SENSOR NETWORK

V Pavani:21b01a12i4:IT
B Reshma:21b01a1216:IT
P Sri Vyshnavi:21b01a5484:AIDS
L Parimala Prasanna:22b05a5408:AIDS
S Kalyani:21b01a0251:EEE

SVECW

March 25,2023

Introduction about the project

- A subset of sensors where each pair of sensors can communicate directly with each other (distance at most d). The goal is to choose as many sensors as possible, given their current locations represented as points in a two-dimensional plane, using Euclidean distance to measure the distance between two points.

Approach

- We used command line arguments to provide input
- math and sys modules are imported
- User defined functions are used
- We used Euclidean distance to measure the distance between two sensors

Learnings

We have learnt about how to pass command line arguments in VS code and ability to work collaboratively and productively with others.How to manage our time effectively and meet deadlines. We have learnt the usage of GitLab and LaTeX.

Challenges

- Integration of code is difficult.
- Indentation errors,data type errors.
- We faced difficulty while using command line arguments in visual studio code

STATISTICS

- The Python code has overall 49 lines.
- The code has three user-defined functions. They are:
    1. command-input()
    2. distance()
    3. subset()
- Built-in Modules:
    1. sys
    2. math

Demo/Screenshots

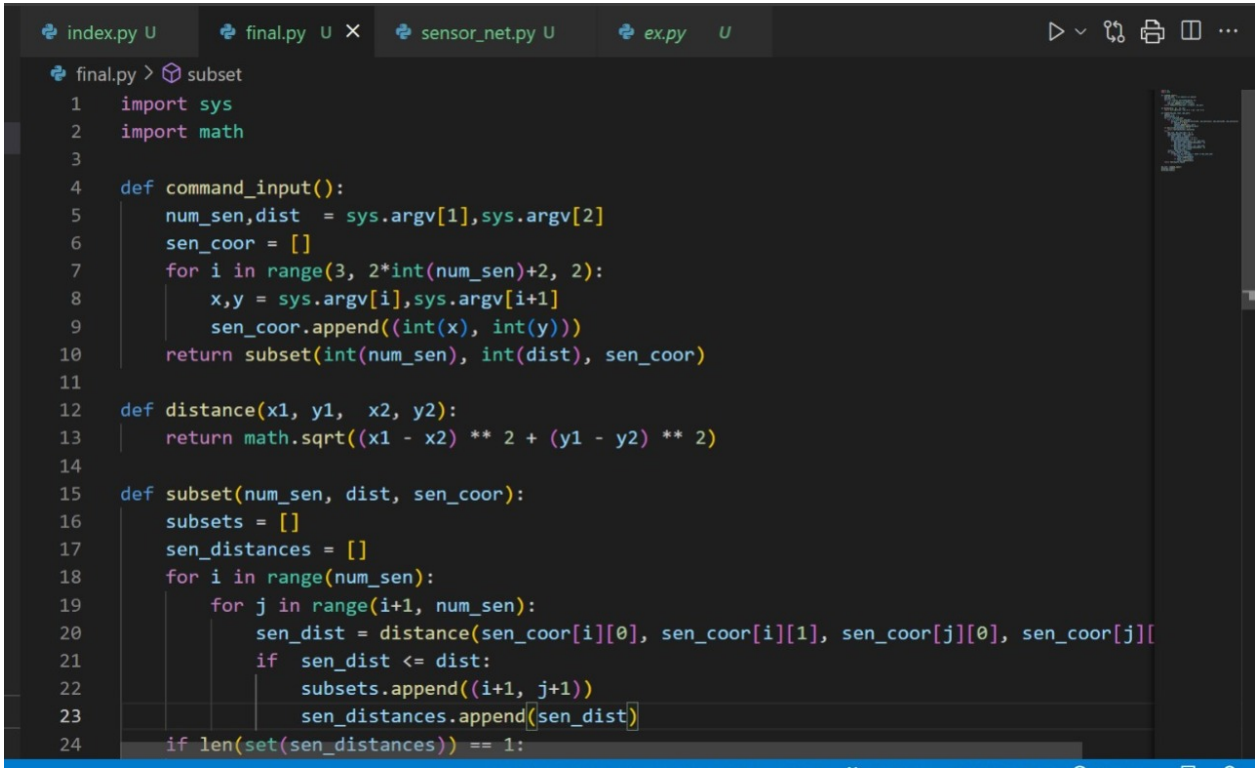

Figure 1: Code of the project

Demo/Screenshots

final.py > ...

```
    if len(set(sen_distances)) == 1:
        return len(subsets[0]),subsets[0]
    else :
        max_conn, max_conn_sens = 0, 0
        sens_connections = [0] * num_sen
        for sens1, sens2 in subsets:
            sens_connections[sens1 - 1] += 1
            sens_connections[sens2 - 1] += 1
            if sens_connections[sens1 - 1] > max_conn:
                max_conn = sens_connections[sens1 - 1]
                max_conn_sens = sens1
            if sens_connections[sens2 - 1] > max_conn:
                max_conn = sens_connections[sens2 - 1]
                max_conn_sens = sens2
        result = [max_conn_sens]
        for sens1, sens2 in subsets:
            if sens1 == max_conn_sens or sens2 == max_conn_sens:
                if sens1 not in result:
                    result.append(sens1)
                if sens2 not in result:
                    result.append(sens2)
    return len(result),result

max_sub = command_input()
print(max_sub[0])
print(max_sub[1])
```

Figure 2: Code of the project

Demo/Screenshots

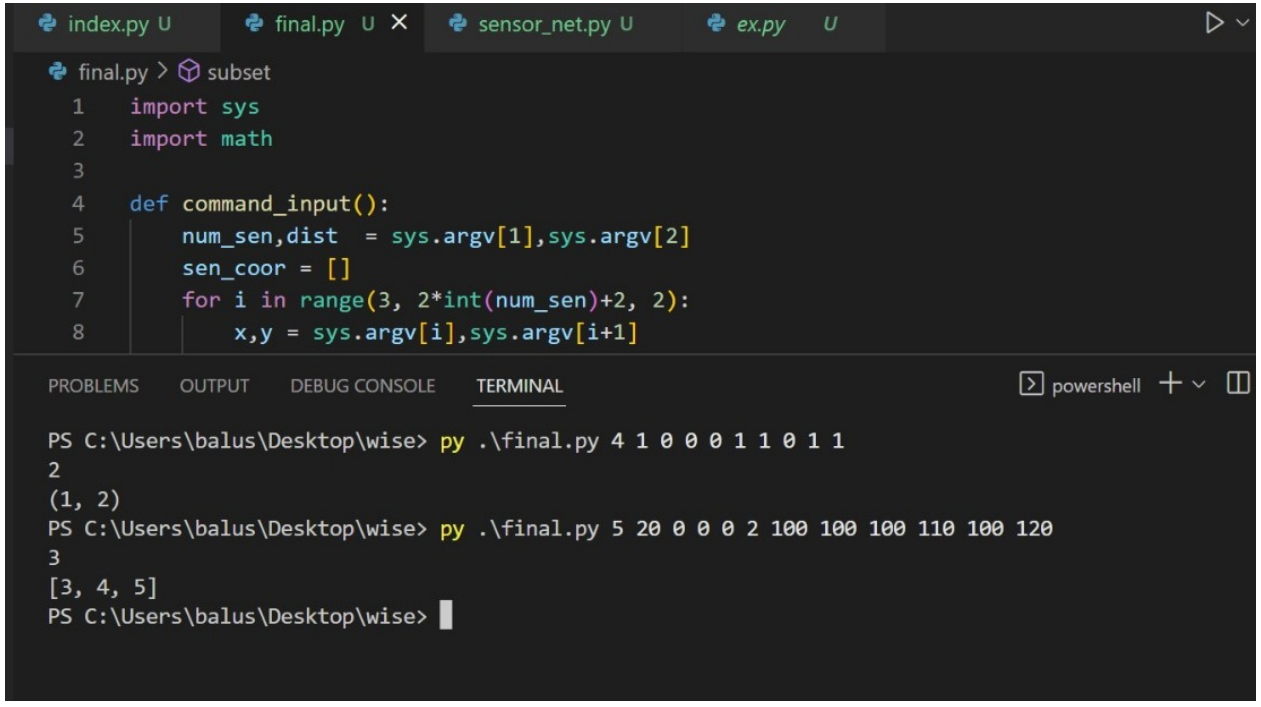

Figure 3: demo output

Thank You