

Лабораторная работа №1

Ключевые понятия: оператор, операция, приоритеты операций, типология операторов.

Необходимо усвоить: методы оценки сложности алгоритмов, ввод/вывод на C++.

Тематика заданий направлена на проверку, повторение и закрепление знаний и навыков, полученных в рамках курса ОАИП. В данной лабораторной работе повторяются языковые конструкции общие для языка C и C++.

Для подготовки заданий, использовалась книга «Меньшиков Ф.В., Олимпиадные задачи по программированию. / Ф. В. Меньшиков., Спб.: Питер, 2006», доступная в электронном варианте.

Для успешной защиты лабораторной работы необходимо: Знать ключевые понятия, усвоить новый материал, выполнить поставленные задания в соответствии с полученным вариантом.

Задание.

Временные ограничения приводятся для компьютера с процессором Intel Celeron 400МГц. Необходимо придерживаться одного из возможных стандартов кодирования и оформления исходных текстов.

За лабораторное занятие **обязательно** необходимо выполнить базовую часть задания, дома реализовать оставшееся. На следующем занятии осуществляется защита лабораторной работы. В заданиях могут предлагаться необязательные требования, они отмечаются «*» и их можно не выполнять.

Вариант 1

Задача 7F. Удаление клеток

Из прямоугольного листа клетчатой бумаги (M строк, N столбцов) удалили некоторые клетки. На сколько кусков распадется оставшаяся часть листа? Две клетки не распадаются, если они имеют общую сторону.

Ограничения: $1 \leq M, N \leq 100$, время 1 с.

Ввод из файла gemsquar.in. В первой строке находятся числа M и N , в следующих M строках - по N символов. Если клетка не была вырезана, этому соответствует знак #, если вырезана - точка.

Вывод в файл gemsquar.out. Вывести одно число.

Примеры

Ввод 1

```
4 8
#.#.#.#
.....##
#.#.#.#
##.##.##
```

Вывод 1

```
6
```

Задача 1D. Треугольник и точка

В декартовой системе координат на плоскости заданы координаты вершин треугольника и ещё одной точки. Определить, принадлежит ли эта точка треугольнику.

* Реализовать: класс точка, класс треугольник, метод для определения принадлежности точки треугольнику.

** Реализовать операторы для вывода разработанных классов на консоль с помощью потоков(std::cout).

Ограничения: координаты вершин - целые числа, для любой точки выполняются следующие условия: $-10\,000 \leq x, y \leq 10\,000$, время 1 с.

Ввод из файла tria-pt.in. В четырёх строках находятся пары чисел - координаты точек. Числа в первых трёх строках - это координаты вершин треугольника, в четвёртой строке - координаты тестируемой точки.

Вывод в файл tria-pt.out. Вывести слово "In", если точка находится внутри треугольника, или "Out" - если снаружи.

Примеры

Ввод 1	Ввод 2	Ввод 3	Ввод 4
0 0	0 0	0 0	0 0
100 0	100 0	100 0	100 0
0 100	0 100	0 100	0 100
100 100	10 10	50 50	0 0

Вывод 1	Вывод 2	Вывод 3	Вывод 4
Out	In	In	In

Вариант 2

Задача 1С. Возрастающая подпоследовательность

Даны N целых чисел X_1, X_2, \dots, X_N . Требуется вычеркнуть из них минимальное количество чисел так, чтобы оставшиеся шли в порядке возрастания.

Ограничения: $1 \leq N \leq 10\,000$, $1 \leq X_i \leq 60\,000$, время 4 с.

Ввод из файла incseq.in. В первой строке находится число N . В следующей строке - N чисел через пробел.

Вывод в файл incseq.out. В первой строке выводится количество невычеркнутых чисел, во второй - сами невычеркнутые числа через пробел в исходном порядке. Если вариантов несколько, вывести любой.

Примеры

Ввод 1
6
2 5 3 4 6 1

Вывод 1
4
2 3 4 6

Задача 1Е. Степень (+ооп +ш)

Для натуральных чисел a и n вычислить a^n .

* Реализовать: класс длинное целое, в котором будут методы для вывода, точка, класс треугольник.

Ограничения: $1 \leq a \leq 9$, $1 \leq n \leq 7000$, время 5 с.

Ввод из файла power.in. В первой строке находятся разделённые пробелом a и n .

Вывод в файл power.out. Выводится одно число - результат без стоящих впереди нулей, стоящих впереди и позади пробелов.

Примеры

Ввод 1	Ввод 2
3 20	5 50

Вывод 1	Вывод 2
3486784401	88817841970012523233890533447265625

Вариант 3

Задача 2С. Маршрут

В таблице из N строк и N столбцов клетки заполнены цифрами от 0 до 9. Требуется найти

такой путь из клетки (1, 1) в клетку (N, N), чтобы сумма цифр в клетках, через которые он пролегает, была минимальной; из любой клетки ходить можно только вниз или вправо.

Ограничения: $2 \leq N \leq 250$, время 1 с.

Ввод из файла route.in. В первой строке находится число N . В следующих N строках содержатся по N цифр без пробелов.

Вывод в файл route.out. Выводятся N строк по N символов. Символ решётка показывает, что маршрут проходит через эту клетку, а минус - что не проходит. Если путей с минимальной суммой цифр несколько, вывести любой.

Примеры

Ввод 1

```
3
943
216
091
```

Вывод 1

```
#--
###
--#
```

Задача 2В. Перестановки (+ооп: множество на вход, картежи на выходе, +другие символы)

Дана строка, состоящая из M попарно различных символов. Вывести все перестановки символов данной строки.

Ограничения: $2 \leq M \leq 8$, символы - буквы латинского алфавита и цифры, время 1 с.

Ввод из файла permut.in. В первой строке файла находится исходная строка.

Вывод в файл permut.out. Вывести в каждой строке файла по одной перестановке.

Перестановки можно выводить в любом порядке. Повторений и строк, не являющихся перестановками исходной, быть не должно.

Примеры

Ввод 1

AB

Ввод 2

IOX

Вывод 1

AB
BA

Вывод 2

XOI
OIX
IXO
XIO
OXI
IOX

Вариант 4

Задача 2Г. Спираль

Вывести квадрат, состоящий из $N \times N$ клеток, заполненных числами от 1 до N^2 по спирали (см. примеры).

Ограничения: $2 \leq N \leq 100$, время 1 с.

Ввод из файла spiral.in. В первой строке находится единственное число N .

Вывод в файл spiral.out. Выводятся N строк по N чисел, разделённых пробелами. Не допускается начинать спираль в ином, кроме верхнего левого, углу, закручивать спираль против часовой стрелки или изнутри наружу.

Примеры

Ввод 1

3

Ввод 2

4

Ввод 3

5

Вывод 1

```
1 2 3
8 9 4
7 6 5
```

Вывод 2

```
1 2 3 4
12 13 14 5
11 16 15 6
10 9 8 7
```

Вывод 3

```
1 2 3 4 5
16 17 18 19 6
15 24 25 20 7
14 23 22 21 8
```

Задача 2D. Пересечение отрезков (+ооп: отрезок, точка, методы: принадлежность точки отрезку, пересечение отрезков, печать через поток)

Два отрезка на плоскости заданы целочисленными координатами своих концов в декартовой системе координат. Требуется определить, существует ли у них общая точка.

Ограничения: координаты целые и по модулю не превосходят 10 000, время 1 с.

Ввод из файла segments.in. В первой строке содержатся координаты первого конца первого отрезка, во второй - второго конца первого отрезка, в третьей и четвёртой - координаты концов второго отрезка.

Вывод в файл segments.out. Выводится слово "Yes", если общая точка есть, или слово "No" - в противном случае.

Примеры

Ввод 1	Ввод 2
0 0	0 0
1 0	1 0
1 0	2 0
1 1	3 0
Вывод 1	Вывод 2
Yes	No

Вариант 5

Задача 3F. Змейка

Вывести квадрат, состоящий из $N \times N$ ячеек, заполненных числами от 1 до N^2 "змейкой" (см. примеры).

Ограничения: $2 \leq N \leq 100$, время 1 с.

Ввод из файла serpent.in. В первой строке находится единственное число N .

Вывод в файл serpent.out. Выводится N строк по N чисел, разделённых пробелами. Не допускаются начало змейки в другом углу или другое её направление.

Примеры

Ввод 1	Ввод 2	Ввод 3
3	4	5
Вывод 1	Вывод 2	Вывод 3
1 2 6	1 2 6 7	1 2 6 7 15
3 5 7	3 5 8 13	3 5 8 14 16
4 8 9	4 9 12 14	4 9 13 17 22
	10 11 15 16	10 12 18 21 23
		11 19 20 24 25

Задача 2Е. Длинная сумма (+ооп)

Даны два целых неотрицательных числа: M и N . Найти их сумму.

Ограничения: $0 \leq M, N < 10^{30\,000}$, время 1 с.

Ввод из файла longsum.in. В первой строке содержится M , во второй - N .

Вывод в файл longsum.out. В первой строке вывести сумму без пробелов и ведущих нулей.

Примеры

Ввод 1
12345678901234567890123456789
111111111111111111111111111111
Вывод 1
13456790012345679001234567900

Вариант 6

Задача 4В. Разложение на слагаемые

Вывести все представления натурального числа N суммой натуральных чисел. Перестановка слагаемых нового способа представления не даёт.

Ограничения: $2 \leq N \leq 40$, время 2 с.

Ввод из файла descomp.in. В первой строке находится единственное число N .

Вывод в файл descomp.out. В каждой строке выводится одно из представлений. В сумме слагаемые разделяются знаком "+".

Примеры

Ввод 1

4

Вывод 1

1+1+1+1

1+2+1

1+3

2+2

Задача 3Е. Длинное произведение (+ооп)

Даны целые неотрицательные числа M и N . Найти $M * N$.

Ограничения: $0 \leq M, N \leq 10^{2500}$, время 5 с.

Ввод из файла longprod.in. В первой строке находится число M , во второй - N .

Вывод в файл longprod.out. Вывести одно число - результат умножения.

Примеры

Ввод 1

9876543210

1023456789

Вывод 1

10108215200126352690

Вариант 7

Задача 3В. Перестановки (2) (+ооп: множества и прочие)

Дана строка, состоящая из M символов. Вывести все перестановки символов данной строки.

Ограничения: $2 \leq M \leq 8$, символы - буквы латинского алфавита и цифры, время 1 с.

Ввод из файла permut2.in. В первой строке файла находится исходная строка.

Вывод в файл permut2.out. Вывести в каждой строке файла по одной перестановке.

Перестановки можно выводить в любом порядке. Повторений и строк, не являющихся перестановками исходной, быть не должно.

Примеры

Ввод 1

AB

Ввод 2

122

Вывод 1

AB

BA

Вывод 2

122

212

221

Задача 3С. Копилка (ооп: копилка)

Задан вес E пустой копилки и вес F копилки с монетами. В копилке могут находиться монеты N видов, для каждого вида известна ценность P_i и вес W_i одной монеты. Найти минимальную и максимальную суммы денег, которые могут находиться в копилке.

Ограничения: $1 \leq E \leq F \leq 10\,000$, $1 \leq N \leq 500$, $1 \leq P_i \leq 50\,000$, $1 \leq W_i \leq 10\,000$, все числа целые, время 2 с.

Ввод из файла piggy.in. В первой строке находятся числа E и F , во второй - число N , в следующих N строках - по два числа, P_i и W_i .

Вывод в файл piggy.out. Выводятся два числа через пробел - минимальная и максимальная суммы. Если копилка не может иметь точно заданный вес при условии, что она наполнена монетами заданных видов, - вывести "This is impossible."

Примеры

Ввод 1	Ввод 2	Ввод 3
1000 1100	1000 1010	1000 2000
2	2	1
1 1	6 3	10 3
5 2	2 2	
Вывод 1	Вывод 2	Вывод 3
100 250	10 16	This is impossible.

Вариант 8

Задача 4F. Скобки

Дана последовательность из N круглых, квадратных и фигурных скобок. Выяснить, можно ли добавить в неё цифры и знаки арифметических действий так, чтобы получилось правильное арифметическое выражение.

Ограничения: $1 \leq N \leq 100\,000$, время 1 с.

Ввод из файла bracket.in. В первой строке находится число скобок N , во второй - N символов из набора (,), [,], {, }.

Вывод в файл bracket.out. Выводится слово "Yes", если получить правильное арифметическое выражение можно, или "No", если нельзя.

Примеры

Ввод 1	Ввод 2
6	24
([()])	{[() ([{}]) []] ({{}{})} []]
Вывод 1	Вывод 2
No	Yes

Задача 3D. Открытка и конверт

Даны размеры прямоугольных открытки и конверта. Требуется определить, поместится ли открытка в конверт.

Ограничения: размеры открытки и конверта - целые положительные числа, не превосходящие 100, время 1 с.

Ввод из файла postcard.in. В первой строке находятся размеры открытки, во второй - размеры конверта.

Вывод в файл postcard.out. Если открытку можно вложить в конверт, вывести "Possible", если нет - вывести "Impossible".

Примеры

Ввод 1
1 10
9 9
Вывод 1
Possible

Вариант 9

Задача 5B. Скобки (2)

Вывести все правильные скобочные выражения длиной N , состоящие из круглых и квадратных скобок.

Ограничения: $1 \leq N \leq 14$, N - чётное, время 2 с.

Ввод из файла bracket2.in. В первой строке находится единственное число N .

Вывод в файл bracket2.out. Каждое выражение выводится в отдельной строке.

Примеры

Ввод 1
4
Вывод 1
()
[]
[]
()
()
[]
()
()
()
[]

Задача 5D. Выпуклая оболочка (+ооп)

На плоскости заданы N точек своими декартовыми координатами. Найти минимальный периметр многоугольника, содержащего все эти точки. Гарантируется, что искомый многоугольник имеет ненулевую площадь.

Ограничения: $3 \leq N \leq 1000$, $-10\,000 \leq x_i, y_i \leq 10\,000$, все числа целые, все точки различны, время 2 с.

Ввод из файла conhull.in. В первой строке находится число N , далее - N строк с парами координат.

Вывод в файл conhull.out. Вывести одно число - длину периметра с одним знаком после запятой.

Примеры

Ввод 1
5
1 0
0 1
-1 0
0 -1
0 0
Вывод 1
5.7

Вариант 10

Задача 5F. День рождения

Заданы день и месяц рождения, а также текущие день, месяц и год. Определить, сколько дней осталось до дня рождения.

Примечание. Високосные годы - это те, номер которых делится на 400, а также те, номер которых делится на 4, но не делится на 100.

Ограничения: год от 1920 до 3000, месяц - от 1 до 12, день - от 1 до числа дней в месяце, время 1 с.

Ввод из файла birthday.in. В первой строке находятся разделённые пробелами день и месяц рождения, во второй - разделённые пробелами текущие день, месяц и год.

Вывод в файл birthday.out. Вывести число дней, оставшихся до дня рождения.

Примеры

Ввод 1	Ввод 2	Ввод 3
19 04	05 05	29 02
19 04 2002	19 04 2002	28 02 2001
Вывод 1	Вывод 2	Вывод 3
0	16	1096

Задача 5E. Системы счисления (+ооп)

Дано целое неотрицательное число в I -ричной системе счисления. Вывести это число в J -ричной системе счисления.

Ограничения: $2 \leq I, J \leq 36$, для представления цифр 10...35 используются прописные латинские буквы A...Z соответственно, число разрядов исходного числа не превышает 1000, время 5 с.

Ввод из файла scale.in. В первой строке находятся числа I и J (в десятичной системе счисления), во второй строке - число для перевода.

Вывод в файл scale.out. Вывести искомое число. Если число начинается с буквы, перед ней не должно быть нуля.

Примеры

Ввод 1

10 36
29234652

Вывод 1

HELLO

Вариант 11

Задача 6А. Закраска прямой (использование стл алгоритма, перегрузка оператора, класс отрезка + отрезок на прямой, метод пересечения)

На числовой прямой окрасили N отрезков. Известны координаты левого и правого концов каждого отрезка (L_i и R_i). Найти длину окрашенной части числовой прямой.

Ограничения: L_i и R_i - целые, $-1\,000\,000\,000 \leq L_i \leq R_i \leq 1\,000\,000\,000$, $1 \leq N \leq 15\,000$, время 1 с.

Ввод из файла cover.in. В первой строке находится число N , в следующих N строках - пары L_i и R_i .

Вывод в файл cover.out. Вывести одно число - длину окрашенной части прямой.

Примеры

Ввод 1

2
1 3
2 4

Ввод 2

1
10 10

Вывод 1

3

Вывод 2

0

Задача 4D. Площадь многоугольника

Многоугольник на плоскости задан целочисленными координатами своих N вершин в декартовой системе координат. Требуется найти площадь многоугольника. Стороны многоугольника не соприкасаются (за исключением соседних - в вершинах) и не пересекаются.

Ограничения: $3 \leq N \leq 50\,000$, координаты вершин целые и по модулю не превосходят 20 000, время 1 с.

Ввод из файла area.in. В первой строке находится число N . В следующих N строках находятся пары чисел - координаты точек. Если соединить точки в данном порядке, а также первую и последнюю точки, получится заданный многоугольник.

Вывод в файл area.out. Вывести одно число - площадь многоугольника. Его следует округлить до ближайшего числа с одной цифрой после запятой.

Примеры

Ввод 1

4
5 0
0 5
-5 0
0 -5

Ввод 2

4
0 4
0 0
3 0
1 1

Вывод 1

50.0

Вывод 2

3.5

Вариант 12

Задача 6С. Игра "Даты"

Играют двое. Задаётся какая-то дата 2004 года. Каждый игрок на своём ходе называет более позднюю дату, увеличивая на 1 или 2 либо день в месяце, либо месяц, но не то и другое сразу. При этом сочетание дня и месяца должно оставаться датой. Игрок, назвавший 31 декабря, проигрывает. Оба играют наилучшим образом. Исходя из заданной даты вывести, кто выиграет.

Ограничения: месяц от 1 до 12, день от 1 до числа дней в месяце, даты "31 декабря" во входных данных нет, время 1 с.

Ввод из файла dategame.in. В первой строке находятся числа, обозначающие день и месяц.

Вывод в файл dategame.out. Вывести 1, если выигрывает первый (начинающий) игрок, или 2 - в противном случае.

Примеры

Ввод 1	Ввод 2	Ввод 3
30 12	29 12	29 11
Вывод 1	Вывод 2	Вывод 3
2	1	2

Задача 7D. Прямая и квадраты (ооп: квадрат, прямая)

В прямоугольной декартовой системе координат прямая задана двумя принадлежащими ей точками $(0, W)$ и $(100N, E)$. Также заданы N^2 квадратов со сторонами, параллельными осям координат. Квадрат $S_{i,j}$ имеет координаты углов $(100i, 100j)$ и $(100i + 100, 100j + 100)$, $i, j = 1, 2, \dots, N$. Требуется найти количество квадратов, имеющих общую точку с прямой.

Ограничения: $1 \leq N \leq 100$, $0 \leq W, E \leq 100N$, все числа целые, время 1 с.

Ввод из файла sqline.in. В первой строке находятся три целых числа, N , W и E , разделённых пробелами.

Вывод в файл sqline.out. Вывести одно число - количество квадратов.

Примеры

Ввод 1
3 150 50
Вывод 1
4

Вариант 13

Задача Е. Путь коня

Дана шахматная доска, состоящая из $N \times N$ клеток, несколько из них вырезано. Провести ходом коня через невырезанные клетки путь минимальной длины из одной заданной клетки в другую.

Ограничения: $2 \leq N \leq 50$, время 1 с.

Ввод из файла knightw.in. В первой строке задано число N . В следующих N строках содержится по N символов. Символом # обозначена вырезанная клетка, точкой - невырезанная клетка, @ - заданные клетки (таких символов два).

Вывод в файл knightw.out. Если путь построить невозможно, вывести "Impossible", в противном случае вывести такую же карту, как и на входе, но пометить все промежуточные положения коня символом @.

Примеры

Ввод 1	Ввод 2	Ввод 3
5	5	5
.....	@...@.	@.....
..@@..	..##.	..#..
.....#...
.....
.....@

Вывод 1	Вывод 2	Вывод 3
...@.	@...@.	Impossible
.@@..	..##.	
....@	.@...@	
.....	..@..	
.....	@.....	

Задача 6D. Площадь прямоугольников (ооп, вычитать из максимума)

Дано N прямоугольников со сторонами, параллельными осям координат. Требуется определить площадь фигуры, образованной объединением данных прямоугольников.

Ввод из файла rectarea.in. В первой строке находится число прямоугольников - N . Затем идут N строк, содержащих по 4 числа: x_1, y_1, x_2, y_2 - координаты двух противоположных углов прямоугольника.

Ограничения: $1 \leq N \leq 100$, координаты целые и по абсолютному значению не превосходят 10 000, время 3 с.

Вывод в файл rectarea.out. Вывести одно число - площадь фигуры.

Примеры

Ввод 1
2
1 1 3 3
2 2 4 4
Вывод 1
7

Вариант 14 Задача 6E. Lines

В таблице из N строк и N столбцов некоторые клетки заняты шариками, другие свободны. Выбран шарик, который нужно переместить, и место, куда его нужно переместить.

Выбранный шарик за один шаг перемещается в соседнюю по горизонтали или вертикали свободную клетку. Требуется выяснить, возможно ли переместить шарик из начальной клетки в заданную, и, если возможно, то найти путь из наименьшего количества шагов.

Ограничения: $2 \leq N \leq 40$, время 1 с.

Ввод из файла lines.in. В первой строке находится число N , в следующих N строках - по N символов. Символом точки обозначена свободная клетка, латинской заглавной o - шарик, @ - исходное положение шарика, который должен двигаться, латинской заглавной x - конечное положение шарика.

Вывод в файл lines.out. В первой строке выводится y , если движение возможно, или n , если нет. Если движение возможно, далее следует N строк по N символов - как и на вводе, но буква x , а также все точки по пути заменяются плюсами.

Примеры

Ввод 1	Ввод 2	Ввод 3
5	5	5
....X	..X..	...X.
.OOOO
.....	OOOOO	O.OOO
OOOO.
@....	..@..@
Вывод 1	Вывод 2	Вывод 3
Y	N	Y
+++++		..++.
+OOOO		.++..
+++++		O+OOO
OOOO+		.++++
@++++	@

Задача 8D. Дуга на сфере ----

На поверхности планеты, являющейся шаром радиусом R , заданы две точки своими широтой и долготой. Найти минимальную длину пути по поверхности этой планеты из одной точки в другую.

Ограничения: широта в градусах от -90 до 90, долгота в градусах от -180 до 180, $100 \leq R \leq 10\,000$, все числа вещественные, время 1 с.

Ввод из файла `spherarc.in`. В первой строке находится число R , во второй строке заданы широта и долгота первой точки, в третьей строке - широта и долгота второй точки.

Вывод в файл `spherarc.out`. Вывести длину пути с двумя знаками после запятой.

Примеры

Ввод 1

```
4000
45 120
0 120
```

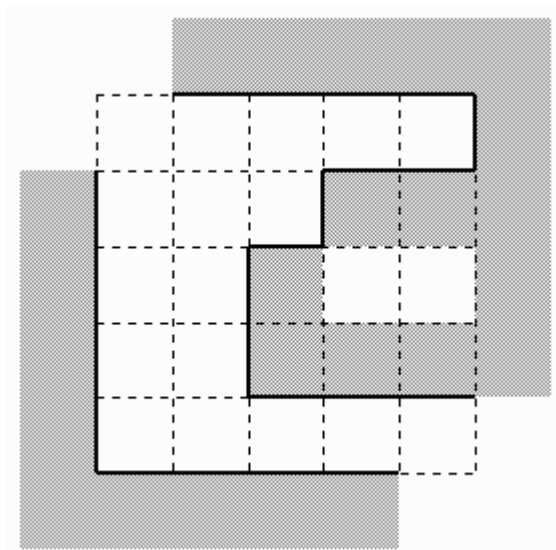
Вывод 1

```
3141.59
```

Вариант 15

Задача 6F. Покраска лабиринта +++

Лабиринт представляет собой квадрат, состоящий из $N \times N$ сегментов. Каждый из сегментов может быть либо пустым, либо заполненным монолитной каменной стеной. Гарантируется, что левый верхний и правый нижний сегменты пусты. Лабиринт обнесён сверху, снизу, слева и справа стенами, оставляющими свободными только левый верхний и правый нижний углы. Директор лабиринта решил покрасить стены лабиринта, видимые изнутри (см. рисунок). Помогите ему рассчитать количество краски, необходимой для этого.



Ограничения: $3 \leq N \leq 33$, размер сегмента 3 x 3 м, высота стен 3 м, время 1 с.

Ввод из файла `paintlab.in`. В первой строке находится число N , затем идут N строк по N символов: точка обозначает пустой сегмент, решётка - сегмент со стеной.

Вывод в файл `paintlab.out`. Вывести одно число - площадь видимой части внутренних стен лабиринта в квадратных метрах.

Примеры

Ввод 1

```
5
```

```
.....
...##
..#..
...##
.....
```

Задача 11D. Круговая площадь

Два круга заданы координатами центров в прямоугольной декартовой системе координат и радиусами. Найти площадь их пересечения.



Ограничения: во входных данных числа вещественные и по модулю не превосходят 1000, время 1 с.

Ввод из файла `circarea.in`. В первой строке находятся шесть вещественных чисел через пробел - координаты центров и радиусы двух кругов: $x_1, y_1, r_1, x_2, y_2, r_2$.

Вывод в файл `circarea.out`. Вывести одно вещественное число с двумя знаками после запятой - площадь пересечения кругов.

Примеры

Ввод 1

```
20.0 30.0 15.0 40.0 30.0 30.0
```

Вывод 1

```
608.37
```

Дополнительные задачи.

Задача 1B. Выражение

Даны N целых чисел X_1, X_2, \dots, X_N . Расставить между ними знаки "+" и "-" так, чтобы значение получившегося выражения было равно заданному целому S .

Ограничения: $2 \leq N \leq 24$, $0 \leq X_i \leq 50\,000\,000$, $-1\,000\,000\,000 \leq S \leq 1\,000\,000\,000$, время 3 с.

Ввод из файла `expr.in`. В первой строке находятся числа N и S . В следующей строке - N чисел через пробел.

Вывод в файл `expr.out`. Если получить требуемый результат невозможно, вывести "No solution", если можно, то вывести равенство. Если решение не единственное, вывести любое.

Примеры

Ввод 1

```
3 10
15 25 30
```

Вывод 1

```
15+25-30=10
```

Ввод 2

```
2 100
10 10
```

Вывод 2

```
No solution
```

Задача 8F. Грядки

Прямоугольный садовый участок шириной N и длиной M метров разбит на квадраты со стороной 1 метр. На этом участке вскопаны грядки. Грядкой называется совокупность квадратов, удовлетворяющая таким условиям:

- из любого квадрата этой грядки можно попасть в любой другой квадрат этой же грядки, последовательно переходя по грядке из квадрата в квадрат через их общую сторону;

- никакие две грядки не пересекаются и не касаются друг друга ни по вертикальной, ни по горизонтальной сторонам квадратов (касание грядок углами квадратов допускается).

Подсчитайте количество грядок на садовом участке.

Ограничения: $1 \leq N, M \leq 200$, время 1 с.

Ввод из файла beds.in. В первой строке находятся числа N и M через пробел, далее идут N строк по M символов. Символ # обозначает территорию грядки, точка соответствует незанятой территории. Других символов в исходном файле нет.

Вывод в файл beds.out. Вывести одно число - количество грядок на садовом участке.

Примеры

Ввод 1

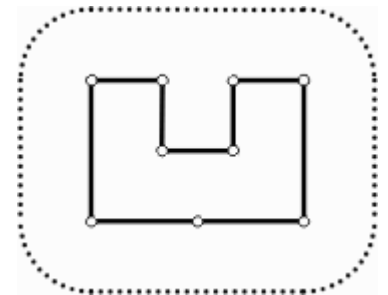
```
5 10
##.....#
.#..#...#
.###....#
..##....#
.....#
```

Вывод 1

3

Задача 15D. Стена

Жил-был жадный Король. Он приказал своему главному Архитектору построить стену вокруг его замка. Король был таким жадным, что не послушал предложение Архитектора построить красивую кирпичную стену совершенной формы с изящными высокими башнями. Вместо этого он приказал построить стену вокруг всего замка, используя минимальное количество камня, но потребовал, чтобы стена не подходила к замку ближе некоторого расстояния. Если Король узнает, что Архитектор использовал больше ресурсов для постройки стены, чем было абсолютно необходимо для удовлетворения требований, Архитектор лишится головы. Более того, Архитектор должен представить проект стены, где указано точное количество ресурсов.



Ваша задача - помочь бедному Архитектору сохранить голову, написав программу, определяющую минимальную длину стены, которую можно построить вокруг замка, удовлетворив требования Короля.

Задача слегка упрощается тем, что замок Короля представляет собой многоугольник и расположен на плоской поверхности. Архитектор уже сопоставил замку прямоугольную декартову систему координат и точно определил координаты каждого угла замка в футах.

Ввод из файла wall.in. Первая строка содержит два целых числа N и L , разделённых пробелом: N - число углов в замке Короля, а L - минимальное число футов, на которое Король разрешил приблизить стену к замку.

Следующие N строк описывают координаты углов замка в порядке обхода по часовой стрелке. Каждая строка содержит два целых числа x_i и y_i , разделённых пробелом и представляющих собой координаты i -го угла в футах. Все углы имеют различные координаты, и стены замка не пересекаются иначе как в углах.

Ограничения: $3 \leq N \leq 1000$, $1 \leq L \leq 1000$, $-10\,000 \leq x_i, y_i \leq 10\,000$, время 2 с.

Вывод в файл wall.out. Выводится единственное число - минимальная длина стены в футах, которая может быть построена вокруг замка согласно требованиям Короля. Вы должны представить Королю целое число футов, потому что вещественные числа ещё не изобретены. Однако результат нужно округлить так, чтобы он отличался не более чем на 8 дюймов от правильного (1 фут = 12 дюймов), потому что большей неточности Король не потерпит.

Примеры

Ввод 1

```
9 100
200 400
300 400
300 300
400 300
400 400
500 400
500 200
350 200
200 200
```

Вывод 1

```
1628
```

Задача 14D. Пирамиды

Рассматриваемые пирамиды имеют треугольник в основании, то есть являются тетраэдрами. Требуется по заданным длинам рёбер пирамиды найти её объём.

Ограничения: длины рёбер - целые положительные числа, не превосходящие 1000, время 1 с.

Ввод из файла pyramids.in. В первой строке находятся 6 чисел через пробел - длины рёбер пирамиды $ABCD$. Порядок рёбер: AB, AC, AD, BC, BD, CD .

Вывод в файл pyramids.out. Вывести одно вещественное число с четырьмя знаками после запятой - объём пирамиды.

Примеры

Ввод 1

```
1 1 1 1 1 1
```

Вывод 1

```
0.1179
```

Ввод 2

```
1000 1000 1000 3 4 5
```

Вывод 2

```
1999.9937
```

Задача 11E. Гомер Симпсон

Обеденный перерыв Гомера Симпсона составляет T миллисекунд. Один гамбургер Гомер съедает за N миллисекунд, один чизбургер - за M . Требуется найти максимальное суммарное число гамбургеров и чизбургеров, которые Гомер может съесть в течение обеденного перерыва.

Ограничения: $1 \leq M, N, T \leq 1\,000\,000$, все числа целые, время 2 с.

Ввод из файла homer.in. В первой строке находятся три числа - M, N и T , разделённые пробелами.

Вывод в файл homer.out. Вывести максимальное суммарное число гамбургеров и чизбургеров. Если остаётся какое-то время, требуется



указать его через пробел. Предпочтителен вариант, когда дополнительного времени остаётся как можно меньше.

Примеры

Ввод 1	Ввод 2	Ввод 3
3 5 54	3 5 55	4 4 6
Вывод 1	Вывод 2	Вывод 3
18	17	1 2

Задача Е. Путь спелеолога

Пещера представлена кубом, разбитым на N частей по каждому измерению (то есть на N^3 кубических клеток). Каждая клетка может быть или пустой, или полностью заполненной камнем. Исходя из положения спелеолога в пещере, требуется найти, какое минимальное количество перемещений по клеткам ему требуется, чтобы выбраться на поверхность. Переходить из клетки в клетку можно, только если они обе свободны и имеют общую грань.

Ограничения: $1 \leq N \leq 30$, время 1 с.

Ввод из файла speleo.in. В первой строке содержится число N . Далее следует N блоков. Блок состоит из пустой строки и N строк по N символов: # - обозначает клетку, заполненную камнями, точка - свободную клетку. Начальное положение спелеолога обозначено заглавной буквой s. Первый блок представляет верхний уровень пещеры, достижение любой свободной его клетки означает выход на поверхность. Выход на поверхность всегда возможен.

Вывод в файл speleo.out. Вывести одно число - длину пути до поверхности.

Примеры

Ввод 1

3

.##

.#.
.#s
.#.

...
###

Вывод 1

6

Комментарий 1

Нужно спуститься на уровень вниз,
сделать два движения на запад,
подняться на уровень вверх,
сделать движение на юг,
подняться на уровень вверх.

Задача 15Е. Семечки

На базаре есть ряд из N мест, где продаются семечки подсолнечника. Потенциальные покупатели идут вдоль ряда, затем в некоторый момент останавливаются и покупают семечки. Качество семечек от места к месту различается незначительно, так что разницы только в цене семечек и положении места.

Перед тем как выйти на рынок в качестве ещё одного продавца семечек, Вы провели исследование рынка, чтобы найти зависимость числа покупателей от двух названных факторов. Исследование показывает, что большинство покупателей следуют одному и тому же шаблону. Они проходят мимо нескольких мест, замечая и запоминая цены, а затем после обхода K мест, возвращаются к месту с наименьшей замеченной ценой, совершают там покупку, затем покидают базар. Если есть несколько мест с одинаковой ценой, покупатель выбирает ближайшее.

Предположим, что есть пять мест с ценами 37, 34, 34, 35, 33. Если покупатель с $K = 4$ идёт слева направо, он видит семечки по ценам 37, 34, 34, 35. В этот момент он решает, что видел достаточно, возвращается к третьему месту и покупает семечки там. Хотя на втором месте цена та же, что и на третьем, покупателю до него идти дальше. Если бы тот же покупатель зашёл справа, он бы увидел цены 33, 35, 34, 34, затем остановился и вернулся бы к пятому месту.

Число мест, пройденных до принятия решения (K), является функцией жадности и терпеливости покупателя, и, очевидно, различается у разных покупателей. Исследование выявило средний процент B_K покупателей для всех значений K ($1 \leq K \leq N$, $0 \leq B_K \leq 99$, сумма всех B_K равна 100).

Вам следует определить оптимальную стратегию на этом рынке (то есть цену и положение нового места, которое максимизирует ожидаемый средний доход) в предположении, что половина клиентов идёт в направлении от первого места к N -му, а другая половина - от N -го места к первому, и они следуют описанному шаблону.

Ввод из файла seeds.in. В первой строке находится число существующих мест N , во второй строке - N целых чисел - цены на каждом месте, в третьей строке - N целых чисел в диапазоне от 0 до 99 - значения B_K для каждого K . Все числа в строках разделены пробелами.

Ограничения: $2 \leq N \leq 100$, исходные цены - целые числа от 1 до 9999, время 2 с.

Вывод в файл seeds.out. Выводятся два целых числа - L и P . L ($0 < L < N$) - это число существующих мест, после которых должно быть размещено новое (Вам не разрешается устанавливать своё место первым или последним). Число P - оптимальная цена. Если существует более чем одно оптимальное решение, Вы должны выбрать решение с минимальным L , а среди них - с минимальным P .

Примеры

Ввод 1

```
5
37 34 34 35 33
10 20 30 30 10
```

Вывод 1

Задача 14А. Марковский цикл

Ограниченный алгоритм Маркова состоит из последовательности предложений

$$s_1s_2\dots s_N \rightarrow d_1d_2\dots d_N,$$

где s_i и d_i - символы из алфавита А, В, С. Подстрока $s_1s_2\dots s_N$ называется левой частью, а $d_1d_2\dots d_N$ - правой частью предложения.

Алгоритм выполняется над исходной текстовой строкой, состоящей из прописных латинских букв А, В, С, следующим образом: перебираются все предложения, начиная с первого. Если левая часть предложения входит в текстовую строку, то самое левое вхождение заменяется правой частью этого предложения, и поиск вновь начинается с первого предложения. Если ни одно предложение не может быть применено, алгоритм останавливается.

При выполнении алгоритма возможны два результата: либо остановка, либо бесконечный цикл с определенным периодом. По данной строке и набору предложений алгоритма Маркова определить количество "ациклических" (выполненных до начала цикла) шагов и длину самого цикла. Если алгоритм останавливается, то длина цикла считается нулевой, а все выполненные шаги - ациклическими.

Ввод из файла markovc.in. В первой строке находится исходная текстовая строка, а в следующих строках - предложения, по одному в строке. Строки могут содержать произвольное количество незначащих пробелов.

Ограничения: длина исходной текстовой строки и левых частей предложений - от 1 до 12 букв, количество предложений - от 1 до 50, время 3 с.

Вывод в файл markovc.out. Вывести два целых числа, разделённых пробелом - количество ациклических шагов и длину цикла.

Примеры

Ввод 1

```
АВАВС
С ->А
АВ ->ВА
ВАА->АВС
```

Вывод 1

```
3 3
```

Задача 14С. Упаковка символов

Билл пытается компактно представить последовательности прописных символов от А до Z с помощью упаковки повторяющихся подпоследовательностей внутри них. Например, один из способов представить последовательность ААААААААААВАВАВССD - это 10 (А) 2 (ВА) В2 (С) D. Он формально определяет сжатые последовательности символов и правила перевода их в несжатый вид следующим образом:

- Последовательность, содержащая один символ от A до Z, является упакованной. Распаковка этой последовательности даёт ту же последовательность из одного символа.
- Если s и q - упакованные последовательности, то sq - также упакованная последовательность. Если s распаковывается в s' , а q распаковывается в q' , то sq распаковывается в $s'q'$.
- Если s - упакованная последовательность, то $x(s)$ - также упакованная последовательность, где x - десятичное представление целого числа, большего 1. Если s распаковывается в s' , то $x(s)$ распаковывается в s' , повторённую x раз.

Следуя этим правилам, легко распаковать любую заданную упакованную последовательность. Однако Биллу более интересен обратный переход. Он хочет упаковать заданную последовательность так, чтобы результирующая сжатая последовательность содержала наименьшее возможное число символов.

Ограничения: длина исходной последовательности от 1 до 100, время 2 с.

Ввод из файла folding.in. В первой строке находится последовательность символов от A до Z.

Вывод в файл folding.out. В единственной строке выводится упакованная последовательность наименьшей длины, которая распаковывается в заданную последовательность. Если таких последовательностей несколько, можно выводить любую.

Примеры

Ввод 1

AAAAAAAAABABABCCD

Вывод 1

9 (A) 3 (AB) CCD

Ввод 2

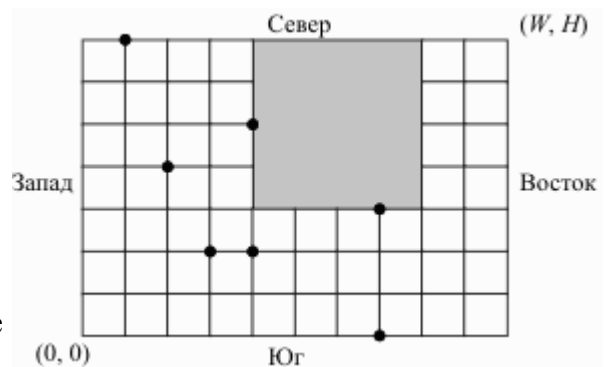
NEERCYESYESYESNEERCYESYESYES

Вывод 2

2 (NEERC3 (YES))

Задача 14Е. Поле для крикета

Жил-был жадный Король. Он приказал своему главному Архитектору построить поле для королевского крикета в парке. Король был таким жадным, что не послушал предложение своего Архитектора построить поле прямо в центре парка и окружить его живописным бордюром деревьев, специально посаженных вокруг. Вместо этого он приказал не срубить деревья и не сажать новых, но построить самое большое поле для крикета, какое только можно. Если Король обнаружит, что Архитектор посмел тронуть даже единственное дерево в парке или спроектировал меньшее поле, чем было возможно, Архитектор лишится головы. Более того, он потребовал от Архитектора представить план поля, где указаны его точное положение и размер.



Ваша задача - помочь бедному Архитектору сохранить голову, написав программу, которая найдёт максимальный размер поля для крикета и его положение внутри парка, удовлетворяющие требованиям Короля.

Задача слегка упрощена тем, что парк Короля имеет прямоугольную форму и расположен на плоской поверхности. Более того, границы парка параллельны направлениям север - юг и восток - запад. В то же время игра в королевский крикет всегда происходит на квадратном поле, границы которого также параллельны направлениям север - юг и восток - запад. Архитектор уже сопоставил парку прямоугольную декартову систему координат и

точно определил координаты каждого дерева. Оси этой системы координат, конечно, параллельны направлениям север - юг и восток - запад. Юго-западный угол парка имеет координаты $(0, 0)$, а северо-восточный - координаты (W, H) , где W и H - длина и ширина парка соответственно.

В этой задаче вы можете пренебречь диаметром деревьев. Деревья не могут находиться внутри поля для крикета, но могут располагаться на его сторонах. Поле для крикета может также касаться границы парка, но не должно лежать вне парка.

Ввод из файла `cricket.in`. Первая строка содержит три целых числа, N , W и H , разделённых пробелами: N - число деревьев в парке, W и H - длина и ширина парка соответственно.

Следующие N строк описывают координаты деревьев в парке. Каждая строка содержит два целых числа x_i и y_i , разделённых пробелом и представляющих собой координаты i -го дерева. Все деревья имеют различные координаты.

Ограничения: $1 \leq N \leq 100$, $1 \leq W, H \leq 10\,000$, $0 \leq x_i \leq W$, $0 \leq y_i \leq H$, время 1 с.

Вывод в файл `cricket.out`. Вывести через пробел три целых числа, P , Q и L , где (P, Q) - координаты юго-западного угла поля для крикета, L - длина его сторон. Если существует несколько возможных положений поля максимального размера, вывести любое.

Примеры

Ввод 1

```
7 10 7
3 2
4 2
7 0
7 3
4 5
2 4
1 7
```

Вывод 1

```
4 3 4
```