**Mark The Answer**

Attempted by: **2988**

/

Accuracy: **94%**

/

Maximum Score: **20**

/

124 Votes

Tag(s):

Data Structures, Easy

**PROBLEM**

**EDITORIAL**

**MY SUBMISSIONS**

**ANALYTICS**

Our friend Monk has an exam that has quite weird rules. Each question has a difficulty level in the form of an Integer. Now, Monk can only solve the problems that have difficulty level less than **X** . Now the rules are-

* Score of the student is equal to the maximum number of answers he/she has attempted without skipping a question.
* Student is allowed to skip just **"one"** question that will not be counted in the continuity of the questions.

**Note**- Assume the student knows the solution to the problem he/she attempts and always starts the paper from first question.

Given the number of Questions, *N* ,the maximum difficulty level of the problem Monk can solve , XX ,and the difficulty level of each question , AiAican you help him determine his maximum score?

**Input Format**  
First Line contains Integer NN , the number of questions and the maximum difficulty XX Monk can solve.  
Next line contains NN integers, AiAi denoting the difficulty level of each question.

**Output Format**  
Maximum score Monk can achieve in the exam.

**Constraints**

* 1≤N≤1051≤N≤105
* 1≤X≤1091≤X≤109
* 1≤Ai≤1091≤Ai≤109

**SAMPLE INPUT**

7 6

4 3 7 6 7 2 2

**SAMPLE OUTPUT**

3

**Explanation**

In this example, maximum difficulty = 6, Monk solves question 0 and 1, but skips the question 2 as **A[2]>6**. Monk then solves the question 3 , but stops at 4 because **A[4]>6** and question 2 was already skipped. As 3 questions **(0,1 and 3)** were solved and 2 questions **(2 and 4)** have been skipped, therefore we print **"3"**.

**Time Limit:**1.0 sec(s) for each input file.

**Memory Limit:**256 MB

**Source Limit:**1024 KB

**Marking Scheme:**Marks are awarded when all the testcases pass.

**Allowed Languages:**C, C++, Clojure, C#, D, Erlang, F#, Go, Groovy, Haskell, Java, Java 8, JavaScript(Rhino), JavaScript(Node.js), Lisp, Lisp (SBCL), Lua, Objective-C, OCaml, Octave, Pascal, Perl, PHP, Python, Python 3, R(RScript), Racket, Ruby, Rust, Scala, Scala 2.11.8, Swift, Visual Basic

<https://www.hackerearth.com/practice/data-structures/arrays/1-d/practice-problems/algorithm/mark-the-answer-1/>

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

namespace ConsoleApplication2

{

class Program

{

static void Main(string[] args)

{

string[] input = Console.ReadLine().Split(' ');

int n = int.Parse(input[0]);

int x = int.Parse(input[1]);

int[] a = Array.ConvertAll(Console.ReadLine().Split(' '), e => int.Parse(e));

//int n = 7;

//int x = 6;

//int[] a = { 4, 3, 7, 6, 7, 2, 2 };

int cont = 0;

bool sobrepasa = false;

for (int i = 0; i < a.Length; i++)

{

if (a[i] > x && !sobrepasa)

{

cont--;

sobrepasa = true;

}

else if (a[i] > x && sobrepasa)

{

break;

}

cont++;

}

Console.WriteLine(cont);

Console.ReadLine();

}

}

}