

UPCISS

# O-Level (M3-R5)

Python Important  
Questions for O-Level

**Video Tutorial**  
**O-Level M3-R5 Full Video**  
Playlist Available on  
YouTube Channel **UPCISS**

**Free Online Computer Classes on**  
YouTube Channel **UPCISS**

**[www.youtube.com/upciss](http://www.youtube.com/upciss)**

**For free PDF Notes**

**Our Website: [www.upcissyoutube.com](http://www.upcissyoutube.com)**

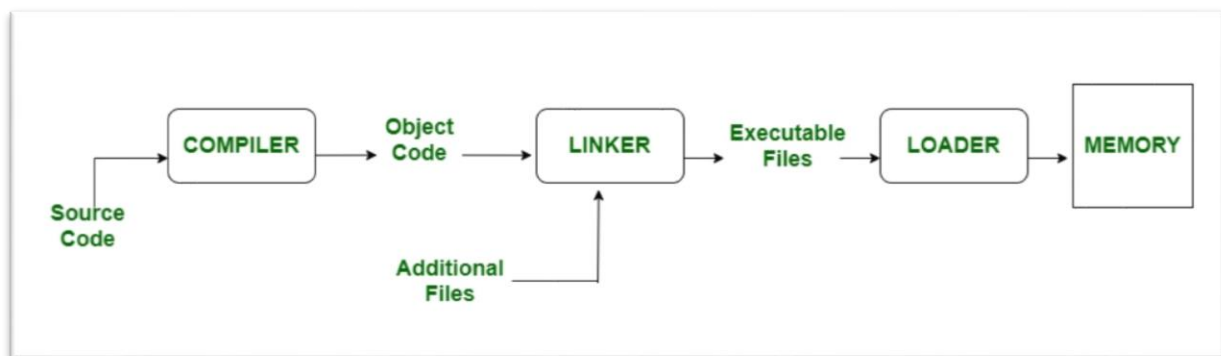


1. Explain the role of linker and loader in compilation.
2. What is a NumPy array? How they are different from lists?
3. Write a flowchart that finds the sum of series:
4. Write a recursive function to find the sum of digits of a number.
5. Write a program that takes a sentence as input from the user and returns the frequency of each letter .Use a variable of dictionary type to maintain the count.
6. Take an array of 2 rows and three columns, populate it and find the transpose.
7. Write a function that reads the contents of the file myfile.txt and counts the number of alphabets, lowercase letters, uppercase letters, digits and number of words.
8. Take two NumPy arrays having two dimensions. Concatenate the arrays on axis 1.
9. Write a Python program to get the smallest number from a list.
10. Write a NumPy program to convert a Python dictionary to a NumPy ndarray.
11. Write a program code to open a data file save element values 2,4,9,10,11 in this data file and print these data values by accessing the file.
12. Write the basic steps required by the interpreter to execute a python program.
13. How memory is managed in Python? Give the tools name that help to find bugs or perform static analysis?
14. Define mutable and immutable data type.
15. Explain the concept of Linear and Binary search with Python program.
16. Illustrate the LEGB rules and its significance with help of suitable diagram.

## 1. Explain the role of linker and loader in compilation.

**Linker:** यह एक Special program है जो Compiler या Assembler द्वारा Generate object files के टुकड़ों को जोड़कर के एक Executable file बनाता है (.exe file). Object files में linker file के Execution के लिए आवश्यक सभी Libraries को Search करके जोड़ने का काम करता है। यह दो या दो से अधिक object program को Merge करता है और उनके बीच एक link बनाता है।

**Loader:** यह एक Special program है जो linker से Executable file को input लेता है और उसे main memory में load करता है, और computer द्वारा execution के लिए इसे तैयार करता है।



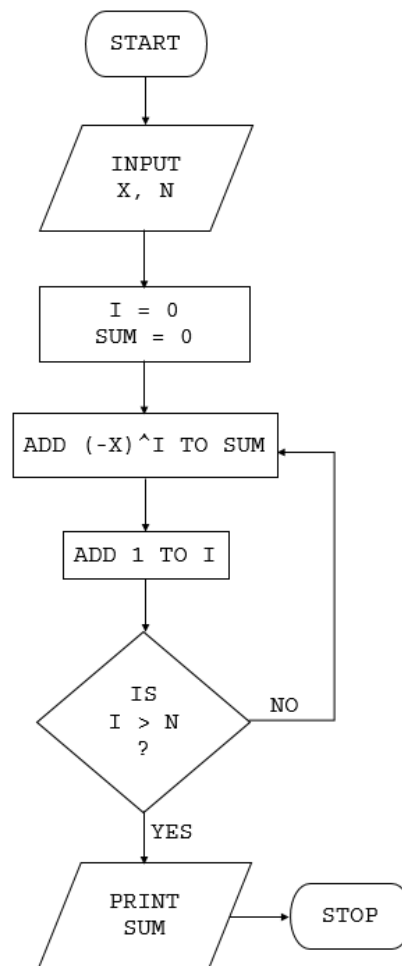
## 2. What is a NumPy array? How they are different from lists?

**NumPy:** NumPy python में scientific calculation के लिए एक बहुत ही अच्छा package है। NumPy array बड़ी संख्या में data पर advance mathematical और अन्य प्रकार के operations करने की सुविधा प्रदान करता है। NumPy कोई दूसरी programming language नहीं है, बल्कि यह एक python extension module है।

NumPy	List
Create NumPy array <code>numpy.array()</code>	Create List <code>L1 = [ ]</code>
by default Homogeneous	Homogeneous or heterogeneous.
Element wise operation is possible	Element wise operation is not possible
Low Memory consumption	High Memory consumption
Fast computing	Slow computing

3. Write a flowchart that finds the sum of series:  
 $s = 1 + x/1 + x^2/2 + x^3/3 + \dots$  upto n terms?

**Flowchart:**



4. Write a recursive function to find the sum of digits of a number.

**Program:**

```
def sum_of_digit( n ):
    if n == 0:
        return 0
    return (n % 10 + sum_of_digit(int(n / 10)))

num = 12345
result = sum_of_digit(num)
print("Sum of digits in",num,"is", result)
```

**Output:**

Sum of digits in 12345 is 15

5. Write a program that takes a sentence as input from the user and returns the frequency of each letter .Use a variable of dictionary type to maintain the count.

### Program:

```
test_str = "UPCISS"  
all_freq = {}  
  
for i in test_str:  
    if i in all_freq:  
        all_freq[i] += 1  
    else:  
        all_freq[i] = 1  
  
print ("Count of all characters ",all_freq)
```

### Output:

Count of all characters {'U': 1, 'P': 1, 'C': 1, 'I': 1, 'S': 2}

6. Take an array of 2 rows and three columns, populate it and find the transpose.

### Program:

```
def transpose(A, B):  
    for i in range(3):  
        for j in range(2):  
            B[i][j] = A[j][i]  
  
A = [ [1, 1, 1],  
      [2, 2, 2]]  
  
B = [[0,0],[0,0],[0,0]]  
transpose(A, B)  
  
print("Result matrix is")  
for i in range(3):  
    for j in range(2):  
        print(B[i][j], " ", end='')  
    print()
```

### Output:

```
Result matrix is  
1 2  
1 2  
1 2
```

7. Write a function that reads the contents of the file myfile.txt and counts the number of alphabets, lowercase letters, uppercase letters, digits and number of words.

### Program:

```
def count_str(st):
    alpha=low=upp=digit=0
    word=1
    for i in st:
        if i.isalpha():
            alpha+=1
            if i.islower():
                low+=1
            else:
                upp+=1
        elif i.isdigit():
            digit+=1
        elif i.isspace():
            word+=1
    print('Alphabets:',alpha)
    print('Lowercase:',low)
    print('Uppercase:',upp)
    print('Digits:',digit)
    print('Words:',word)

f = open('myfile.txt','r')
st= f.read()
count_str(st)
f.close()
```

### Output:

```
Alphabets: 28
Lowercase: 19
Uppercase: 9
Digits: 7
Words: 9
```

```
myfile.txt content
1 Hello Students
2 Welcome to UPCISS
3 4502
```

8. Take two NumPy arrays having two dimensions. Concatenate the arrays on axis 1.

### Program:

```
import numpy as np
arr1 = np.arange(1,10).reshape(3,3)
arr2 = np.arange(10,19).reshape(3,3)
print(arr1,'\n')
print(arr2,'\n')
arrcon = np.concatenate((arr1,arr2),axis=1)
print(arrcon)
```

## Output:

```
[[1 2 3]
 [4 5 6]
 [7 8 9]]
```

```
[[10 11 12]
 [13 14 15]
 [16 17 18]]
```

```
[[ 1  2  3 10 11 12]
 [ 4  5  6 13 14 15]
 [ 7  8  9 16 17 18]]
```

9. Write a Python program to get the smallest number from a list.

## Program:

```
li=[2,6,5,1,4]
small=li[0]
for i in li:
    if small > i:
        small=i
print(small)
```

**Output:** 1

10. Write a NumPy program to convert a Python dictionary to a NumPy ndarray.

## Program:

```
import numpy as np
dict = {0: 3, 1: 1, 2: 8, 3: 5}
nd_array = np.array(list(dict.items()))
print(nd_array)
```

## Output:

```
[[0 3]
 [1 1]
 [2 8]
 [3 5]]
```

11. Write a program code to open a data file save element values 2,4,9,10,11 in this data file and print these data values by accessing the file.

**Program:**

```
f = open('data.txt', 'r+')
f.write('2491011')
f.seek(0)
print(f.read())
f.close()
```

**Output:**

2491011

12. Write the basic steps required by the interpreter to execute a python program.

**Follow the steps written below.**

- Make a text file and save it with the name of your choice with an extension . py.
- Write the code to print hello world in the .py file and save your file.
- Open command prompt.
- Run the command – python filename.py.

13. How memory is managed in Python? Give the tools name that help to find bugs or perform static analysis?

जैसा कि हम जानते हैं, Python dynamic memory allocation को उपयोग करता है जिसे Heap data structure manage करता है।

Memory Heap उन object और अन्य data structure को रखता है जिनका उपयोग प्रोग्राम में किया जाएगा।

PyChecker और Pylint static analysis tool हैं जो Python में bugs खोजने में help करते हैं।

14. Define mutable and immutable data type.

Mutable: mutable data type में हम object की value को change कर सकते हैं। List, Sets, Dictionaries ये सब Mutable data type हैं।

Immutable: immutable data type में हम object की value को change नहीं कर सकते हैं। int, float, bool, str, tuple ये सब immutable data type हैं।



## 15. Explain the concept of Linear and Binary search with Python program.

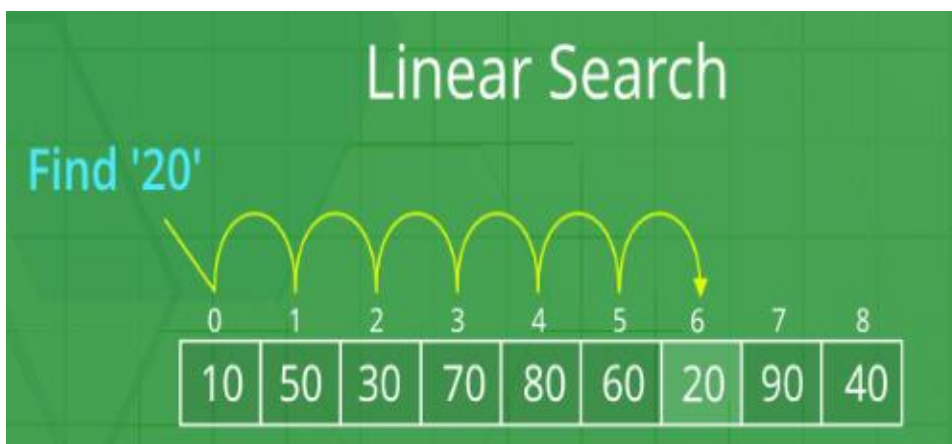
Linear Search simple approach को follow करता है, जैसे अगर हमें किसी array से किसी element को search करना है, तो हम उस element को array के सभी element के साथ एक-एक करके check करेंगे और अगर वो element हमें मिलता है तो हम उसे return कर देंगे। linear search में Binary Search की अपेक्षा समय ज्यादा लगता है। जैसे:-

```
# Program for Linear Search (Sequential Search)
def linear_search(arr, n, x):
    for i in range(0, n):
        if (arr[i] == x):
            return i
    return -1

arr = [10, 50, 30, 70, 80, 60, 20, 90, 40]
x = 20
n = len(arr)
result = linear_search(arr, n, x)
if(result == -1):
    print("Element is not present in array")
else:
    print("Element is present at index", result)
```

### Output:

Element is present at index 6



Binary Search एक searching algorithm है जो array को बार बार आधे में विभाजित करके element को search करता है। binary search हमेशा एक sorted array के साथ किया जाता है। यह Linear search की अपेक्षा कम समय लगाता है।

```
# Program for Binary Search

def binary_search(arr, x):
    low, high = 0, len(arr)-1
    while low <= high:
        mid = (low + high) // 2
        if x > arr[mid]:
            low = mid + 1
        elif x < arr[mid]:
            high = mid - 1
        else:
            return mid
    return -1

arr = [2,5,8,12,16,23,38,56,72,91]
x = 23
result = binary_search(arr, x)
if(result == -1):
    print("Element is not present in array")
else:
    print("Element is present at index", result)
```

### Output:

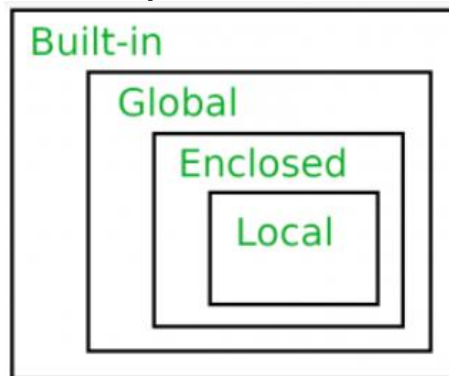
Element is present at index 5

		0	1	2	3	4	5	6	7	8	9
Search 23		2	5	8	12	16	23	38	56	72	91
		L=0	1	2	3	M=4	5	6	7	8	H=9
23 > 16 take 2 <sup>nd</sup> half		2	5	8	12	16	23	38	56	72	91
		0	1	2	3	4	L=5	6	M=7	8	H=9
23 < 56 take 1 <sup>st</sup> half		2	5	8	12	16	23	38	56	72	91
		0	1	2	3	4	L=5, M=5	H=6	7	8	9
Found 23, Return 5		2	5	8	12	16	23	38	56	72	91

## 16. Illustrate the LEGB rules and its significance with help of suitable diagram.

Program के अंतर्गत हर name reference के लिए अर्थात जो आप किसी function या program से variable को access करते हैं तो python हमेशा name resolution rule को अपनाता है जिसे LEGB rule कहते हैं।

- Local(L): Defined inside function/class
- Enclosed(E): Defined inside enclosing functions(Nested function concept)
- Global(G): Defined at the uppermost level
- Built-in(B): Reserved names in Python built-in modules



It takes a lot of hard work to make notes, so if you can pay some fee 50, 100, 200 rupees which you think is reasonable, if you are able to Thank you...

नोट्स बनाने में बहुत मेहनत लगी है , इसलिए यदि आप कुछ शुल्क 50,100, 200 रूपए जो आपको उचित लगता है pay कर सकते है, अगर आप सक्षम है तो, धन्यवाद ।



**G Pay**

**Jitendra Kumar**

Account Number

**17000100008177**

IFSC Code

**BARB0KAFARA**

UPI ID

**Jitendraupciss@okicici**

**Scan QR**

