Guard wall sequences

Input:

outposts∈N，guards∈N，num[1...guards] $\in N^{guards}$

output:

count∈N

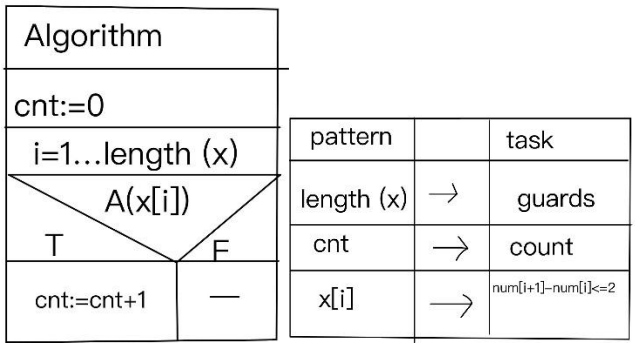
precondition:

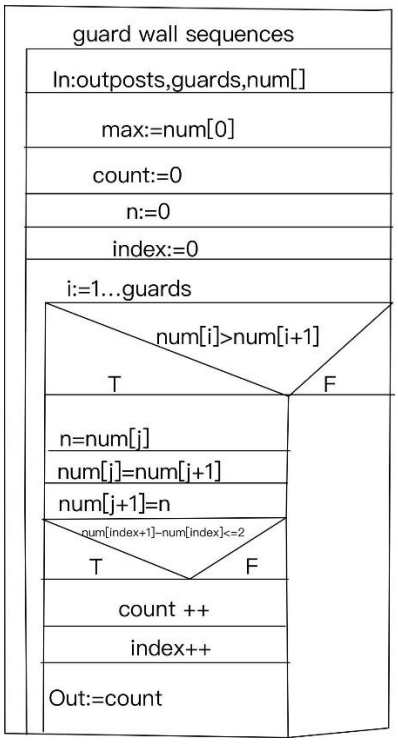
1<outposts<100，1<guards<outposts，$\forall[i]$(1<i<guards):1<num[i]<outposts

Postcondition:

Count:=$\sum_{i=1}^{guards} 1$

(num[i+1]-num[i]<=2)





Code:

```
using System;
namespace 笑死 115
{
    internal class Program
    {
        static void Main(string[] args)
        {
            string input = Console.ReadLine();
            int outposts = Convert.ToInt32((input.Split(" ")[0]));
            int guards = Convert.ToInt32((input.Split(" ")[1]));
            int[] num = new int[guards];
            int count = 0;
            for (int i = 0; i < guards; i++)
            {
                num[i] = Convert.ToInt32((Console.ReadLine()));
                int max = num[0];
            }
            for (int i = 0; i < num.Length - 1; i++)
            {
                for (int j = 0; j < num.Length - 1; j++)
                {
                    if (num[j] > num[j + 1])
                    {
                        int n = num[j];
                        num[j] = num[j + 1];
                        num[j + 1] = n;
                    }
                }
            }
            int Index = 0;

            while(Index<num.Length - 1)
            {
                while(Index < num.Length - 1 && num[Index + 1] - num[Index] <=2 )
                {
                    Index++;
                }
                count++;
                Index++;
            }
            Console.WriteLine(count);
            Console.WriteLine();
        }
    }
}
```