

Name - Yash Lakhtariya  
Enrollment number - 21162101012  
Branch - CBA      Batch - 51  
AAD Practical 2

**Institute of Computer Technology**  
**B. Tech Computer Science and Engineering**

**Sub: Algorithm Analysis and Design**

**Practical 2**

Let us suppose that you are having an array containing both positive and negative numbers. Given the numbers you are supposed to find 2 such elements such that the sum of those numbers is closest to zero.

Sample Input 1

15, 5, -20, 30, -45

Sample Output 1

15, -20

Explanation 1 : In all the comparison, the sum of 15 and -20 is smallest amount among all other comparison.

**Code:**

```
from YSL_io import *  
  
def closetozero(y):  
    if len(y) > 2:  
        sum = y[0] + y[1]  
        ans = [y[0], y[1]]  
        alt = []
```

Name - Yash Lakhtariya  
Enrollment number - 21162101012  
Branch - CBA      Batch - 51  
AAD Practical 2

```
for i in range(len(y)):
    for j in range(i + 1, len(y)):
        curr = y[i] + y[j]
        if abs(curr) ≤ abs(sum):
            if abs(curr) < abs(sum):
                sum = curr
            ans = [y[i], y[j]]
        else:
            sum = curr
        alt.append([y[i], y[j]])
    return ans, alt

y = [25, 15, 5, -20, 30, -45, 50, -55]
ans, alt = closetozero(y)
alt.pop(0)

if len(alt) == 0 :
    print('\nThe required numbers whose sum is closest to zero are : ',
          end='')
    printGRN(f'{ans[0]} and {ans[1]}')
else :
    print('\nThe required numbers whose sum is closest to zero are : ',
          end='')
    printGRN(f'{ans[0]} and {ans[1]}', end=', ')
```

Name - Yash Lakhtariya  
Enrollment number - 21162101012  
Branch - CBA      Batch - 51  
AAD Practical 2

```
for elem in alt:

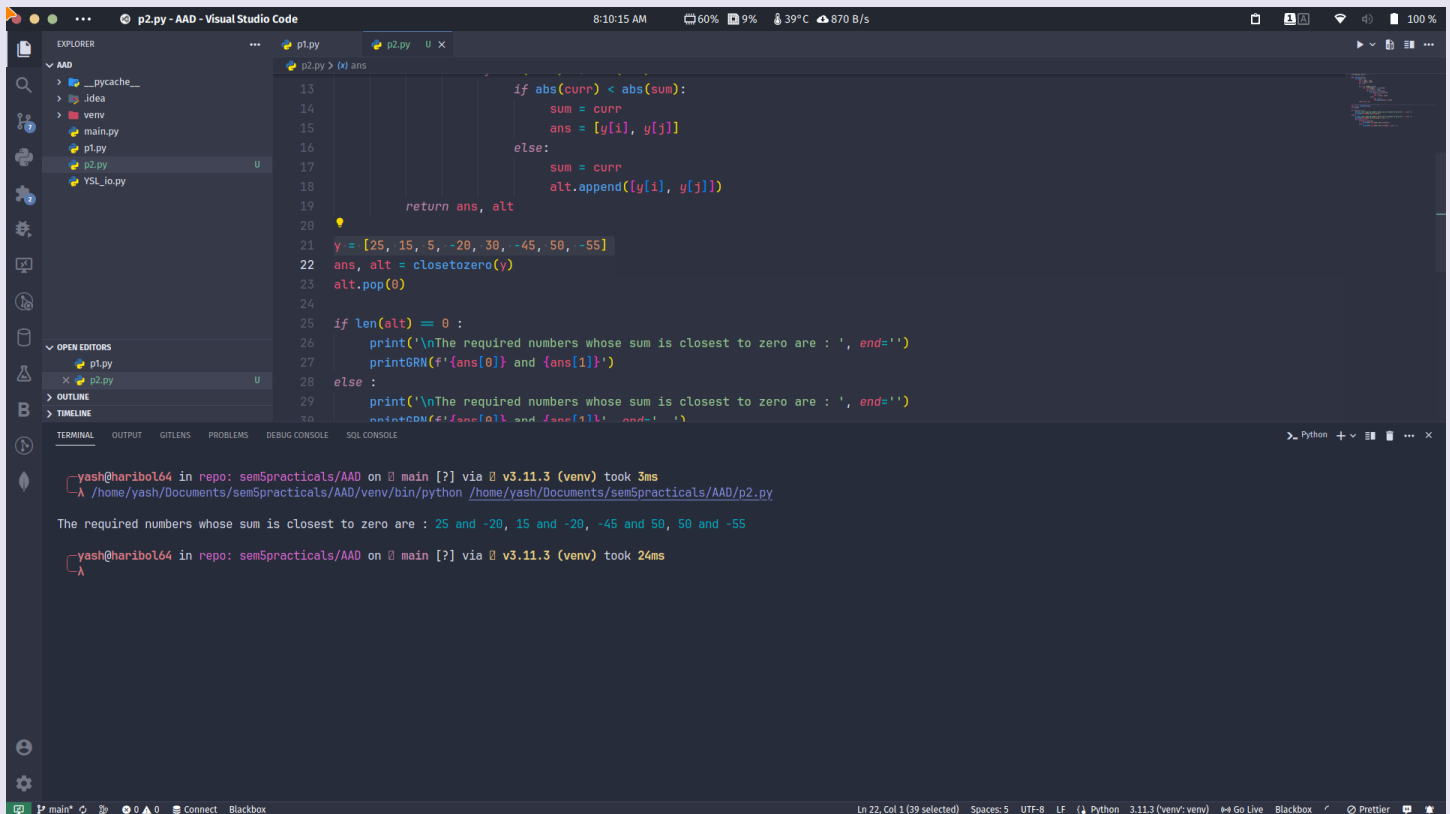
if elem == alt[-1]:

printGRN(f'{elem[0]} and {elem[1]}')

else :

printGRN(f'{elem[0]} and {elem[1]}', end=', ')
```

Output :



The screenshot shows a Visual Studio Code editor with a Python file named `p2.py` open. The code in the editor is as follows:

```
13     if abs(curr) < abs(sum):
14         sum = curr
15         ans = [y[i], y[j]]
16     else:
17         sum = curr
18         alt.append([y[i], y[j]])
19     return ans, alt
20
21 y = [25, 15, 5, -20, 30, -45, 50, -55]
22 ans, alt = closetozero(y)
23 alt.pop(0)
24
25 if len(alt) == 0 :
26     print('\nThe required numbers whose sum is closest to zero are : ', end='')
27     printGRN(f'{ans[0]} and {ans[1]}')
28 else :
29     print('\nThe required numbers whose sum is closest to zero are : ', end='')
30     printGRN(f'{ans[0]} and {ans[1]}', end=', ')
```

The terminal output shows the execution of the script:

```
yash@haribol64 in repo: semSpracticals/AAD on main [?] via v3.11.3 (venv) took 3ms
^C /home/yash/Documents/semSpracticals/AAD/venv/bin/python /home/yash/Documents/semSpracticals/AAD/p2.py

The required numbers whose sum is closest to zero are : 25 and -20, 15 and -20, -45 and 50, 50 and -55

yash@haribol64 in repo: semSpracticals/AAD on main [?] via v3.11.3 (venv) took 24ms
^C
```