>Пример входной строки:</p> <p style="text-align: center;">Иванов Иван 50 87</p> <p>Пример выходных данных:</p> <p style="text-align: center;">78</p>
----	--

Задание 6. Перегрузка операций

№	Название класса	Поля	Методы
1	Time	byte hours byte minutes	Добавление произвольного количества минут (uint) к объекту типа Time. Результат должен быть типа Time.
2	Money	uint rubles, byte kopeks	Добавление произвольного количества копеек (uint) к объекту типа Money. Результат должен быть типа Money.
3	Point	double x, double y (координаты точки)	Вычислить расстояние от одной точки до другой. Результат должен быть типа double.
4	RightTriangle	double a, double b (длины катетов)	Вычислить площадь треугольника. Результат должен быть типа double.
5	LineSegment	double x, double y (координаты начала и конца отрезка на координатной прямой)	Определить, попадает ли заданное число в отрезок. Результат должен быть типа bool.
6	QuadraticEquation	double a, double b, double c (коэффициенты)	Вычисление корней квадратного уравнения. Результат должен быть массивом величин типа double (в нем от 0 до 2-х элементов, в зависимости от количества корней).
7	Time	byte hours byte minutes	Вычитание переменной типа Time (учесть, что возможен переход в предыдущие сутки). Результат должен быть типа Time.
8	Money	uint rubles, byte kopeks	Вычитание переменной типа Money (учесть, что денежная величина не может быть меньше 0). Результат должен быть типа Money.
9	Point	double x, double y (координаты точки)	Вычислить расстояние от точки до начала координат. Результат должен быть типа double.

10	Triangle	double a, double b, double c (длины сторон треугольника)	Определить, существует ли треугольник с такими длинами сторон. Результат должен быть типа bool.
11	LineSegment	double x, double y (координаты начала и конца отрезка на координатной прямой)	Определить пересекаются ли заданные отрезки. Результат должен быть типа bool.
12	Time	byte hours byte minutes	Вычитание минут (uint) из объекта типа Time (учесть, что возможен переход в какие-либо предыдущие сутки). Результат должен быть типа Time.
13	Money	uint rubles, byte kopeks	Вычитание копеек (uint) из объекта типа Money (учесть, что денежная величина не может быть меньше 0). Результат должен быть типа Money.
14	Time	byte hours byte minutes	Вычитание времени (величины типа Time) из объекта типа Time (учесть, что возможен переход в предыдущие сутки). Результат должен быть типа Time.
15	Money	uint rubles, byte kopeks	Сравнение двух объектов типа Money. Результат должен быть типа bool.
16	LineSegment	double x, double y (координаты начала и конца отрезка на координатной прямой)	Определить пересечение заданных отрезков. Результат должен быть типа LineSegment (он равен null, если отрезки не пересекаются).

Задание 7. Перегрузка операций

№	Название класса	Методы
1	Time	<p><u>Унарные операции:</u> ++ добавление минуты к объекту типа Time -- вычитание минуты из объекта типа Time</p> <p><u>Операции приведения типа:</u> byte (явная) – результатом является количество часов (минуты отбрасываются); bool (неявная) – результатом является true, если часы и минуты не равны нулю, и false в противном случае.</p> <p><u>Бинарные операции:</u> + Time t, беззнаковое целое число (минуты) (лево- и правосторонние операции). Результат должен быть типа Time. – Time t, беззнаковое целое число (минуты) (лево- и правосторонние операции). Результат должен быть типа Time.</p>
2	Money	<p><u>Унарные операции:</u> ++ добавление копейки к объекту типа Money -- вычитание копейки из объекта типа Money</p> <p><u>Операции приведения типа:</u> uint (явная) – результатом является количество рублей (копейки отбрасываются); double (неявная) – результатом является копейки, переведенные в рубли, целые рубли отбрасываются, результат <1.</p> <p><u>Бинарные операции:</u> + Money m, беззнаковое целое число (лево- и правосторонние операции). Результат должен быть типа Money. – Money m, беззнаковое целое число (лево- и правосторонние операции). Результат должен быть типа Money.</p>

3	Point	<p><u>Унарные операции:</u> ++ увеличить координаты х на 1, -- уменьшение координаты х на 1.</p> <p><u>Операции приведения типа:</u> int (явная) – результатом является целая часть координаты х; double (неявная) – результатом является координата у.</p> <p><u>Бинарные операции:</u> + Point p – вычисляется расстояние до точки p. Результатом должно быть вещественное число. + целое число (лево- и правосторонние операции). Увеличивается координата x, результатом является объект типа Point.</p>
4	RightTriangle	<p><u>Унарные операции:</u> ++ увеличивает стороны треугольника в 2 раза (чтобы вновь получить катеты прямоугольного треугольника); -- уменьшает стороны треугольника в 2 раза (чтобы вновь получить катеты прямоугольного треугольника);</p> <p><u>Операции приведения типа:</u> double (явная) – результатом является площадь треугольника, если треугольник существует и отрицательное число в противном случае; bool (неявная) – результатом является true, если треугольник с такими длинами сторон существует и false в противном случае.</p> <p><u>Бинарные операции:</u> <= Trianglet1, Trianglet2 – сравнивает площади треугольников; >= Trianglet1, Trianglet2 – сравнивает площади треугольников.</p>
5	LineSegment	<p><u>Унарные операции:</u> ! - вычислить длину отрезка, результат должен быть типа double; ++ увеличить координаты границ отрезка на 1.</p> <p><u>Операции приведения типа:</u> int (явная) – результатом является целая часть координаты х; double (неявная) – результатом является координата у.</p> <p><u>Бинарные операции:</u> +int d, целое число (лево- и правосторонняя операция, координаты границ увеличиваются на число d); < LineSegment, целое число – результат равен true, если целое число попадает в заданный отрезок и false – в противном случае.</p>
6	QuadraticEquation	<p><u>Унарные операции:</u> ++ увеличивает коэффициенты уравнения на 1; -- уменьшает коэффициенты уравнения на 1.</p> <p><u>Операции приведения типа:</u> double (неявная) – результатом является дискриминант уравнения. bool (явная) – результатом является true, если корни существуют и false в противном случае;</p> <p><u>Бинарные операции</u> == QuadraticEquation1, QuadraticEquation2 — уравнения равны, если равны их коэффициенты; != QuadraticEquation1, QuadraticEquation2 — уравнения не равны, если не равны их коэффициенты.</p>

7	Time	<p><u>Унарные операции:</u> ++ добавление минуты к объекту типа Time. -- вычитание минуты из объекта типа Time</p> <p><u>Операции приведения типа:</u> int (неявная) – результатом является количество минут (время переводится в минуты); bool (явная) – результатом является true, если часы и минуты не равны нулю, и false в противном случае.</p> <p><u>Бинарные операции:</u> < Timet1, Timet2 — время переводится в минуты, результатом является true, если количество минут в левом операнде меньше, чем количество минут в правом операнде, и false–в противном случае. > Timet1, Timet2 — время переводится в минуты, результатом является true, если количество минут в левом операнде больше, чем количество минут в правом операнде, и false–в противном случае.</p>
8	Money	<p><u>Унарные операции:</u> ++ добавление копейки к объекту типа Money -- вычитание копейки из объекта типа Money</p> <p><u>Операции приведения типа:</u> uint (неявная) – результатом является количество рублей (копейки отбрасываются); double(явная) – результатом является копейки, переведенные в рубли, целые рубли отбрасываются, результат <1.</p> <p><u>Бинарные операции:</u> - Money m, беззнаковое целое число (лево- и правосторонние операции), результат должен быть типа Money. - Money m, Money m, результат должен быть типа Money. Результат не может быть отрицательным.</p>
9	Point	<p><u>Унарные операции:</u> --уменьшить координаты х и у на 1; - поменять координаты х и у местами.</p> <p><u>Операции приведения типа:</u> int (неявная) – результатом является целая часть координаты х; double(явная) – результатом является координата у.</p> <p><u>Бинарные операции:</u> – Point p, целое число (левосторонняя операция, уменьшается координата х); – целое число, Point p (правосторонняя операция, уменьшается координата у); – Point p – вычисляется расстояние до точки p, результатом должно быть вещественное число.</p>
10	Triangle	<p><u>Унарные операции:</u> – вычисляется площадь треугольника, результатом должно быть вещественное число.</p> <p><u>Операции приведения типа:</u> double (неявная) – результатом является периметр треугольника; bool (явная) – результатом является true, если треугольник с такими длинами сторон существует, и false в противном случае.</p> <p><u>Бинарные операции:</u> < - сравниваются площади треугольников; > - сравниваются площади треугольников.</p>

11	LineSegment	<p><u>Унарные операции:</u></p> <p>! вычислить длину диапазона, результатом должно быть вещественное число</p> <p>++ расширить отрезок на 1 вправо и влево</p> <p><u>Операции приведения типа:</u></p> <p>int (неявная) – результатом является целая часть координаты x;</p> <p>double(явная) – результатом является координата y.</p> <p><u>Бинарные операции:</u></p> <ul style="list-style-type: none"> - LineSegment целое число (левосторонняя операция, уменьшается координата x); - целое число LineSegment (правосторонняя операция, уменьшается координата y); < LineSegment d – результат равен true, если отрезки пересекаются, и false – в противном случае.
12	Time	<p><u>Унарные операции:</u></p> <ul style="list-style-type: none"> - обнуление часов и минут в объекте типа Time. -- обнуление минут в объекте типа Time <p><u>Операции приведения типа:</u></p> <p>byte (неявная) – результатом является количество часов (минуты отбрасываются);</p> <p>bool(явная) – результатом является true, если часы и минуты не равны нулю, и false в противном случае.</p> <p><u>Бинарные операции:</u></p> <p>== Timet1, Timet2 результатом является true, если значение часов и минут левого операнда равно соответствующим значениям правого операнда, и false – в противном случае.</p> <p>!= Timet1, Timet2 результатом является true, если значение часов и минут левого операнда не равно соответствующим значениям правого операнда, и false – в противном случае.</p>
13	Money	<p><u>Унарные операции:</u></p> <ul style="list-style-type: none"> -- вычитание копейки из объекта типа Money ++ добавление копейки к объекту типа Money <p><u>Операции приведения типа:</u></p> <p>uint (явная) результатом является количество рублей (копейки отбрасываются);</p> <p>bool(неявная) результатом является true, если денежная сумма не равна 0.</p> <p><u>Бинарные операции:</u></p> <ul style="list-style-type: none"> - Money m, беззнаковое целое число (лево- и правосторонние операции). Число обозначает копейки - Money m1, Money m2 вычитание денежных сумм
14	Time	<p><u>Унарные операции:</u></p> <ul style="list-style-type: none"> -- вычитание минуты из объекта типа Time <p><u>Операции приведения типа:</u></p> <p>byte (явная) – результатом является количество часов (минуты отбрасываются);</p> <p>bool (неявная) – результатом является true, если часы и минуты не равны нулю, и false в противном случае.</p> <p><u>Бинарные операции:</u></p> <ul style="list-style-type: none"> + Timet, беззнаковое целое число (лево- и правосторонние операции) – добавление минут к времени. + Timet1, Timet2 – сложить два времени.

15	Money	<p><u>Унарные операции:</u></p> <p>-- вычитание копейки из объекта типа Money</p> <p>++ добавление копейки к объекту типа Money</p> <p><u>Операции приведения типа:</u></p> <p>uint (неявная) результатом является количество рублей (копейки отбрасываются);</p> <p>double (явная) результатом является копейки, переведенные в рубли, целые рубли отбрасываются, результат <1.</p> <p><u>Бинарные операции:</u></p> <p>- Money m, целое число (лево- и правосторонние операции) вычитание копеек из денежной суммы</p>
16	LineSegment	<p><u>Унарные операции:</u></p> <p>! установить соответствующую координату в 0, чтобы получился правильный диапазон. Ответом должен быть объект LineSegment</p> <p><u>Операции приведения типа:</u></p> <p>int (неявная) – результатом является целая часть координаты y;</p> <p>double (явная) – результатом является координата x.</p> <p><u>Бинарные операции:</u></p> <p>+ LineSegment целое число (левосторонняя операция, увеличивается координата x, ответом должен быть правильный диапазон);</p> <p>+ целое число LineSegment (правосторонняя операция, увеличивается координата y, ответом должен быть правильный диапазон);</p> <p>> LineSegment d – результат равен true, если левый диапазон включает правый, и false – в противном случае.</p>