Zhanabaikyzy Karakat

Task 4

You are given an integer N and an array of N integers. The data is a cubed profile of a mountain range in a flat world (Figure 1). Let the goat move through the mountains from left to right. Because of the wind and sun, the grass grows only on the southern side of the mountains (let it be on the right for us). Therefore, if in a step to the left to the right the goat rises higher, it will lose one energy, if it goes down, it will receive one energy, if the level does not change, then the energy level of the goat will not change. Where will he get the most energy benefit? Its energy indicator on a segment can have negative values, because he has a sufficient initial supply of energy.

We have array N\*N,

case when it goes higher

if(a[i][j]<a[i+index][j+height]) then

energy--;

it loses energy

case when it goes lower

if(a[i][j]>a[i+index][j+height]) then

energy++;

it gets energy

case when it moves horizontally

if(a[i][j]==a[i+index][j+height])

nothing is changed.

Task 6

You are given an integer N and an array of N integers. The data is a cubed profile of a mountain range in a flat world (Figure 1). Let the camel move through the mountains from left to right. We assume that there is water only in ravines, that is, at points of local minimum. For each step, the camel spends 1 unit of water, if at the same time it rises higher, then it spends another 1 unit of water. At the points of the local minimum, it is filled with 10 units of water. Find where it will get the most water benefit. It can be assumed that he will never be dehydrated, because it has an initial hump headroom.

We have array N\*N

For(int i = 0; i < array.length; i++)

step--;

if(array[i][j] < array[i+length][j+height]);

step -=2;

and where local min, then it is filled with 10 units of water:

boolean min = false;

if(array[i][j] < array[i+length][j+height])

min = true;

rising = rising + 10;

Task 7

You are given an integer N and an array of N integers. These numbers represent a graph of the function (Figure 2) with integer values ​​in integer coordinates. In data analysis, it is often necessary to find cases of stability of some indicator, i.e. minimal changes over time. Find the longest interval in which the values ​​of the function change by no more than 3 (the difference between the maximum and minimum).

def longestInterval():

#we input length of array

n = int(input("input your N: "))

arr = []

#here input numbers in the array

for i in range(n):

numb: int = int(input("input Number for array: "))

arr.append(numb)

maxChangings = 0

# then define for longest interval

for i in range(n - 1):

intervalChangings = abs(arr[i + 1] - arr[i])

if 3 >= intervalChangings > maxChangings:

maxChangings = intervalChangings

# in the end we print The longest interval as the result

print("Array: ", arr)

print("The longest interval: ", "Defined Interval is bigger than 3" if maxChangings == 0 else maxChangings)

longestInterval()