Algorithms Design

Project 2 - Dynamic Programing

4012 - Semester

Overview

Tim, a good-natured boy, grew restless from being confined indoors for an extended period. One day, while playing with his toys, he noticed his old puzzle pieces interlocked. Determined to find a way to separate them while keeping boredom at bay, he devised a game. He discovered that some puzzle pieces of the same color were arranged in a consecutive row.

Excited by this observation, **he challenged himself to earn points by separating groups of k same-colored pieces in a row, rewarding himself with k^2 for each successful separation**. Motivated by his gaming idea, Tim enthusiastically began untangling the puzzle pieces to achieve the highest possible score.

Input

You will give an array of numbers that each number denotes the color of a magnet.

Output

Print one number which is the maximum total points Tim can get from separating puzzle pieces interlocked.

Examples

Input:

    puzzle pieces = [1, 1, 1]

Output:

    3*3 = 9

    because there is one choice separating all three magnets to get maximum points.

Input:

    puzzle pieces = [1, 3, 2, 2, 2, 3, 4, 3, 1]

Output:

    3*3 = 9 remaining puzzle pieces: [1, 3, 3, 4, 3, 1]

1*1 = 1 remaining puzzle pieces: [1, 3, 3, 3, 1]

3*3 = 9 remaining puzzle pieces: [1, 1]

2*2 = 4 remaining puzzle pieces: []

summation: 9 + 1 + 9 + 4 = 23

Hint: For each group of colors Tim has two choices: either he can remove it or maintain part of it for the future to make a larger group.

Goal

1. **Design Dynamic Programming Algorithm that can help Tim for obtaining maximum total points from separating puzzle pieces interlocked.**

Attention!

1. All of the projects will be tested for similarities by a coded script, So if we find an obvious similarity between 2 or more projects **all** of them will get **-100** points.
2. You are free to choose any programming language you desire in this problem.
3. You have to write a document file and **explain deeply** what you've done in your code and the algorithm you wrote, Explain the algorithm **before** you begin talking about the code.
4. It's necessary to present your implementation. There is no score for just uploading the documentation.