

**Assignment -1**  
Python Programming

Assignment Date	29 September 2022
Student Name	A.Nabesha Fathima
Student Roll Number	724019104016
Maximum Marks	2 Marks

**Question-1:**

Write a Python program to remove all instances of a given value from a given array of integers and find the length of the new array.

Solution:

```
def remove_element(array_nums, val):
    i = 0
    while i < len(array_nums):
        if array_nums[i] == val:
            array_nums.remove(array_nums[i])

        else:
            i += 1

    return len(array_nums)

print(remove_element([1, 2, 3, 4, 5, 6, 7, 5], 5))
print(remove_element([10,10,10,10,10], 10))
print(remove_element([10,10,10,10,10], 20))
print(remove_element([], 1))
```

## Output:

```
main.py  Run Shell Clear
1 def remove_element(array_nums, val):
2     i = 0
3     while i < len(array_nums):
4         if array_nums[i] == val:
5             array_nums.remove(array_nums[i])
6
7         else:
8             i += 1
9
10    return len(array_nums)
11    print(remove_element([1, 2, 3, 4, 5, 6, 7, 5], 5))
12    print(remove_element([10,10,10,10,10], 10))
13    print(remove_element([10,10,10,10,10], 20))
14    print(remove_element([], 1))
```

## Question-2:

Write a Python program to print a given N by M matrix of numbers line by line in forward > backwards > forward >... order.

Input matrix:

```
[[1, 2, 3, 4],
[5, 6, 7, 8],
[0, 6, 2, 8],
[2, 3, 0, 2]]
```

Solution:

```
def print_matrix(nums):
```

```
    flag = True
```

```
    for line in nums:
```

```
        if flag == True:
```

```
            i = 0
```

```
            while i < len(line):
```

```
                print(line[i])
```

```
i += 1
```

```
flag = False
```

```
else:
```

```
i = -1
```

```
while i > -1 * len(line) - 1:
```

```
    print(line[i])
```

```
    i = i - 1
```

```
flag = True
```

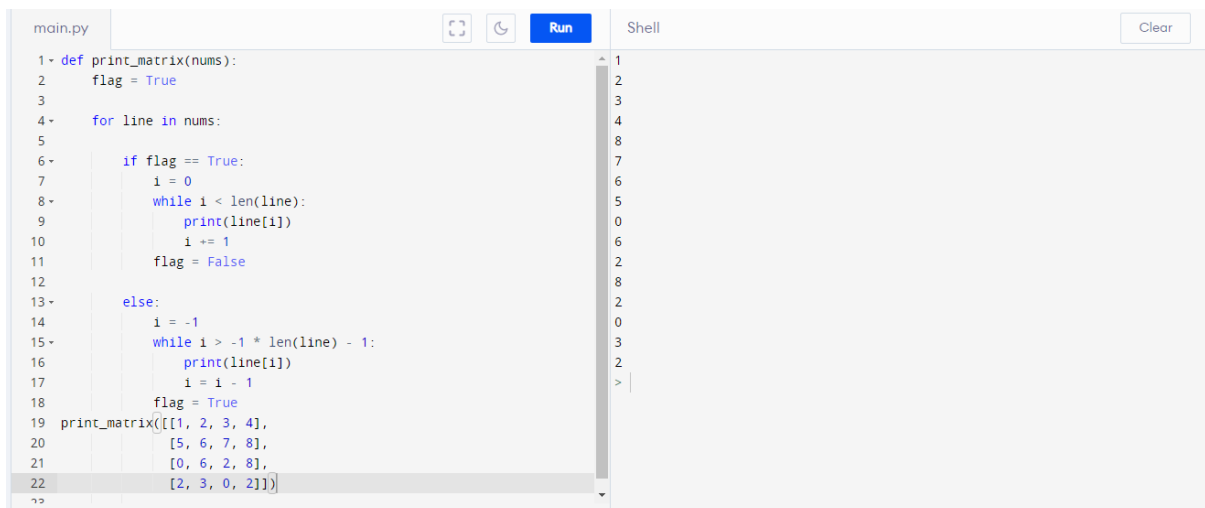
```
print_matrix([[1, 2, 3, 4],
```

```
              [5, 6, 7, 8],
```

```
              [0, 6, 2, 8],
```

```
              [2, 3, 0, 2]])
```

Output;



The screenshot shows a code editor with a file named 'main.py'. The code defines a function 'print\_matrix(nums)' that iterates through a list of lists. It uses a 'flag' to alternate between printing the first element of each row (when 'flag' is True) and the last element (when 'flag' is False). The input list is [[1, 2, 3, 4], [5, 6, 7, 8], [0, 6, 2, 8], [2, 3, 0, 2]]. The output in the 'Shell' pane shows the elements printed in order: 1, 2, 3, 4, 8, 7, 6, 0, 2, 3, 2, 0.

```
main.py  Run  Shell  Clear

1 def print_matrix(nums):
2     flag = True
3
4     for line in nums:
5
6         if flag == True:
7             i = 0
8             while i < len(line):
9                 print(line[i])
10                i += 1
11            flag = False
12
13        else:
14            i = -1
15            while i > -1 * len(line) - 1:
16                print(line[i])
17                i = i - 1
18            flag = True
19    print_matrix([[1, 2, 3, 4],
20                  [5, 6, 7, 8],
21                  [0, 6, 2, 8],
22                  [2, 3, 0, 2]])
23
```

```
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
>
```

Question-3:

Write a Python program to compute the largest product of three integers from a given list of integers.

Sample Input:

[-10, -20, 20, 1]

[-1, -1, 4, 2, 1]

[1, 2, 3, 4, 5, 6]

Solution:

```
def largest_product_of_three(nums):  
    max_val = nums[1]  
  
    for i in range(len(nums)):  
        for j in range(i+1, len(nums)):  
            for k in range(j+1, len(nums)):  
                max_val = max(nums[i] * nums[j] * nums[k], max_val)  
  
    return max_val  
  
print(largest_product_of_three([-10, -20, 20, 1]))  
print(largest_product_of_three([-1, -1, 4, 2, 1]))  
print(largest_product_of_three([1, 2, 3, 4, 5, 6]))
```

**Output:**

<pre>1 def largest_product_of_three(nums): 2     max_val = nums[1] 3 4     for i in range(len(nums)): 5         for j in range(i+1, len(nums)): 6             for k in range(j+1, len(nums)): 7                 max_val = max(nums[i] * nums[j] * nums[k], max_val) 8 9     return max_val 10 11 print(largest_product_of_three([-10, -20, 20, 1])) 12 print(largest_product_of_three([-1, -1, 4, 2, 1])) 13 print(largest_product_of_three([1, 2, 3, 4, 5, 6]))</pre>	<pre>4000 8 120 &gt;  </pre>
--	------------------------------

#### Question-4:

Write a Python program to find the first missing positive integer that does not exist in a given list.

Sample Input:

[2, 3, 7, 6, 8, -1, -10, 15, 16]

[1, 2, 4, -7, 6, 8, 1, -10, 15]

[1, 2, 3, 4, 5, 6, 7]

[-2, -3, -1, 1, 2, 3]

Solution:

```
def first_missing_number(nums):
    if len(nums) == 0:
        return 1

    nums.sort()
    smallest_int_num = 0

    for i in range(len(nums) - 1):

        if nums[i] <= 0 or nums[i] == nums[i + 1]:
            continue
        else:
            if nums[i + 1] - nums[i] != 1:
                smallest_int_num = nums[i] + 1
                return smallest_int_num
    if smallest_int_num == 0:
        smallest_int_num = nums[-1] + 1
    return smallest_int_num

print(first_missing_number([2, 3, 7, 6, 8, -1, -10, 15, 16]))
```

```
print(first_missing_number([1, 2, 4, -7, 6, 8, 1, -10, 15]))
print(first_missing_number([1, 2, 3, 4, 5, 6, 7]))
print(first_missing_number([-2, -3, -1, 1, 2, 3]))
```

### Output:

```
1 def first_missing_number(nums):
2     if len(nums) == 0:
3         return 1
4
5     nums.sort()
6     smallest_int_num = 0
7
8     for i in range(len(nums) - 1):
9
10        if nums[i] <= 0 or nums[i] == nums[i + 1]:
11            continue
12        else:
13            if nums[i + 1] - nums[i] != 1:
14                smallest_int_num = nums[i] + 1
15                return smallest_int_num
16    if smallest_int_num == 0:
17        smallest_int_num = nums[-1] + 1
18    return smallest_int_num
19
20 print(first_missing_number([2, 3, 7, 6, 8, -1, -10, 15, 16]))
21 print(first_missing_number([1, 2, 4, -7, 6, 8, 1, -10, 15]))
22 print(first_missing_number([1, 2, 3, 4, 5, 6, 7]))
23 print(first_missing_number([-2, -3, -1, 1, 2, 3]))
```

### Question-5:

Write a Python program to randomly generate a list with 10 even numbers between 1 and 100 inclusive.

Note: Use random.sample() to generate a list of random values.

Solution:

```
import random
print(random.sample([i for i in range(1,100) if i%2==0], 10))
```

### Output:

main.py

Run

```
1 import random
2 print(random.sample([i for i in range(1,100) if i%2==0], 10))
```

Shell

Clear

```
[66, 74, 76, 38, 68, 52, 30, 64, 90, 58]
> |
```

main.py

Run

```
1 import random
2 print(random.sample([i for i in range(1,100) if i%2==0], 10))
```

Shell

Clear

```
[66, 74, 76, 38, 68, 52, 30, 64, 90, 58]
> |
```