

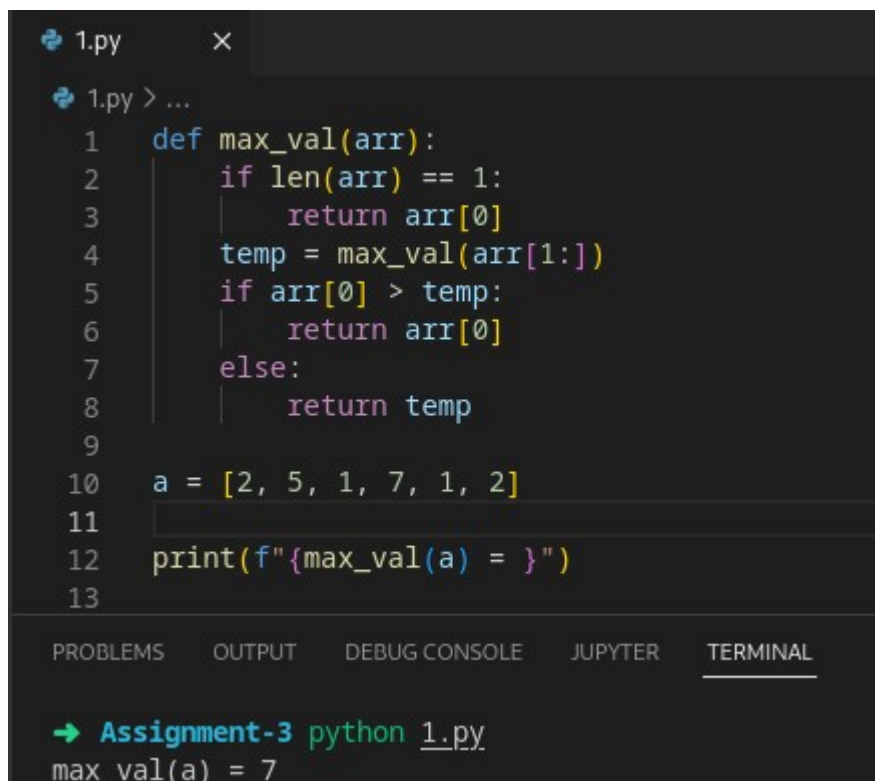
Assignment-3 (30 points)

Recursive Functions

Due date: 9/13/2022

Exercise-1: (7.5 points)

Write a recursive Python function that has a parameter representing a list of integers and returns the maximum stored in the list. Thinking recursively, the maximum is either the first value in the list or the maximum of the rest of the list, whichever is larger. If the list only has 1 integer, then its maximum is this single value, naturally. Copy and paste the screenshot of the result.



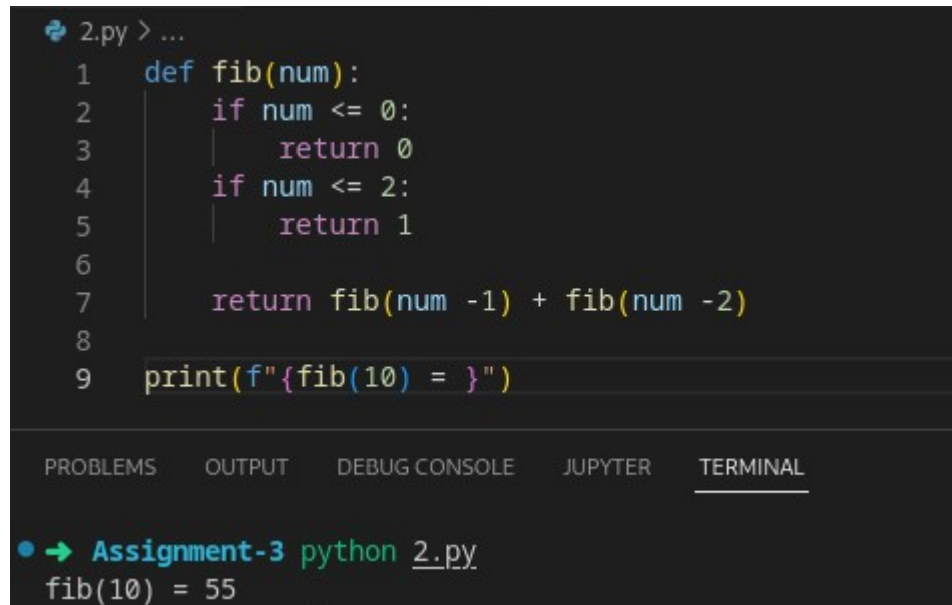
```
1.py x
1.py > ...
1  def max_val(arr):
2      if len(arr) == 1:
3          return arr[0]
4      temp = max_val(arr[1:])
5      if arr[0] > temp:
6          return arr[0]
7      else:
8          return temp
9
10 a = [2, 5, 1, 7, 1, 2]
11
12 print(f"{max_val(a)} = ")
13
```

PROBLEMS OUTPUT DEBUG CONSOLE JUPYTER TERMINAL

→ Assignment-3 python 1.py
max_val(a) = 7

Exercise-2: (7.5 points)

Run the Recursive Fibonacci function defined in the class notes with input a) 50 and b) 115. Copy and paste the screenshot of the result. Explain your understanding about this recursive function and the iterative function which you have written for your earlier assignment.



```
2.py > ...
1  def fib(num):
2      if num <= 0:
3          return 0
4      if num <= 2:
5          return 1
6
7      return fib(num -1) + fib(num -2)
8
9  print(f"{fib(10) = }")

PROBLEMS  OUTPUT  DEBUG CONSOLE  JUPYTER  TERMINAL

• → Assignment-3 python 2.py
fib(10) = 55
```

Note:

If the program doesn't work, test it with lesser values (let's say 10 or 15) and record your observations.

This recursive function, every time it is called it calls itself two additional times, and those sub calls do the same which make the function calls explode exponentially thus slowing the program down to an immense level

Exercise-3: (7.5 points)

Write a recursive function to calculate the sum of the positive integers of $n+(n-2)+(n-4)\dots$

Copy and paste the screenshot of the result.

```
3.py > ...
1  def sum(num):
2      if num % 2 != 0:
3          num -= 1
4      if num == 2:
5          return 2
6      if num <= 0:
7          return 0
8
9      return num + sum(num - 2)
10
11 print(f"sum(4) = ")
12
```

PROBLEMS OUTPUT DEBUG CONSOLE JUPYTER TERMINAL

• → Assignment-3 python 3.py
sum(4) = 6

Exercise-4: (7.5 points)

Write a recursive function to calculate the value of 'a' to the power 'b'? Copy and paste the screenshot of the result.

```
4.py > pow
1  def pow(a, b):
2      if b == 1:
3          return a
4      elif b == 0:
5          return 1
6
7      return a * pow(a, b-1)
8
9
10 print(f"{pow(2, 6)} = ")

PROBLEMS  OUTPUT  DEBUG CONSOLE  JUPYTER  TERMINAL

• → Assignment-3 python 4.py
pow(2, 6) = 64
```