Q Ans:-

PGM:-

```
def Sub_list(l):
    list = [[]]
for i in range(len(l)+1):
for j in range(i):
list.append(l[j:i])
return lists
l1 = [1, 2, 3]
print(Sub_lists(l1))
```

Output:-

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

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

(2) PGM:

```
def Count_range_in_list(li, min, max):
    ctr = 0
    for x in li:
        if min <= x <= max:
            ctr += 1
    return ctr
list1 = [10, 20, 30, 40, 40, 70, 80, 99]
print (Count_range_in_list(list1, 40, 100))
list2 = ['a', 'b', 'c', 'd', 'e', 'f']
print (Count_range_in_list(list2, 'a', 'e'))
```

O/P:-

Count_range_in_list(li, min, max)

list :-

0	1	2	3	4	5	6	7	8
10	20	30	40	40	40	70	80	99

list:-

0	1	2	3	4	5
"a"	"b"	"c"	"d"	"e"	"f"

(3) Ans:

```
def Order(lst):
    assending = descending = True
    for i in range(len(lst)-1):
        if lst[i] > lst[i+1]:
            assending = False
        elif lst[i] < lst[i+1]:
            descending = False
    return (assending or descending)
```

O/P:-

Order([1, 2, 3, 4])

True.

Order([1, 2, 3, 2])

False.

Order([3, 2, 1, 0])

True.

Order([3, 2, 1, 4]) -> False.

④ PGM:-

```
n = int(input("Enter an integer:"))
print("the division of the
        number are:")
for i in range(1, n+1):
    if(n%i==0):
        print(i)
```

O/P:

```
Enter an integer: 25
The division of the
number are :-
1
5
25
```

⑤ Sol:-

input a positive number: 1000

6 is a perfect number.

$6 = 3+2+1$

28 is a perfect number.

$28 = 14+7+4+2+1$

496 is a perfect number:

$496 = 248+124+62+31+16+8+4+2+1$

8128 is perfect number:-

$8128 = 4064+2032+1016+508+254+127+64+32+16+8+4+2+1$

Input a positive number: 6

$6 = 3+3+2+1$

Input a positive no:-10

Error!

Number must be positive.

(6) List:-

```
def multiplylist (mylist):
    result = 1
    For x in mylist:
    result = result * x
    return result

# Driver Code:
    list 1 = [1,2,3]
    list 2 = [3,2,4]
Print (multiplylist (list1))
Print (multiplylist (list2))
```

o/p:-

6

24.

(7)

Sol:-

Main list:-

a=[10, 20, 30, 20, 10, 50, 60, 40,
80, 50, 40].

Ans:-

empty temporary list

temp[ ]

removing duplicates list.

for elements in a:

If (element not in temp):

temp.append (element)

Assiging a temporary list.-

a= temp

print ("unique list:", a)

Unique list:-

[10, 20, 30, 40, 50, 60, 80]

(8) Two list:-

Sol:-

```
def Common_member(a,b):
    a_set = set(a)
    b_set = set(b)
if (a_set & b_set):
print (a_set & b_set);

else:
    print ("No Common Elements")
```

a = [5, 15, 38, 8, 98]

b = [10, 5, 15, 38, 8, 98]

Common_member(a,b)

a = [5, 15, 38, 8, 98]

b = [5, 15, 38, 8, 98]

output:-

{10}

No Common Elements.