

Задание № 1. Мудрый гоблин

В небольшой деревне, расположенной рядом с горой Каджаро, в которой издавна добывали каджамитовую руду, жил был старый гоблин Скуззакс.

Среди своих сородичей он был известен как талантливый механик и изобретатель. За всю свою жизнь ему удалось скопить огромное состояние. Свои деньги и драгоценности он не спешил нести в сберегательную кассу, а хранил все в огромном сундуке, который сам для себя же и смастерил. Много раз воры пытались открыть замок на этом сундуке, но так и не смогли.

Работа замка заключалась в следующем:

Замок состоит из четырех рядов по четыре кнопки. Нажав на все кнопки в определенном порядке, он отрывается. Чтобы легче было запоминать последовательность букв, Скуззакс сделал для себя подсказку - деревянную дощечку с четырьмя прорезями. Если ее приложить, так чтобы отверстия совпали с кнопками, нажать на кнопки, попавшие в прорези и повторить эту операцию еще 3 раза, поворачивая каждый раз доску на 90 градусов по часовой стрелке, то получится нужная последовательность. Кнопки в прорезях нажимаются слева-направо, сверху-вниз.

Пример:

Кнопки на сундуке:

[V][e][r][y]
[H][a][r][d]
[P][a][s][s]
[W][o][r][d]

Доска с прорезями (#-прорезь):

[][][#][#]
[][][][]
[][][#][]
[][][#][]

Получение ключа, наложением доски на панель кнопок:

1 шаг)	2 шаг)	3 шаг)	4 шаг)
[][][#][#]	[][][][]	[][#][][]	[#][][][]
[][][][]	[][][][]	[][#][][]	[#][][#][#]
[][][#][]	[#][#][][#]	[][][][]	[][][][]
[][][#][]	[][][][#]	[#][#][][]	[][][][]
[][][r][y]	[][][][]	[][e][][]	[V][][][]
[][][][]	[][][][]	[][a][][]	[H][][r][d]
[][][s][]	[P][a][][s]	[][][][]	[][][][]
[][][r][]	[][][][d]	[W][o][][]	[][][][]

gysr Pasd eaWo VHrd

Итоговая последовательность:

gysrPasdeaWoVHrd

Задание № 2. Космический ковбой Джо

У лихого космического ковбоя Джо есть верный конь Максвелл. Хотя конем его можно назвать с большой натяжкой, потому что в отличие от его земных сородичей у Максвелла n пар ног. Чтобы замена подков не превращалась в сплошные мучения, Джо выработал следующий алгоритм: он вытягивает из ящика подкову и примеряет ее к левой ноге Максвелла, если подкова подходит, он закрепляет ее. Это занимает у него 1 минуту. Если подкова не подходит на левую ногу, т.е. является правой, и при этом есть необутые правые ноги, Джо закрепляет подкову на ближайшей из них. Это занимает еще одну минуту, т.е. на такую подкову Джо тратит в итоге 2 минуты. Если же свободных правых ног нет, Джо относит подкову на склад. Это также отнимает одну дополнительную минуту, т.е. в этом случае на подкову тратится в итоге также 2 минуты. Процесс продолжается, пока все левые ноги будут подкованы. Далее Джо аналогичным образом занимается правыми ногами своего коня. Если все левые ноги уже подкованы, а Джо снова вытащил левую подкову, то он сразу несет её на склад за 1 мин. Если все правые уже подкованы, а Джо вытащил правую подкову, то он сначала пробует надеть её на левую ногу, и после неудачи несет подкову на склад. т.е. процесс занимает 2 мин.

Сколько времени в худшем случае займет у Джо этот процесс, если известно, что в ящике a левых и b правых подков?

Задание №3. Средняя скорость

Полдороги водитель проехал со скоростью X , а вторую половину — со скоростью Y . Какая у него была средняя скорость Z на всем пути? Скорость лежит в интервале $[0,100]$. X и Y — целые числа. Ответ вывести с точностью до трех знаков после запятой.

Вход:

$X Y$

Выход:

Z

Пример №1:

10 10

Результат:

10.000

Задание №4. База студентов

Группы в специальности КС называются КС10, КС20, .. КС50 - в зависимости от курса. Однако в году X группа КС50 - это одни студенты, а в году У - это другие студенты. Эта система всем надоела, поэтому в мае 2015 года решили перейти к другой нумерации КС-2015, КС-2016 и т.п., в зависимости от года поступления. Напишите программу, чтобы это сделать. Все студенты учатся или учились 5 лет. Нужно вывести всех студентов по группам, начиная с самой первой исторически.

Вход:

Первая строка содержит число N — количество студентов в общем списке [1,100].

Далее идут N строк в формате:

<фамилия> <группа> <год защиты или -1>

Выход:

M — количество групп

далее должны идти M блоков в формате

<название группы> <K-количество студентов>

далее для данной группы K строк с фамилиями студентов

Пример №1:

10

Варданян КС-30 -1

Гуменюк КС-30 -1

Егоров КС-30 -1

Жуков КС-30 -1

Захарова КС-30 -1

Анцеров КС-50 2002

Васюков КС-50 2002

Гринчик КС-50 2002

Дыньков КС-50 2002

Ершова КС-50 2002

Результат:

2

КС-1997 5

Анцеров

Васюков

Гринчик

Дыньков

Ершова

КС-2012 5

Варданян

Гуменюк

Егоров

Жуков

Захарова

Задание №5. Социализация

Обучение в университете - это не только приобретение знаний, но и навыков социализации. В начале 1-го курса студенты не знают друг друга, но потом, в процессе занятий, обедов и экзаменов завязываются знакомства. В конце 1-го года обучения куратор курса решил узнать, все ли студенты знакомы друг с другом. Если студент А знает студента В, а студент В - студента С, то считается, что А знаком с С через В. Другими словами, куратору стало интересно, является ли сеть знакомств связной, или же студенты разбиты на несвязные группы. Для этого он выгрузил из социальной сети ВКонтакте граф знакомств в виде матрицы смежности.

Вам нужно вывести по этому графу количество его компонент связности.

Вход:

N — количество студентов

Далее идет матрица $N \times N$ — N строк, N в интервале $[1, 100]$

Выход:

m — количество компонент связности

Пример №1:

7

1 1 0 0 0 0 0

1 1 1 0 0 0 0

0 1 1 1 0 0 0

0 0 1 1 0 0 0

0 0 0 0 1 0 0

0 0 0 0 0 1 1

0 0 0 0 0 1 1

Результат:

3

Задание №6. Рассеянный профессор

При написании докторской диссертации Семен Петрович Лисоед использовал около 1000 литературных источников. При этом после прохождения теста на плагиат объем его работы заметно сократился и, соответственно, сократилось и число литературных источников.

Помогите Семену Петровичу учесть какие ссылки остались, а какие нет.

На вход программы подаются количество оставшихся литературных источников и их номера, записанные в неотсортированном порядке через пробел.

На выходе требуется получить отсортированный по возрастанию список литературных источников, такой, что в случае присутствия 3-х и более подряд идущих ссылок они были записаны как диапазон.

Пример 1

вход

6 2 6 8 1 5 4

Выход

1 2 [4-6] 8

Пример 2

Вход

10 16 12 17 11 10 24 18 2 7 1

Выход

1 2 7 [10-12] [16-18] 24

Задание № 7. Расщепление времени

Для безопасных путешествий во времени необходимо строго следить за временем нахождения в точках пространства и не допускать пересечений. Для соблюдения этого фундаментального принципа Повелителям Времени необходимо изобрести Великий Расщепитель Времени. Основной задачей Расщепителя является анализ временных промежутков возможных путешествий во времени.

На вход Расщепителя подается список отрезков времени (желаемое время посещения одной из точек пространства). Программа отбрасывает все пересекающиеся интервалы, отдавая при этом предпочтение тем, которые стоят в списке первыми (т.е. из двух пересекающихся интервалов выбирается, тот, что идет в списке первым).

На выходе программы получается список временных интервалов, на которые разбиваются земные сутки (изначальный интервал составляет 00:00-23:59) для рассчитываемой точки пространства.

Пример 1

Вход

6

09 00 12 20

12 20 15 25

14 15 16 12

15 30 18 45

18 40 21 30

22 00 23 50

Выход

00:00-09:00

09:00-12:20

12:20-15:25

15:25-15:30

15:30-18:45

18:45-22:00

22:00-23:50

23:50-23:59

Пример 2

Вход

5

03 00 04 15

02 02 16 45

06 18 12 14

12 00 13 20

18 01 19 59

Выход

00:00-03:00

03:00-04:15

04:15-06:18

06:18-12:14

12:14-18:01

18:01-19:59

19:59-23:59

Задание № 8. Змеи-Пожиратели

Охотничьи угодья змей-пожирателей представляют собой квадратную поляну $N*N$ метров. Чтобы случайно не поглотить друг друга змеи-пожиратели охотятся по одиночке. Они вползают на полянку ночью и мгновенно пожирают все, что попадется им на пути. На полянке в густой траве любят спать слоники и бегемоты. Если змея пожирает слоника, то ее длина увеличивается на 3 метра и она поворачивает налево, а если ей попадется бегемот, то она удлиняется на 2 метра и поворачивает вправо. Начальная длина каждой голодной змеи - 1 метр.

Зная расположение всех слоников и бегемотов, а так же место появления змеи и ее направление требуется узнать ее окончательную длину при выходе с полянки.

На вход программе подается размер полянки, общее количество травоядных, координаты слонов и бегемотов (буквой e обозначены слоны, буквой b обозначены бегемоты), место вползания змеи и ее направление (направления: r - вправо, l - влево, t - вверх, b - вниз).

На выходе из программы - одно число - длина змеи.

Пример № 1:

Вход

5
10
1 1 e
1 3 e
2 1 b
2 2 e
2 4 e
3 2 b
3 3 e
4 3 b
5 2 e
5 5 e
4 5 l

Выход

16

Пример № 2:

Вход

7
10
1 2 e
1 5 b
2 3 b
3 2 e
4 3 b
4 6 e
6 1 e
6 2 e
6 4 b
7 4 e
3 7 t

Выход

12