BRONZE PROBLEMS

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Three problems numbered 11 through 13

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Problem 11: Checkers [Rob Kolstad, 2008]

The cows have taken up the game of checkers with a vengeance.

Unfortunately, despite their infinite enjoyment of playing, they

are terrible at the endgame. They want your help.

Given an NxN (4 <= N <= 30) checkboard, determine the optimal set

of moves to end the game on the next move. Checkers move only on

the '+' squares and capture by jumping 'over' an opponent's piece.

The piece is removed as soon as it is jumped. See the example below

where N=8:

- + - + - + - + The K's mark Bessie's kings; the o's represent the

+ - + - + - + - opponent's checkers. Bessie always moves next. The

- + - K - + - + Kings jump opponent's checkers successively in any

+ - + - + - + - diagonal direction (and removes pieces when jumped).

- o - o - + - +

+ - K - + - + - For this board, the solution requires the lower left

- o - + - + - + King to jump successively across all three of the

+ - K - + - K - opponents' checkers, thus ending the game (moving K

marked as >K<):

Original After move 1 After move 2 After move 3

- + - + - + - + - + - + - + - + - + - + - + - + - + - + - + - +

+ - + - + - + - + - + - + - + - + - + - + - + - + - + - + - + -

- + - K - + - + - + - K - + - + - + - K - + - + - + - K - + - +

+ - + - + - + - + - + - + - + - + ->K<- + - + - + - + - + - + -

- o - o - + - + - o - o - + - + - + - o - + - + - + - + - + - +

+ - K - + - + - >K<- K - + - + - + - K - + - + - + - K ->K<- + -

- o - + - + - + - + - + - + - + - + - + - + - + - + - + - + - +

+ ->K<- + - K - + - + - + - K - + - K - + - K - + - K - + - K -

The moves traversed these squares:

1 2 3 4 5 6 7 8 R C

1 - + - + - + - + start: 8 3

2 + - + - + - + - move: 6 1

3 - + - K - + - + move: 4 3

4 + - \* - + - + - move: 6 5

5 - o - o - + - +

6 \* - K - \* - + -

7 - o - + - + - +

8 + - K - + - K -

Write a program to determine the (unique, as it turns out) game-ending

sequence for an NxN input board if it exists. There is at least a

king and at least one opponent piece on the board.

PROBLEM NAME: chkr

INPUT FORMAT:

\* Line 1: A single integer: N

\* Lines 2..N+1: Line i+1 contains N characters (each one of: '-', '+',

'K', or 'o') that represent row i of a proper checkboard.

SAMPLE INPUT (file chkr.in):

8

-+-+-+-+

+-+-+-+-

-+-K-+-+

+-+-+-+-

-o-o-+-+

+-K-+-+-

-o-+-+-+

+-K-+-K-

OUTPUT FORMAT:

\* Lines 1..?: If this sequence of moves is impossible, output

"impossible" on a line by itself. If such a sequence exist,

each line contains two space-separated integers that represent

successive locations of a king whose moves will win the game.

SAMPLE OUTPUT (file chkr.out):

8 3

6 1

4 3

6 5

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Problem 12: Bad Grass [Fatih , 2008]

Bessie was munching on tender shoots of grass and, as cows do,

contemplating the state of the universe. She noticed that she only

enjoys the grass on the wide expanses of pasture whose elevation

is at the base level of the farm. Grass from elevations just 1 meter

higher is tougher and not so appetizing. The bad grass gets worse

as the elevation increases.

Continuing to chew, she realized that this unappetizing food grows

the sides of hills that form a set of 'islands' of bad grass among

the sea of tender, verdant, delicious, abundant grass.

Bessie donned her lab coat and vowed to determine just how many

islands of bad grass her pasture had. She created a map in which

she divided the pasture into R (1 < R <= 1,000) rows and C (1 < C

<= 1,000) columns of 1 meter x 1 meter squares. She measured the

elevation above the base level for each square and rounded it to a

non-negative integer. She noted hungrily that the tasty grass all

had elevation 0.

She commenced counting the islands. If two squares are neighbors

in any of the horizontal, vertical or diagonal directions then they

are considered to be part of the same island.

How many islands of bad grass did she count for each of the supplied

maps?

PROBLEM NAME: badgras

INPUT FORMAT:

\* Line 1: Two space-separated integers: R and C

\* Lines 2..R+1: Line i+1 describes row i of the map with C space

separated integers

SAMPLE INPUT (file badgras.in):

8 7

4 3 2 2 1 0 1

3 3 3 2 1 0 1

2 2 2 2 1 0 0

2 1 1 1 1 0 0

1 1 0 0 0 1 0

0 0 0 1 1 1 0

0 1 2 2 1 1 0

0 1 1 1 2 1 0

OUTPUT FORMAT:

\* Line 1: A single integer that specifies the number of islands.

SAMPLE OUTPUT (file badgras.out):

2

OUTPUT DETAILS:

There are two islands. The big one on the left side that extends

all the way to the bottom through a diagonal and the small one on

the upper-right corner.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Problem 13: Hay Expenses [Neal Wu, 2008]

Every day Farmer John feeds the cows a lavish meal of premium gourmet

hay. He then records the number of bales on the next line of his

expense notebook.

When tax time comes, FJ realizes that he neglected to record the

dates for the hay feedings. He must calculate a number of different

totals of successive hay feedings in order to solve the puzzle of

which feedings went with which month of expenses.

FJ has created a dataset with N (4 <= N <= 500) days conveniently

numbered 1..N of hay bale counts H\_i (1 <= H\_i <= 1,000). He has

an additional Q (1 <= Q <= 500) queries -- each query is a pair of

integers S\_j and E\_j (1 <= S\_j <= E\_j <= N) that denote that start

and end indices of some hay bale counts. Your job is to sum the

hay bale counts for the days S\_j..E\_j (inclusive) and report one

sum for each query.

PROBLEM NAME: hayexp

INPUT FORMAT:

\* Line 1: Two space-separated integers: N and Q

\* Lines 2..N+1: Line i+1 contains a single hay bale count for day i:

H\_i

\* Lines N+2..N+Q+1: Line j+N+1 describes query j with a pair of

integers: S\_j and E\_j

SAMPLE INPUT (file hayexp.in):

4 2

5

8

12

6

1 3

2 4

INPUT DETAILS:

Four days; two queries. Bale counts: 5, 8, 12, and 6. Count

Days 1..3 and 2..4.

OUTPUT FORMAT:

\* Lines 1..Q: Line j of the output file contains a single integer that

is the sum of the hay bale counts for days S\_j through E\_j

SAMPLE OUTPUT (file hayexp.out):

25

26

OUTPUT DETAILS:

Days: 1 2 3 4

Counts: 5 8 12 6

query 1..3: 5 + 8 + 12 = 25

query 2..4: 8 + 12 + 6 = 26

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*