

Задача А. Мобильный интернет

Мобильный интернет прочно вошел в нашу жизнь. Операторы связи предлагают различные способы оплаты мобильного интернета и, зная свои потребности, можно выбрать наиболее дешевый из подходящих тарифов.

Рассмотрим следующие тарифные планы:

1. Единовременно каждый месяц платится 350Р за 3000 мегабайт. Также можно докупать дополнительные пакеты по 300 мегабайт за 30Р каждый, которые действуют до конца месяца.
2. 500 мегабайт в день за 29Р в сутки. За дни, в которые интернет не используется (скачано 0 мегабайт), плата не взимается.
3. Оплата за использованный трафик — 1,2Р за 1 мегабайт.
4. Безлимитный интернет на месяц за 790Р.
5. Лимитированный тариф — 16000 мегабайт на месяц за 590Р.

По известному количеству трафика в каждый из 31 дней одного месяца определите, сколько денег уйдет на оплату интернета при использовании каждого из тарифных планов или сообщите, что использование тарифа невозможно (недостаточно трафика).

В единственной строке входных данных содержится тридцать одно целое неотрицательное число — планируемый трафик по дням следующего месяца в мегабайтах.

Выведите пять чисел в отдельных строках — стоимость трафика за месяц при использовании соответствующего тарифа или -1 , если требуемое использование интернета недопустимо в рамках соответствующего тарифа (например, суммарный или суточный трафик превосходит ограничение тарифа).

Стоимость требуется вывести в формате *<рубли> <копейки>*.

Вам дано 2 файла:

a1.txt — первый тест к задаче. Оценка за тест: 30 баллов, баллы выставляются только в случае полностью правильного решения. Проверка осуществляется в режиме on-line (результат виден сразу).

a2.txt — второй тест к задаче. Оценка за тест: 70 баллов, за каждую неверно рассчитанную стоимость оценка уменьшается на 30 баллов, однако не может стать меньше нуля. Проверка осуществляется в режиме off-line (после окончания тура). Ответ принимается на проверку, если в нем содержится ровно 5 описаний потраченной суммы (пары рубли-копейки или число -1).

Пример

Входные данные	Результат
3001 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	380 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	-1
	3601 20
	790 0
	590 0

Задача В. Сливаемся!

Слияние двух упорядоченных последовательностей чисел в одну упорядоченную — основная идея сортировки слиянием. Эта сортировка работает быстро, а слияние двух упорядоченных последовательностей легко выполняется в том числе и человеком вручную.

В этой задаче по полученной в результате слияния неубывающей последовательности чисел вам предстоит восстановить две исходных неубывающих последовательности одинакового размера.

Некоторые числа в исходных последовательностях известны, а некоторые заменены знаком «?». Результат слияния известен полностью. Вам необходимо подставить на место знаков вопроса числа так, чтобы исходные последовательности были неубывающими и при слиянии образовывали заданную результирующую последовательность.

В первой входного файла строке содержится число N — количество элементов в каждой из исходных последовательностей. Во второй и третьей строках записано по N чисел и знаков вопроса — содержимое первой и второй последовательности соответственно. В четвертой строке записано $2 \times N$ чисел — результат слияния.

Выходной файл должен содержать 2 строки по N чисел в каждой: какой-нибудь из вариантов восстановления исходных последовательностей, которые при слиянии дадут тот же результат. Если в исходных данных в последовательности стояло число, то в выходных данных на том же месте должно стоять то же число.

В этой задаче оба теста проверяются **онлайн**.

Вам дано 2 файла:

b1.txt — первый (маленький) тест к задаче, $N = 5$. Оценка за тест: 30 баллов, баллы выставяются только в случае полностью правильного решения. Проверка осуществляется в режиме on-line (результат виден сразу).

b2.txt — второй (большой) тест к задаче, $N = 500$. Оценка за тест: 70 баллов, баллы выставяются только в случае полностью правильного решения. Проверка осуществляется в режиме on-line (результат виден сразу).

Пример

Входные данные	Результат
2	1 4
? 4	3 5
3 ?	
1 3 4 5	

Задача С. Голосование

На острове Мумба-Юмба вопросы традиционно решались тайным голосованием: каждый человек кидал в мешок белый или черный камешек (это соответствовало голосу «за» или «против» соответственно). Решение «за» принималось, если белых камешков было больше.

Но прогресс не стоит на месте: пришло время внедрить электронную систему голосования. На острове действует программа импортозамещения, поэтому систему голосования решено производить из собственных компонентов. Электронная промышленность Мумба-Юмбы не очень развита и может изготовить только простейшие логические элементы: «и», «или» и «не».

Вам необходимо построить две схемы, на вход которым поступают результаты голосования, а на выходе должно быть число 1, если больше половины проголосовали «за» и 0 — в противном случае.

В голосовании принимают участие N человек. Если человек голосует «за», то это обозначается единицей, если «против» — нулем.

В первой строке выведите общее число элементов схемы K (не более 5000), включая N входов и один элемент для вывода результата голосования. Входы в схеме имеют номера от 1 до N .

Каждая из следующих $K - N$ строк описывает логический элемент (они получают номера начиная с $N + 1$ подряд в порядке описания) и начинается с одного из символов "&", "|", "!", "#", обозначающих, соответственно «и», «или», «не» и выход схемы. После каждого из символов следует одно или два числа — номера элементов схемы (входов или описанных в других строках логических элементов), результат которых принимает на вход этот элемент. После «или» и «и» указываются два числа, после «не» и выходного элемента — одно. Выходной элемент в схеме должен быть ровно один.

В этой задаче нет входных файлов.

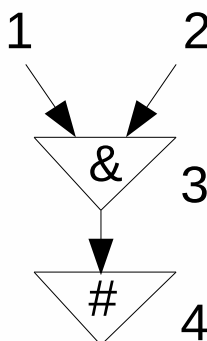
В первом (маленьком) тесте необходимо построить схему для подсчета результата голосования трех человек. Оценка за тест: 30 баллов, баллы выставляются только в случае, если схема выдает правильный результат для всевозможных результатов голосования. Проверка осуществляется в режиме on-line (результат виден сразу).

Во втором (большом) тесте необходимо построить схему для подсчета результата голосования 15 человек. Оценка за тест: 70 баллов, баллы выставляются только в случае, если схема выдает правильный результат для всевозможных результатов голосования. Проверка осуществляется в режиме off-line (после окончания тура). Ответ считается принятым на проверку, если в схеме не более 5000 элементов, ровно один выход, ни один из элементов не принимает ничего из выхода, а также соблюден формат описания операций.

Пример

Входные данные	Результат
2	4 & 1 2 # 3

Ответу соответствует схема:



Задача D. Линии связи

Оператор связи «Мумба-Юмба-Телеком» решил соединить все поселения острова оптоволоконными линиями связи. В каждом из поселений установлен роутер, в который может входить любое количество линий связи из других роутеров. При этом оптоволокно можно прокладывать по прямой и оно должно соединять все поселения, т.е. сигнал может пройти от любого поселения до любого другого, возможно, через промежуточные поселения.

Оператор составил карту страны, где все поселения представлены в виде точек с целыми координатами. Кроме того, на складе оператора нашлось K лишних роутеров, которые можно установить в любой целочисленной точке плоскости (два роутера не могут стоять в одной точке, дополнительный роутер нельзя устанавливать в поселении).

Определите, где поставить дополнительные роутеры, чтобы суммарная длина оптоволоконных линий была как можно меньше. Не обязательно устанавливать все роутеры.

В первой строке входного файла указывается количество поселений N и количество дополнительных роутеров на складе K . В следующих N строках содержатся по 2 целых числа от 0 до 10^4 — координаты соответствующего поселения.

В первой строке выходного файла укажите количество использованных дополнительных роутеров M , в следующих M строках укажите по паре целых чисел от 0 до 10^4 — координаты дополнительных роутеров.

Для вашего удобства вам предоставлены программы `wire1.exe` и `wire2.exe`, которые принимают на вход файлы `d1out.txt` и `d2out.txt`, формат которых описан выше, и записывают в файлы `d1ans.txt` и `d2ans.txt` минимально возможную суммарную длину оптоволоконных линий для вашей расстановки роутеров.

Вам дано 2 файла:

`d1.txt` — первый (маленький) тест к задаче, $N = 4, K = 2$. Максимальная оценка за тест: 30 баллов. Проверка осуществляется в режиме on-line (результат виден сразу).

`d2.txt` — второй (большой) тест к задаче, $N = 20, K = 10$. Максимальная оценка за тест: 70 баллов. Проверка осуществляется в режиме off-line (после окончания тура). Во время тура осуществляется проверка на корректность (два роутера не стоят в одном месте, координаты дополнительных роутеров удовлетворяют ограничениям, использовано не более K дополнительных роутеров).

Баллы за тест вычисляются по формуле

$$TestScore \times \frac{CleanAns - UserAns}{CleanAns - BestAns}$$

где $TestScore$ — максимальный балл за тест, $UserAns$ — минимально возможная суммарная длина линий в вашем решении, $CleanAns$ — минимально возможная суммарная длина линий в случае, если дополнительные роутеры не установлены, $BestAns$ — минимально возможная суммарная длина линий в лучшем решении (в маленьком тесте — в решении жюри, в большом тесте — лучшее среди решений участников). Баллы не могут стать отрицательными.

Пример

Входные данные	Результат
3 1	1
0 0	2 2
4 0	
2 5	

Задача Е. Покупка билетов

Команда города А, состоящая из школьников и их руководителей, собралась поехать на олимпиаду по информатике в город В поездом в плацкарте. К сожалению, результаты отборочного этапа олимпиады были опубликованы довольно поздно и часть мест в подходящем поезде уже была куплена другими пассажирами.

Организаторы поездки хотят купить билеты на свободные места так, чтобы максимизировать суммарную радость от поездки. Каждый человек получает единицу радости, если едет на верхней боковой полке, две единицы радости от нижней боковой полки, три единицы от верхней небоковой полки и четыре единицы от нижней небоковой полки. Кроме того, в команде города А все знакомы друг с другом и получают радость, если едут вместе, а не с чужими пассажирами, а именно человек дополнительно получает столько единиц радости, сколько других людей из команды города А, включая руководителей, едут с ним вместе в одном шестиместном отсеке. Кроме того, по правилам СЭС, в каждом вагоне со школьниками должен ехать хотя бы один руководитель. Радости для школьников и руководителей вычисляются одинаково.

В плацкартном вагоне 9 отсеков по 6 мест, нумерация мест показана на рисунке. «В» — верхние полки, «Н» — нижние, «ВБ» — верхние боковые полки, «НБ» — нижние боковые полки.

В	2	4	6	8	10	12	14	16	18	20	22	24	26	28	30	32	34	36
Н	1	3	5	7	9	11	13	15	17	19	21	23	25	27	29	31	33	35
ВБ	54		52		50		48		46		44		42		40		38	
НБ	53		51		49		47		45		43		41		39		37	

В первой строке входных данных указаны 4 числа: P — количество вагонов в поезде, N — количество людей в команде (школьников и руководителей вместе), K — количество руководителей и D — количество еще свободных мест в поезде. В следующих D строках входных данных перечислены места, билеты на которые еще не куплены. Каждое описание состоит из двух чисел — номера вагона и номера места в вагоне. Места перечисляются в порядке возрастания номеров вагонов, и в порядке возрастания номеров мест внутри одного вагона.

В выходном файле выведите N строк — описание мест, билеты на которые нужно купить. Каждое описание состоит из двух чисел: номера вагона и номера места. Порядок вывода может быть любой.

Баллы за решения будут начисляться прямо пропорционально качеству рассадки. Например, если суммарная радость в вашем решении равна 50, а в оптимальном решении — 100, то ваше решение получит половину баллов.

Вам дано 2 файла:

e1.txt — первый (маленький) тест к задаче, необходимо купить билеты для 15 человек. Оценка за тест: 30 баллов. Проверка осуществляется в режиме on-line (результат виден сразу).

e2.txt — второй (большой) тест к задаче, необходимо купить билеты для 80 человек. Оценка за тест: 70 баллов. Проверка осуществляется в режиме off-line (после окончания тура). Во время тура осуществляется проверка решения на корректность: никакие двое людей не едут на одном месте, указаны допустимые номера вагонов и мест, билеты куплены не более чем в K различных вагонов.

Пример

Входные данные	Результат
1 4 1 6	1 6
1 1	1 8
1 3	1 51
1 6	1 52
1 8	
1 51	
1 52	

В примере члены команды получают $3 + 3 + 2 + 1$ единиц радости от купленных им мест и 4×3 единицы радости от того, что все едут в одном отсеке.

Задача F. Сочинение

Экзамен по литературе у студентов Хогвартса будет проходить в формате сочинения. К огромному сожалению, большинство студентов не умеют грамотно излагать свои мысли, а также у них крайне ограниченный словарный запас. Но, благодаря новой системе оценивания сочинений, разработанной Министерством Магии, у них есть шанс не только не получить за него неудовлетворительную оценку, но и написать его лучше, чем круглая отличница Гермиона.

Эксперты из министерства убеждены, что кропотливая и вдумчивая проверка каждого сочинения занимает слишком много времени, поэтому они разработали более простые правила проверки и оценивания сочинений:

- Сочинение состоит из одного или более предложений. Каждое предложение должно начинаться с прописной буквы и заканчиваться точкой. Все буквы в предложении, кроме первой, должны быть строчными. Предложения должны разделяться пробелом.
- В каждом предложении должно быть хотя бы одно существительное.
- В каждом предложении должен быть ровно один глагол.
- Следующим словом после каждого прилагательного должно быть существительное.
- Количество слов в сочинении должно быть не меньше числа L и не больше числа R .
- В сочинении не должно быть двух одинаковых слов.

Для оценивания используется набор *ключевых* слов, составленный министерством: оценка за сочинение численно равна количеству ключевых слов в сочинении.

Но проводить экзамен слишком затратно для школы, и вы, будучи преподавателем *магического программирования, танцев и гадания на кофейной гуще* взялись определить наилучший результат, который может набрать каждый студент.

В первой строке входного файла содержится целое число T — количество студентов. Далее идут T описаний вариантов экзамена, полученных каждым из студентов, включающих в себя словарь известных студенту слов, некоторые из них помечены как ключевые.

В первой строке описания каждого варианта содержатся три целых числа N_i , V_i и A_i — количество известных студенту существительных, глаголов и прилагательных соответственно. Далее в отдельных строках идут описания N_i существительных, затем V_i глаголов и затем A_i прилагательных. Каждое описание состоит из самого слова, записанного строчными буквами латинского алфавита, и через пробел числа 1, если слово является *ключевым*, и 0 — в противном случае.

После описаний всех слов, известных студенту, в строке содержатся два числа L_i и R_i — нижняя и верхняя границы количества слов в сочинении для данного варианта.

Гарантируется, что все слова, известные любому студенту, различны. Для каждого варианта гарантируется наличие сочинения, удовлетворяющего условию.

Для каждого студента выведите в отдельной строке сочинение, получающее максимально возможную оценку. Если таких сочинений несколько, выведите любое из них.

Вам дано 2 файла:

f1.txt — первый (маленький) тест, оценка за тест: 30 баллов. Проверка осуществляется в режиме on-line (результат виден сразу). Баллы ставятся только при нахождении одного из возможных правильных ответов для всех вариантов сочинения.

f2.txt — второй (большой) тест, максимальная оценка за тест: 70 баллов. При нахождении правильного ответа для всех вариантов, начисляются все 70 баллов. В противном случае результат вычисляется по формуле:

$$\text{result} = \left\lfloor \frac{\text{passed}}{\text{variants}} \times 40 \right\rfloor,$$

где variants — количество вариантов во втором тесте, а passed — количество вариантов, для которых был найден правильный ответ.

Для того, чтобы ваш ответ на второй тест был принят на проверку, в нем должно быть variants строк. Проверка осуществляется в режиме off-line (после окончания тура).

Примеры

Входные данные	Результат
1 2 1 1 potter 1 herminy 1 lived 0 harry 1 2 3	Harry potter lived.
2 2 1 2 mimsy 1 borogoves 1 were 1 all 1 the 1 0 10 3 1 1 mome 1 outgrabe 1 and 0 raths 1 the 0 5 5	All mimsy were the borogoves. And the mome raths outgrabe.
1 5 2 1 tom 0 monsters 0 pets 0 hagrid 0 riddle 0 said 1 make 1 good 1 7 8	Monsters make good pets hagrid. Tom riddle said.

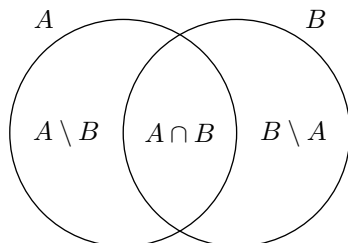
Пояснения к примерам (полужирным выделены *ключевые* слова):

- Harry potter lived.
- All mimsy were the borogoves.
- And the mome raths outgrabe.
- Monsters make good pets hagrid. Tom riddle said.

Задача Г. Диаграмма Эйлера-Венна

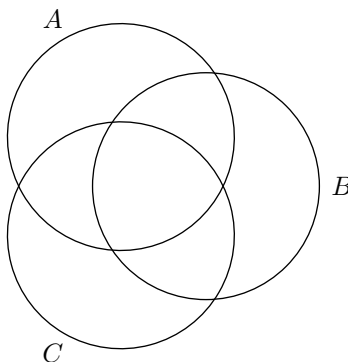
Диаграмма Эйлера-Венна — это наглядный способ изображения всевозможных пересечений некоторых n множеств. Рассмотрим несколько примеров.

Пусть $n = 2$. Хороший способ изобразить взаимосвязь двух множеств на плоскости — нарисовать их в виде двух пересекающихся кругов, а дальше поместить все интересующие нас элементы в подходящие части рисунка.



Например, если множество A состоит из всех мировых столиц, а множество B — из всех городов России, то Санкт-Петербург попадёт в правую область (российский город, но не столица), Берлин — в левую (столица Германии), в общую центральную часть попадёт только Москва, а, например, Сирмионе (небольшой городок в Италии) окажется снаружи обоих кругов, так как он не является ни российским городом, ни мировой столицей.

Аналогичным образом можно изобразить диаграмму для $n = 3$.



Нетрудно проверить, что любой объект всегда можно отнести в одну из восьми частей, на которые три нарисованных круга делят плоскость так, чтобы он оказался внутри/вне круга тогда и только тогда, когда он принадлежит/не принадлежит соответствующему множеству.

Более формально: диаграмма Эйлера-Венна для n множеств представляет собой n фигур A_1, A_2, \dots, A_n на плоскости, обладающих следующим свойством. Рассмотрим любое подмножество $\{i_1, i_2, \dots, i_k\}$ (возможно, пустое) чисел от 1 до n . Тогда на плоскости должна быть точка, которая лежит внутри фигур $A_{i_1}, A_{i_2}, \dots, A_{i_k}$, и которая лежит вне всех остальных фигур.

К сожалению, для больших значений n диаграмма Эйлера-Венна не будет иметь такой простой и наглядный вид, как показано выше. В качестве наглядного представления множеств требуется изобразить диаграмму как набор клетчатых фигур.

Вы должны изобразить n прямоугольников одинакового размера, в которых «нарисовать» каждую из фигур, обозначая клетки, лежащие внутри очередной фигуры, как ‘*’, а клетки, лежащие вне очередной фигуры, как ‘.’. Диаграмма будет строиться по этим прямоугольникам путём совмещения их углов и будет состоять из n образовавшихся фигур.

В первой строке запишите два числа h и w , обозначающие ширину и высоту каждого из прямоугольников. Размеры должны удовлетворять условию $(1 \leq h, w \leq 500)$.

Далее должна следовать строка, содержащая символ ”-” (минус).

Далее должно следовать n прямоугольников, в i -м из которых изображена очередная фигура A_i . Прямоугольник должен состоять из h строк по w символов, каждый из которых — либо ‘*’, либо ‘.’. Разделяйте прямоугольники минусами.

Все клетки очередной фигуры должны образовывать **связное** множество, т. е. из каждой клетки фигуры должен существовать путь в каждую, пролегающий только по клеткам фигуры, при этом перемещаться разрешается только в смежную по стороне клетку.

Ваша задача — изобразить диаграмму Эйлера-Венна для $n = 4$ в первой подзадаче (30 баллов) и для $n = 8$ во второй (70 баллов).

Примеры

Входные данные	Результат
3	<pre> 9 8 -**.... .****... .****... .****... ..**.... -**.. ...****. ...****. ...****. ...****. ...**.. -**.. ...****. ...****. ...****. ...**.. </pre>