

NAME:ELAMATHI P

PHN0:7639238996

## HASH AGILE CODING CHALLENGE

### PROBLEM STATEMENT:

HATFD1025

Find the Second Largest Element in an Array

Write a program to find the second-largest element in an array of integers without using any sorting algorithms or built-in array functions.

Instructions: Traverse the array manually to find both the largest and second-largest elements

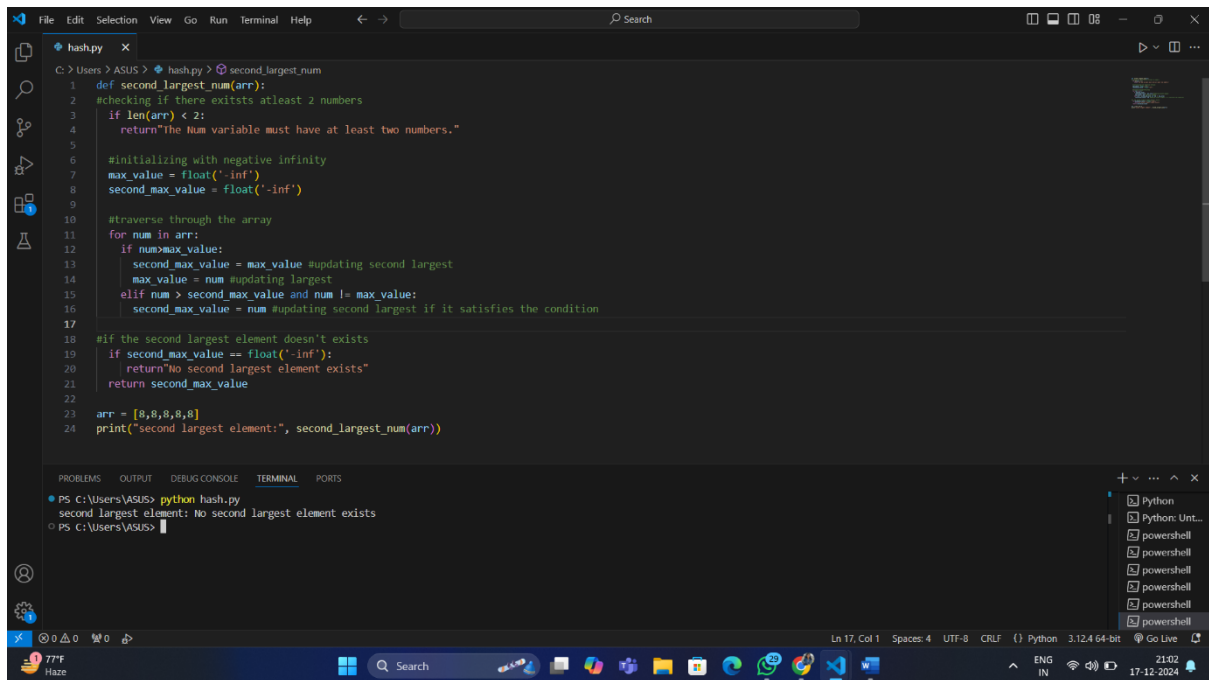
### PROGRAM CODE:

#### 1. SAMPLE INPUT

```
def second_largest_num(arr):
    #checking if there exists atleast 2 numbers
    if len(arr) < 2:
        return"The Num variable must have at least two numbers."
    #initializing with negative infinity
    max_value = float('-inf')
    second_max_value = float('-inf')
    #traverse through the array
    for num in arr:
        if num>max_value:
            second_max_value = max_value #updating second largest
            max_value = num #updating largest
        elif num > second_max_value and num != max_value:
            second_max_value = num #updating second largest if it satisfies the condition

    #if the second largest element doesn't exist
    if second_max_value == float('-inf'):
        return"No second largest element exists"
    return second_max_value

arr = [8,8,8,8,8]
print("second largest element:", second_largest_num(arr))
```



```
1 def second_largest_num(arr):
2     #checking if there exists atleast 2 numbers
3     if len(arr) < 2:
4         return "The Num variable must have at least two numbers."
5
6     #initializing with negative infinity
7     max_value = float('-inf')
8     second_max_value = float('-inf')
9
10    #traverse through the array
11    for num in arr:
12        if num>max_value:
13            second_max_value = max_value #updating second largest
14            max_value = num #updating largest
15        elif num > second_max_value and num != max_value:
16            second_max_value = num #updating second largest if it satisfies the condition
17
18    #if the second largest element doesn't exists
19    if second_max_value == float('-inf'):
20        return "No second largest element exists"
21    return second_max_value
22
23 arr = [8,8,8,8]
24 print("second largest element:", second_largest_num(arr))
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

PS C:\Users\ASUS> python hash.py  
second largest element: No second largest element exists  
PS C:\Users\ASUS>

## 2.SAMPLE OUTPUT

```
def second_largest_num(arr):
```

```
#checking if there exists atleast 2 numbers
```

```
if len(arr) < 2:
```

```
    return "The Num variable must have at least two numbers."
```

```
#initializing with negative infinity
```

```
max_value = float('-inf')
```

```
second_max_value = float('-inf')
```

```
#traverse through the array
```

```
for num in arr:
```

```
    if num>max_value:
```

```
        second_max_value = max_value #updating second largest
```

```
        max_value = num #updating largest
```

```
    elif num > second_max_value and num != max_value:
```

```
        second_max_value = num #updating second largest if it satisfies the condition
```

```
#if the second largest element doesn't exists
```

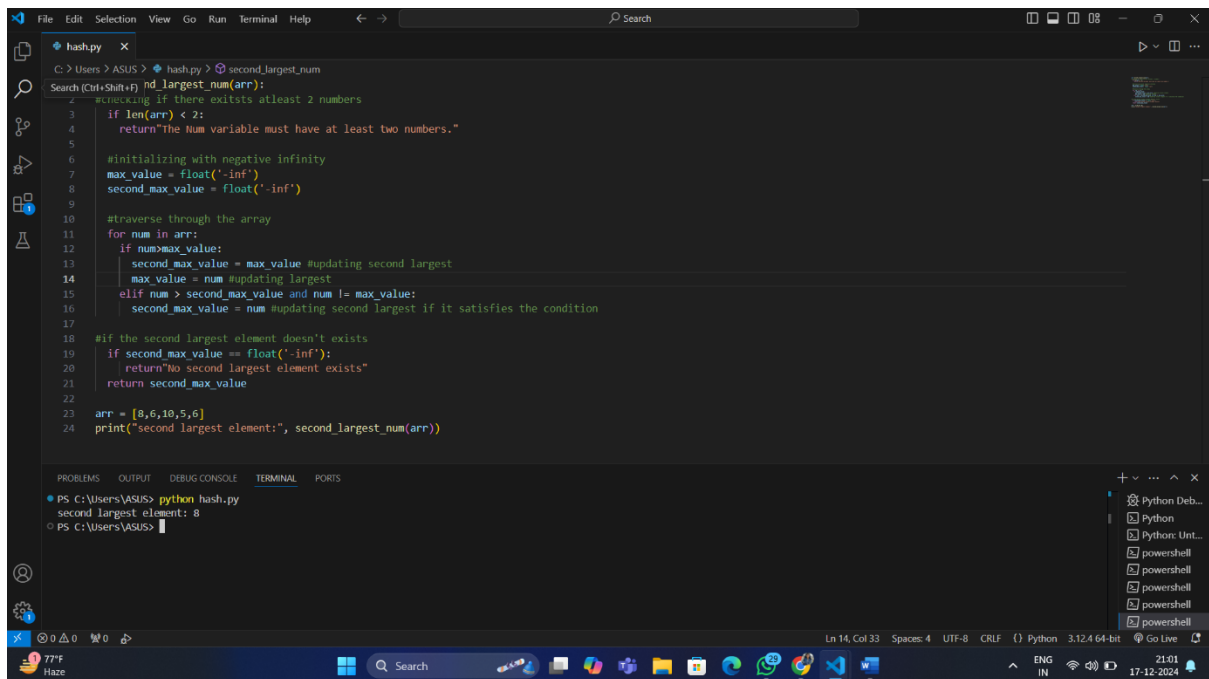
```
if second_max_value == float('-inf'):
```

```
    return "No second largest element exists"
```

```
return second_max_value
```

```
arr = [8,6,10,5,6]
```

```
print("second largest element:", second_largest_num(arr))
```



The screenshot shows a Visual Studio Code editor window with a Python file named `hash.py`. The code defines a function `second_largest_num` that finds the second largest number in an array. It includes comments for each step: checking for at least two numbers, initializing with negative infinity, traversing the array to find the maximum and second maximum, and handling the case where a second maximum does not exist. The array `[8, 6, 10, 5, 6]` is used as an example, and the output is printed. The terminal at the bottom shows the command `python hash.py` being executed, resulting in the output `second largest element: 8`.

```
File Edit Selection View Go Run Terminal Help
hash.py
C:\Users\ASUS> hash.py
2nd largest_num(arr):
#checking if there exists atleast 2 numbers
3 if len(arr) < 2:
4     return "The Num variable must have at least two numbers."
5
6 #initializing with negative infinity
7 max_value = float('-inf')
8 second_max_value = float('-inf')
9
10 #traverse through the array
11 for num in arr:
12     if num > max_value:
13         second_max_value = max_value #updating second largest
14         max_value = num #updating largest
15     elif num > second_max_value and num != max_value:
16         second_max_value = num #updating second largest if it satisfies the condition
17
18 #if the second largest element doesn't exists
19 if second_max_value == float('-inf'):
20     return "No second largest element exists"
21 return second_max_value
22
23 arr = [8,6,10,5,6]
24 print("second largest element:", second_largest_num(arr))

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
PS C:\Users\ASUS> python hash.py
second largest element: 8
PS C:\Users\ASUS>
```

### 3.SAMPLE OUTPUT

```
def second_largest_num(arr):
```

```
#checking if there exists atleast 2 numbers
```

```
if len(arr) < 2:
```

```
    return "The Num variable must have at least two numbers."
```

```
#initializing with negative infinity
```

```
max_value = float('-inf')
```

```
second_max_value = float('-inf')
```

#traverse through the array

for num in arr:

if num>max\_value:

second\_max\_value = max\_value #updating second largest

max\_value = num #updating largest

elif num > second\_max\_value and num != max\_value:

second\_max\_value = num #updating second largest if it satisfies the condition

#if the second largest element doesn't exists

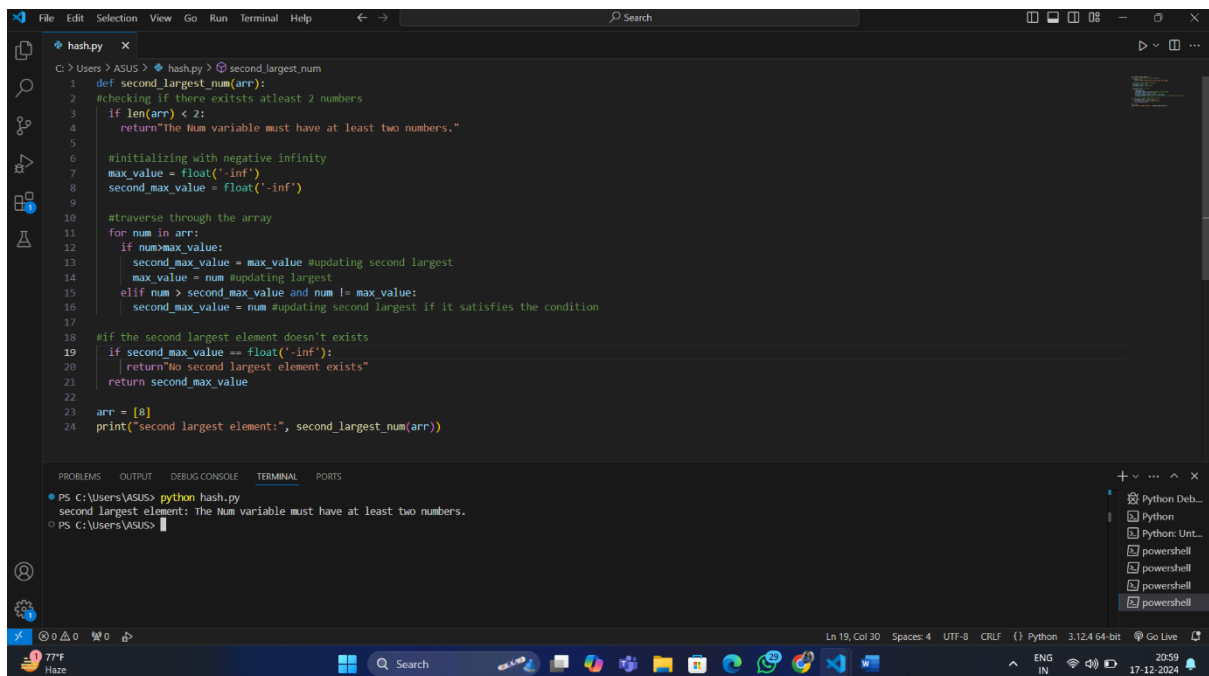
if second\_max\_value == float('-inf'):

return "No second largest element exists"

return second\_max\_value

arr = [8]

print("second largest element:", second\_largest\_num(arr))



The screenshot shows a Visual Studio Code editor window with a Python file named 'hash.py'. The code defines a function 'second\_largest\_num(arr)' that finds the second largest number in an array. It includes comments for each step: checking for at least two numbers, initializing with negative infinity, traversing the array to update max and second\_max values, and handling the case where no second largest element exists. The array 'arr' is set to [8], and the function is called. The terminal at the bottom shows the command 'python hash.py' being executed, resulting in the output: 'second largest element: The Num variable must have at least two numbers.'

```
hash.py
1 def second_largest_num(arr):
2     #checking if there exists atleast 2 numbers
3     if len(arr) < 2:
4         return "The Num variable must have at least two numbers."
5
6     #initializing with negative infinity
7     max_value = float('-inf')
8     second_max_value = float('-inf')
9
10    #traverse through the array
11    for num in arr:
12        if num>max_value:
13            second_max_value = max_value #updating second largest
14            max_value = num #updating largest
15        elif num > second_max_value and num != max_value:
16            second_max_value = num #updating second largest if it satisfies the condition
17
18    #if the second largest element doesn't exists
19    if second_max_value == float('-inf'):
20        return "No second largest element exists"
21    return second_max_value
22
23 arr = [8]
24 print("second largest element:", second_largest_num(arr))
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

PS C:\Users\ASUS> python hash.py  
second largest element: The Num variable must have at least two numbers.  
PS C:\Users\ASUS>

Ln 19, Col 30 Spaces: 4 UTF-8 CRLF Python 3.12.4 64-bit Go Live