Indian National Olympiad in Informatics, 2011

Time: 3 hours

22 January, 2011

Instructions

(a) You will have to return this question paper at the end of the examination with relevant parts filled out.

(b) There are two questions. You have to write working programs in Pascal, C or C++ to solve each of these questions.

- *Only your source code will be submitted for evaluation. Your program will be recompiled and run on the evaluation computer.*

- *Make sure your C/C++ programs compile with the GNU C compiler (**dev-cpp** or **djgpp**, under Windows). Programs written using Turbo C++ may not compile and run in the evaluation environment and may hence lose marks.*
  If you work in C/C++, make sure you do not write #include <conio.h> in your program or use any functions defined in conio.h. If you do so, your program will not compile and you will get no marks.

(c) At the end of each question, there is a space to indicate the location of the source code file for your solution. Please fill up this information without fail. Otherwise, your solution cannot be evaluated.

(d) All input for your programs will come from the keyboard. All output from your programs should be written to the screen.

(e) Please fill out your contact details on the reverse of this page as completely as you can. Use your serial number as reported in the ZIO/ZCO results page as your roll number for INOI. If you qualified under multiple categories, use the ZIO serial number as your roll number. Ask your centre supervisor if you do not know your roll number.

(f) At the beginning of each program, include a comment with your name, roll number and the problem number. For instance, if your name is Sachin Tendulkar, your roll number is 107 and the problem you are writing a C/C++/Pascal program for is Problem 1, add a comment of the form

```
/*  Sachin Tendulkar, 107, Problem 1 */
```

at the top of your file.

This will help us if files accidentally get mixed up when being sent from the exam centres to the evaluation centre.
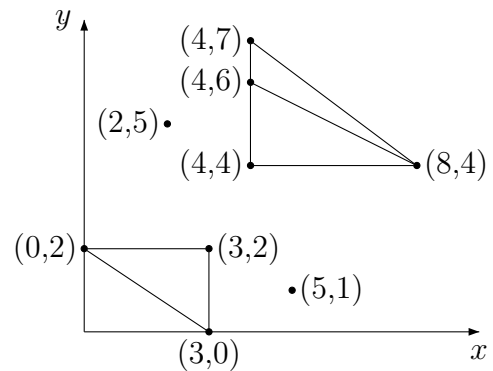
. . . P.T.O.

Contact details	
Roll Number:	
Name:	
Home Address: PIN Code	
Phone Number(s): (with STD code)	
Email address:	

Question 1   Counting triangles

You are given N points with non-negative integer x and y-coordinates. Your task is to count the number of right-angled triangles that can be formed using these points as vertices such that one side of the triangle is parallel to the x-axis and another side is parallel to the y-axis.

For example, using the nine points on the right, we can form three such right-angled triangles.

Input format

The first line of input is an integer N, denoting the number of points. This is followed by N lines of input, each containing two integers, separated by a space, specifying the x and y coordinate of one point. You may assume that each coordinate x and y lies in the range 0 to 10^5.

Output format

Your output should be a single integer, denoting the number of right-angled triangles of the stated type that can be formed from the input points.

Since this number may exceed the maximum value representable in an `int`, *you should report your answer modulo* 10^4. *In other words, output the remainder obtained by dividing the number of triangles by* 10^4.

Test Data

All N points are always distinct. You may assume that $1 \leq N \leq 10^5$. In 30% of the cases, $N \leq 3000$.

Example

Here is the sample input and output corresponding to the example above.

```
9
0 2
4 6
3 0
2 5
5 1
3 2
4 7
8 4
4 4
```

```
3
```

Time and memory limits

The time limit for this task is 1 second. The memory limit is 32 MB.

Note: Your program should not print anything other than what is specified in the output format. Please remove all diagnostic print statements before making your final submission. A program with extraneous output will be treated as incorrect!

Important

Indicate the location of the source code file for your solution to Question 1 in the box below. *If you fail to do this, your solution cannot be evaluated!*

Source file:

Question 2   IOI Training Camp 20xx

We are well into the 21st century and school children are taught dynamic programming in class 4. The IOI training camp has degenerated into an endless sequence of tests, with negative marking. At the end of the camp, each student is evaluated based on the sum of the best *contiguous segment (i.e., no gaps)* of marks in the overall sequence of tests.

Students, however, have not changed much over the years and they have asked for some relaxation in the evaluation procedure. As a concession, the camp coordinators have agreed that students are allowed to drop upto a certain number of tests when calculating their best segment.

For instance, suppose that Lavanya is a student at the training camp and there have been ten tests, in which her marks are as follows.

Test	1	2	3	4	5	6	7	8	9	10
Marks	6	-5	3	-7	6	-1	10	-8	-8	8

In this case, without being allowed to drop any tests, the best segment is tests 5–7, which yields a total of 15 marks. If Lavanya is allowed to drop upto 2 tests in a segment, the best segment is tests 1–7, which yields a total of 24 marks after dropping tests 2 and 4. If she is allowed to drop upto 6 tests in a segment, the best total is obtained by taking the entire list and dropping the 5 negative entries to get a total of 33.

You will be given a sequence of N test marks and a number K. You have to compute the sum of the best segment in the sequence when upto K marks may be dropped from the segment.

Input format

The first line of input contains two integers N and K, where N is the number of tests for which marks will be provided and K is the limit of how many entries may be dropped from a segment.

This is followed by N lines of input each containing a single integer. The marks for test i, $i \in \{1, 2, \ldots, N\}$ are provided in line $i+1$.

Output format

The output is a single number, the maximum marks that can be obtained from a segment in which upto K values are dropped.

Test Data

You may assume that $1 \leq N \leq 10^4$ and $0 \leq K \leq 10^2$. The marks for each test lie in the range $[-10^4 \ldots 10^4]$. In 40% of the cases you may assume $N \leq 250$.

Example

Here are the sample inputs and outputs corresponding to the examples above.

Sample input 1	Sample output 1	Sample input 2	Sample output 2
10 2	24	10 6	33
6		6	
-5		-5	
3		3	
-7		-7	
6		6	
-1		-1	
10		10	
-8		-8	
-8		-8	
8		8	

Time and memory limits

The time limit for this task is 2 seconds. The memory limit is 64 MB.

Note: Your program should not print anything other than what is specified in the output format. Please remove all diagnostic print statements before making your final submission. A program with extraneous output will be treated as incorrect!

Important

Indicate the location of the source code file for your solution to Question 2 in the box below. *If you fail to do this, your solution cannot be evaluated!*

Source file:

4