

Exercise 1

You are given a set of 2-dimensional points representing biological samples.

The goal is to perform **Agglomerative Hierarchical Clustering** using the following strategy:

- initially, each point forms its own cluster;
- at each iteration:
 1. find the two closest clusters;
 2. merge them into a new cluster;
 3. recompute the distances between clusters.

Use the **single-linkage criterion**, meaning that the distance between two clusters is defined as:

the minimum distance between any pair of points belonging to the two clusters.

Use the Euclidean distance.

Input

```
points = {  
    "A": (1, 1),  
    "B": (2, 1),  
    "C": (8, 8),  
    "D": (9, 8),  
    "E": (5, 5)  
}
```

Example of the algorithm behavior

Initially, each point is a separate cluster:

```
{A} {B} {C} {D} {E}
```

The algorithm then repeatedly merges the two closest clusters.

For example:

- Let us suppose that A and B are very close, they are merged first:

```
{A, B} {C} {D} {E}
```

- then C and D are merged:

`{A, B} {C, D} {E}`

The process continues until only one cluster remains.

Task

Write pseudocode for an algorithm that performs Agglomerative Hierarchical Clustering on the input points.

Your pseudocode must clearly specify:

1. how clusters are initialized;
2. how the closest clusters are selected;
3. how clusters are merged;
4. how distances are recomputed;
5. when the algorithm terminates.

Then, give the time complexity and space complexity using Big-O notation.

Notes

You may assume the availability of:

`distance(p1, p2)`

which computes the Euclidean distance between two points.

You may also use a matrix or table to store pairwise distances between clusters.

Exercise 2

You are given an integer n representing the length of a synthetic protein chain.

The chain must be assembled using building blocks of fixed lengths:

- a small protein unit, which occupies exactly 1 position;
- a large protein unit, which occupies exactly 2 consecutive positions.

The order in which the building blocks are placed matters.

For example:

- placing two small units followed by one large unit is considered different from placing one large unit followed by two small units.

Your task is to compute the number of distinct ways to assemble a protein chain of length n .

Example

Input

$n = 4$

Intermediate values

$ways[0] = 1$

$ways[1] = 1$

$ways[2] = 2$

$ways[3] = 3$

$ways[4] = 5$

Output

5

Hint

Consider how a chain of length n can be obtained:

- by appending a small protein unit to a valid chain of length $n - 1$, or by appending a large protein unit to a valid chain of length $n - 2$.
- Note that we are counting the number of possible assemblies, not the number of building blocks used.

Task

- Write pseudocode for an algorithm that computes the number of distinct ways to assemble a protein chain of length n .
- Then, give the time complexity and space complexity of your algorithm using Big-O notation.

Exercise 3

A researcher has to complete a set of laboratory tasks.

Each task has a basic effort cost, but the later a task is performed, the more expensive it becomes because the researcher becomes progressively more tired.

If the researcher has already completed j tasks and then performs a task with effort cost e , the actual cost of that task is:

$$e \times (2j + 1)$$

The researcher can perform the tasks in any order.

Your task is to find the minimum total effort required to complete all tasks.

Example

Input

$tasks = [4, 2, 7]$

If the tasks are performed in the order:

$7, 4, 2$

then the total effort is:

$$7 \times 1 + 4 \times 3 + 2 \times 5 = 29$$

Output

29

Task

Write pseudocode for an algorithm that computes the minimum total effort required to complete all tasks.

Then:

- explain which algorithmic strategy you are using and justify why this strategy gives the minimum total effort;
- give the time complexity and space complexity of your algorithm using Big-O notation.