



# South African Computer Olympiad

## Final Round

### Day 2



## Overview

Author	Keegan Carruthers-Smith	Marco Gallotta	Carl Hultquist
Problem	parrots	seen	search
Source	parrots.java parrots.py parrots.c parrots.cpp parrots.pas	seen.java seen.py seen.c seen.cpp seen.pas	N/A
Input file	stdin	seen.in	search.in
Output file	stdout	seen.out	search.out
Time limit	1 second	1 second	N/A
Number of tests	10	10	10
Points per test	10	10	10
<b>Total points</b>	<b>100</b>	<b>100</b>	<b>100</b>

The maximum total score is 300 points.



# South African Computer Olympiad

## Final Round

### Day 2



## Living Dead Parrot

### Author

Keegan Carruthers-Smith

### Introduction

The Pet Shop Owner has a lot of cages with parrots in them at the back of his store. The owner needs to find out which cages have parrots that are alive, but the door to the back of his store is locked. The cages are lined up in the back and each cage is attached to a rope. Each rope goes through a pulley system to the front of the store. Pulling these ropes from the front of the store causes the cages to rattle in the back. A parrot will squawk if its cage is rattled.

The pulley system causes a range of cages that are adjacent to each other to rattle. The Pet Shop Owner has been using this pulley system for many years and has learned how to pull the ropes such that he can rattle any range of cages he wants.

### Task

You must work out which cages have parrots that are alive in them. The owner can work out how many live parrots are in the range  $a$  to  $b$  by listening to the number of squawks. The owner's time is limited, so he can only pull on the pulley up to  $M$  times. The cages are numbered from 1 to  $N$ . There are  $K$  live parrots.

### Input/Output

You must read input from standard input (as you would normally read input from the keyboard). You should begin by reading in a line containing the space-separated integers  $N$  and  $M$ , the number of cages and the number of pulls the shop owner can make respectively.

From here you have 2 possible responses, which you write to standard output (as you would normally write to the console):

- **Range Query:**  $Q\ a\ b$
- **Answer:**  $A\ S$

The range query will return the number of parrots that are alive in the range  $a$  to  $b$ , including  $a$  and  $b$ .

You must remember to flush the output stream after every query, to ensure that the evaluator receives it.

- Pascal: `flush(output);`
- C stdio: `fflush(stdout);`
- C++ iostreams: `cout << flush;`
- Java: `System.out.flush();`
- Python: `sys.stdout.flush()`  
# Requires 'import sys'

The answer,  $S$ , is a space-separated sequence of 0's and 1's representing whether the parrot in each cage is alive (1) or dead(0).

You can only make up to  $M$  range queries, and give one answer. After you have given an answer your program must terminate without producing further output.

### Example

For the sake of clarity in this example, we shall ignore the constraints on the number of parrots that are alive,  $K$ , as well as the number of questions allowed,  $M$ .

See Table 1 for an example of the interaction with the evaluator when  $N = 20$  and  $M = 10$ .

Input	Output
20 10	
0	Q 5 10
2	Q 11 20
2	Q 17 19
2	Q 18 19
1	Q 1 5
1	Q 3 3
	A 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 0

Table 1: Sample interaction.  $N = 20$ ,  $M = 10$  and cages 3, 18 and 19 contain alive parrots.

### Testing

During the contest, you may wish to test your program using some set of dead and living parrots. The format of the file that you should upload to the server to test your program with is as follows. The first line should contain a single integer  $N$ , the number of cages. The next line should contain  $N$  space-separated integers, each of which



# South African Computer Olympiad

## Final Round

### Day 2



is either 0 or 1 to indicate whether the parrot is dead or alive respectively.

Please take note of the constraint below on  $K$ , the number of live parrots — if your test-case has no live parrots or too many live parrots, then the test will fail.

### Constraints

- $100 \leq N \leq 100000 = 10^5$
- $M = \lceil \log_2(N) \times \frac{N}{100} \rceil$
- $1 \leq K \leq \frac{N}{100}$

Note that  $\lceil x \rceil$  is the smallest integer greater than or equal to  $x$ .

### Time limit

1 second.

### Scoring

If you ask more than  $M$  range queries or output a parrot as alive when it is dead you will receive 0% for the test case. You will also receive 0% if you make any range query that does not satisfy  $1 \leq a \leq b \leq N$ , if any of your outputs are formatting incorrectly (e.g., outputting values other than 0's and 1's in S).

You will receive 100% for a test case if you guess the positions of all parrots correctly.

Otherwise, if you correctly identify the positions of  $D$  live parrots, you will get  $60 \times \frac{D}{K}$ %.



# South African Computer Olympiad

## Final Round

### Day 2



## How Not To Be Seen

### Author

Marco Gallotta

### Introduction

The How Not To Be Seen competition has returned this year, and the contestants have become very skillful over the years. This is no good for the organisers and so they have added some spies to the garden to help find the good contestants. The trouble with spies is that in order to blend in with the real contestants they won't be able to tell each other apart very easily.

To allow for more than one spy they have to come up with a strategy. They decide that they'll take positions such that if they were to look North, West, South and East they would see at most one other spy in total in all four directions. If one person is behind another, the spy will only see the closest person in that direction as the others will be hidden behind him. With this strategy, if a spy sees more than one person from looking in the four directions then he knows he's found a contestant.

### Task

The spies want to know how much choice they have in choosing their positions. Given the dimensions of the rectangular  $N \times M$  garden find out the number of different choices  $K$  spies could make when choosing the entire group's positions. Note that it makes no difference to the spies who gets placed where and therefore two choices are *not* considered different if the only difference is spies swapping positions.

Since the number of choices can get unmanageably large, they only ask for the number modulo 1000001 (one million and one) — that is, the remainder when the number is divided by 1000001. The competition organisers claim they have a secret formula for obtaining the full number given this reduced answer, but that's not something that should concern you.

### Example

Suppose the garden is of size  $2 \times 3$  and there are 3 spies. There are six possible positions they could take, which are shown in Figure 1.

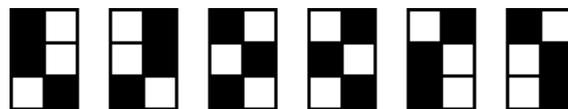


Figure 1: Example showing all possible positions. A black square represents a spy.

### Input (seen.in)

The single line of input contains three space-separated integers,  $N$ ,  $M$  and  $K$ .

### Sample input

```
2 3 3
```

### Output (seen.out)

The output should contain a single integer, the total number of different choices the spies could make modulo 1000001.

### Sample output

```
6
```

### Constraints

- $1 \leq N, M \leq 100$
- $1 \leq K \leq 100$

Additionally, in 50% of the test-cases:

- $1 \leq NM \leq 30$

### Time limit

1 second.

### Scoring

A correct solution will score 100%, while an incorrect solution will score 0%.



# South African Computer Olympiad

## Final Round

### Day 2



# The Crimson Permanent Assurance

## Author

Carl Hultquist

## Introduction

Financial districts beware! A group of disgruntled elderly office clerks have overthrown their rulers, and launched their pirate ship “The Crimson Permanent Assurance”. They have already crushed all their competitors in London, and are preparing to set sail and conquer the rest of the modern world. We know that one day they will seal their own fate by sailing off the edge of the world, but before that they will be stopping in Cape Town and, as chief executive of a local insurance firm, you need to find a way of stopping them.

Fortunately, being commandeered by elderly people the ship’s crew are quite fond of puzzles. They particularly enjoy word search puzzles, and whenever they conquer a new city they give all the local insurance firms the chance to compete in solving a word search puzzle. They promise to spare the lives of the firm that finds the best solution, whilst the losing firms must perish and receive the full wrath of their vicious accounting practices.

## Task

Given a number of word search puzzles, you must find the best scoring solution possible and so save the lives of your firm. You will be given all of the word search puzzles, and instead of handing in a program you will hand in only your solutions. You will receive a score for each puzzle based on the best solution found by all the contestants, as described in the section on scoring below.

A word search puzzle consists of a grid of letters and a list of words. The goal is to locate consecutive groups of adjacent letters within the grid that form one of the words in the list. Letters can be grouped either vertically (upwards or downwards), horizontally (forwards or backwards) or diagonally, and no letter in the grid may be used to form more than one word. There is no restriction on the number of times that any word from the list may be used on the grid.

Given a grouping of the letters into words from the list, the score of a grouping is calculated as:

$$n + \sum_{i=1}^n \text{length}(\text{word}_i)$$

that is, the number of words found plus the length of every word found. If a word from the list appears more than once in the grouping on the grid, then it should be counted multiple times in the score calculation (see the example below for clarification).

## Example

Suppose the grid looks like that shown in Figure 2, and the list of words being searched for is as given in

Table 2:

H	H	K	F	M
U	E	W	R	E
I	L	L	L	A
P	P	R	L	M
E	M	C	V	O

Figure 2: An example letter grid

HELLO
ME
ALL
I
HELP
WE

Table 2: The list of words to be searched for in Figure 2

One possible solution is shown in Figure 3, where the words found are circled by dashed lines.

In this case, we found two occurrences of the word ME, and one occurrence of each of I and HELLO. This gives a total of  $4 + (2 + 2 + 1 + 5) = 14$  points.

## Input (search.in)

The first line of each input file will contain two space-separated integers,  $N$  and  $M$ , which are the number of rows and columns respectively in the letter grid. The next  $N$  lines will each contain exactly  $M$  uppercase letters describing the grid. The following line will contain a single



# South African Computer Olympiad

## Final Round

### Day 2

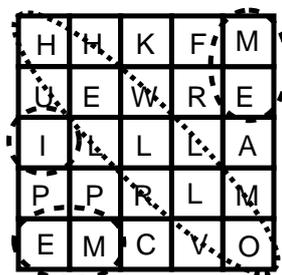


Figure 3: A possible solution for the letter grid and word list shown in Figure 2 and Table 2.

integer,  $W$ , the number of words in the list. The next  $W$  lines will each contain a single uppercase word, giving the words in the list.

#### Sample input

```
5 5
HHKFM
UEWRE
ILLLA
PPRLM
EMCVO
6
HELLO
ME
ALL
I
HELP
WE
```

#### Output (search.out)

Each of your output files should be formatted as follows. The first line must contain a single integer,  $K$ , which is the number of words found in the grid. The next  $K$  lines must each contain four space-separated integers  $R_S$ ,  $C_S$ ,  $R_E$  and  $C_E$ .  $R_S$  and  $C_S$  give the row and column respectively in the grid at which the first letter of the word can be found, and  $R_E$  and  $C_E$  give the row and column respectively in the grid at which the last letter of the word can be found. The rows in the grid are numbered from 1 to  $N$ , and the columns from 1 to  $M$ .

#### Sample output

```
4
1 1 5 5
5 2 5 1
1 5 2 5
3 1 3 1
```

#### Constraints

- $1 \leq N, M \leq 250$
  - $1 \leq W \leq 12000$
- Additionally, in 50% of the test-cases:
- $1 \leq N, M \leq 20$
  - $1 \leq W \leq 100$

#### Time limit

N/A.

#### Scoring

For a given test-case, if any of the following is true

- you did not submit an output file for the test-case;
- your output file does not adhere to the output format specified above;
- the starting and ending co-ordinates you give for a word do not specify a horizontal, vertical or diagonal placement;
- the co-ordinates you give for a word do not lie within the grid;
- the co-ordinates you give for a word do not match up to a word in the list

then you will score 0% for that test-case.

Otherwise, your score will be calculated based on the number of points earned by your solution in comparison to the maximum number of points earned by any contestant for this test-case.

More specifically, if your solution scores  $S$  points and the maximum number of points earned by any contestant for this case is  $B$ , then you will score

$$\max\left(900\frac{S}{B} - 800, 10\right)\%$$