CT-01 (A)

1. sort the list asending order

→ numbers = [3, 8, 1, 5, 12, 7, 9, 3, 8, 2]

```
numbers.sort()
print(numbers)
```

2. Make the list unique and find the second largest number

└ কারণের জন্য set - এ নিতে হবে।

→

```
numbers = list(set(numbers))
print("second largest number", numbers[len(numbers)-2]
```

3. Create a new list where each number is multiplied by its index position in the unique list.

→

```
new list = [ ]
for i in range(len(numbers)):
    new list.append(i * numbers[i])
print(new_list)
```

CamScanner

Q1:- (B)

Q1. Output: 20

x → Integer

int(y) → Streing

✓ Print (x + int(y))

Q2. for i in range (5)
Print (i, end = " ")

→ for loop iterates from i = 0 to 4. (exclusive 5)

Q3. 5 times. (0 1 2 3 4)

Q4. Nested Loop.

i from 0 to 2.
j (from 0) to 1.

0 0
0 1
1 0
1 1
2 0
2 1

Q5. NOT Equal.

CamScanner

06. True

[In python any non-zero number is considered as "Truth" in boolean context.

since, num = -5 [which is non-zero value]

07. True

[In python 0 and empty sequence consider as "false"]

List slicing

08. copy

1st = [1, 2, 3, 4]

Print (1st [1:3])

Use list [start: end] to get a part of the list.

End is exclusive.

index 1 (inclusive) to index 3 (exclusive).

Output: 2, 3

(09) If an element exists in a list using (in) Keyword.

The in operator returns True if the specified element is found in the list and False otherwise.

```
my_list = [1, 2, 3, 4]
if 3 in my_list
Print("Element exists")
```

(10) = (Assignment operator) (assign a value to a variable)

== (Equality operator) (compare two values to check if they are equal)

CT-01(c)

Remove duplicates

```
numbers = [10, 20, 30, 10, 40, 20, 50, 30]
n_numbers = list(dict.fromkeys(numbers))
Print (n_numbers)
```

Find Average of all numbers

```
average = sum(numbers) / len(numbers)
Print(average)
```

Convert Unique-number

```
unique_numbers = list(dict.fromkeys(numbers))
result_string = " ".join(map(str, unique_numbers))
Print (result_string)
```

CamScanner

CT-1 (D)

Extended packing

① a, *b, c = (1, 2, 3, 4, 5)

→ The asterisk (*) operator tells python to collect all the leftover items from the tuple into a new list.

String Reversal with slicing

② s = "python"
Print(s[ : : -1])

start : stop : step

→ nohtyp

→ start from the end of the string and go to the beginning with a step of -1. This effectively reverse the string.

③ [1, 2, 3, 4, [5, 6]]

④ list.remove(value)

→ method searches the list for the first occurence of the specified value & removes it.

list.pop(index)

→ removes an item by its index, not it's value.

del list[index]

→ also removes an item by its index.

⑤ True.

06. Indexing:

```
my_list = ['a', 'b', 'c', 'd']
Print (my_list[1])  #output: b
```

Slicing:

```
my_list = ['a', 'b', 'c', 'd', 'e']
print (my_list [1:4]        #output: b, c, d

new_list = my_list [:]
Print (new_list)            #output: ['a', 'b', 'c', 'd', 'e']
```

Spring '23

1.(b) my_list = [1,2,3,4,5]

Print (list[-1])

↳ negative indices elements from end of the list.

list.append(6) [1,2,3,4,5,6]

Print (list) → append() ; a single element adds end of the list.

list.insert(2,7) [1,2,7,3,4,5,6]

↳ insert() adds a specified index.

The first argument is the index where the element will be inserted.

The second argument is the element itself.

print (list)

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

my_list.pop(2) (popped element: 7)

↳ remove and returns at the specified index.

If no index is provided; it removes and returns the last element

Print(List)

[2,3,4,5,6]

my_list.remove(1)

↳ remove the first occurrence of the specified value from the list.

Print (list)

list = list + [8,9,10] [2,3,4,5,6,8,9,10]

Print (list)

Print (len(list)) 8

Print (list [1:4] 3,4,5

not including index 3

Print (list [ : 3]) 2,3,4

Print (list * 2) [2,3,4,5,6,8,9,10,2,3,4,5,6,8,9,10]

↳ multiplying a list by an integer (n)

Create a new list that contains the original list's elements repeated n times.

1, 2, 3, 4, 5, 6, 7, 8, 9, 10.

Fall'23

Output = my_list [-2 : -9 : -2] → 9, 7, 5, 3

-2 (starting index): refers to second to last element

-9 the stopping index (exclusive)

ninth element from the end of the list

-2 (the step)

Slice will move backward through the list, taking every 2nd element

Result: 0

Practice problem:

1. Sum of prime numbers in a tuple:

```
def is_prime(n):
    if n<2:
        return False
    for i in range (2,n):
        if n % i == 0 :
            return False
    return True

tup=(2, 4, 5, 6, 7, 9)
sum_of_primes= 0

for num in tup:
    if is_prime(num):
        sum_of_primes += num
print (sum_of_primes)
```

2. Even elements present in set1 but not in set2.

set1 = {1, 2, 3, 4, 5}

set2 = {2, 5, 6}

```
result = set()
for num in set1:
    if num not in set2 and num % 2 == 0:
        result.add(num)
print(result)
```

3. Students who scored above average

→ scores = {'Alice': 70, 'Bob': 85, 'Carol': 90, 'Dave' = 65}

```
total = 0
Count = 0
for score in scores.values():
    total += score
    Count += 1
average = total / Count
for name, score in Scores.item():
    if score > average
        Print(name)
```

4. Count elements divisible by 3 and 5.

```
numbers = {15, 30, 22, 45, 10}
count = 0.
for num in numbers:
    if num % 3 == 0 and num % 5 == 0:
        count += 1
print (count)
```

5. Count vowels and consonants in a string.

```
text = "Programming"
vowels = "aeiouAEIOU"
vowel_count = 0
consonant_count = 0
for char in text:
    if char.isalpha():  # ignore spaces or symbols.
        if char in vowels:
            vowel_count += 1
    else
        consonant_count += 1
print ("vowels:", vowel_count)
print ("consonants:", consonant_count)
```

CamScanner

6. print strings start and end with same letter.

```
tup = {"level", "madam", "world", "python"}
for char in tup:
    if char[0] == char[-1]:
        Print(char)
```

7. print unique odd numbers in sorted order

```
list = [3,4,5,3,7,8,7]
odd_numbers = set()
for num in list:
    if num% 2 != 0:
        odd_numbers.add(num)
sorted_odds = sorted(odd_numbers)
for num in sorted_odds:
    Print(num, end = " ")
```

8. print words with frequency > 2 from dictionary

```
freq = {"apple": 2, "banana": 4, "cherry": 3,
        "date": 1}
result = {}
    for char, count in freq.items():
        if count > 2:
            result [char] = count
Print (result)
```

9. print the second largest number

```
numbers = [10, 20, 20, 30, 40, 40]
unique_numbers = list(set(numbers))
unique_numbers.sort(reverse = true)
print (unique_numbers[1])
```

10. Check if first 3 and last characters of a string are the same.

```
s = "abcXYZabc"
if s[:3] == s[-3:]
    Print ("Yes")
else:
    Print ("NO")
```

11. Count employees in each department.

```
employees = {'Alice':'HR', 'Bob':'IT', 'Carol':'HR',
             'Dave':'IT'}
dept_count = { }
for name, department in employees.items():
    if department in dept_count:
        dept_count [department] += 1
    else:
        dept_count [department] = 1

for department, count in dept_count.items():
    Print (department + ":", count)
```

Symmetric difference

12. ~~result~~ set1 = {1, 2, 3}

set 2 = {3, 4, 5}

```
result = set1.symmetric_difference(set2)
print (result)
```

Tuple, Nested Tuple, Indexing: print all integers recursively

13.
```
def print_integers (tup):
    for element in tup:
        if type(element) == int:
            print (element)
        elif type (element) == tuple:
            Print_integers(element)

nested_tuple = (1, (2, 3), (4, (5, 6)))
Print_integers (nested_tuple)
```

List, For loop, If-Else:

14. list = [1, 2, 4, 7, 9, 16, 20]

```
for num in list
    if int(num ** 0.5) ** 2 == num:
        print(num, end = " ")
```

15. String, for loop: reverse a string without slicing or reversed()

```
text = "python"
reversed_str = " "
for char in text:
    reversed_str = char + reversed_str
print(reversed_str)
```

✓ Dictionary, If-Else:

```
d = {'a': 1, 'b': 2}
Key_to_check = 'c'
if Key_to_check in d:
    print(d[Key_to_check])
else:
    print("Not Found")
```

CamScanner

Set, For loop, Membership

17.
```
list = {1,2,3,4,5}
s = {2,4}
for num in list:
    if num not in s:
        print(num, end="")
```

Tuple, For loop()

18.
```
tup = (1,2,2,3,2,4,4)
max_count = 0
for element in set(tup):
    count = tup.count(element)
    if count > max_count
Print (max_count)
```

List, If-else

19.
```
list = [1,2,3,5,4]
if list == sorted(list)
    print ("Sorted")
else:
    Print ("NOt sorted")
```

String, For loop, If-Else

20. sentence = "An apple is on the table"

vowels = {'a', 'e', 'i', 'o', 'u'}

Count = 0

words = sentence.split()

for word in words:

    if word[0].lower() in vowels:

        Count += 1

Print (count)

Dictionary, For-loop, If Else

21. products = {'Pen': 20, 'notebook': 70, 'backpack': 150,

    'eraser': 50}

for product, price in products.item():

    if 50<= price <= 100;

        Print (product)

## set: Adding and Removing Elements

22.
```
s = set()
#step1 : Add numbers 1 to 10
for num in range(1,11)
    s.add(num)
for num in range(1,11)
    if num% 3 ==0
    s.remove(num)
print(s)
```

Topic, Unpacking, For loop

23.
```
list = [('Alice', 25), ('Bob', 35), ('Carol', 40)]
for name, age in list:
    if age>30:
        Print(name)
```

List, For loop, if-Else

24. list = ['cat', 'apple', 'dog', 'banana']

```
for char in list:
    if len(char) > 3 and 'a' in char:
        Print (char)
```

Dictionary, Nested Dictionaries

25. employees = {

```
    'e1': {'name': 'Alice', 'salary': 60000}
    'e2': {'name': 'Bob', 'salary': 45000}
    'e3': {'name': 'Carol', 'Salary': 75000}
}

for emp_id, details in employees.items():
    if details['Salary'] > 50000
        Print (details['name'], details['salary'])
```

CamScanner

String, For Loop

26. text = "Programming"

```
for i in range(len(text)):
    if i % 2 == 1:
        Print (text[i], end = "")
```

Set, Membership, If-Else

27. text = "hello"

```
seen = set()
for char in text:
    if char in seen:
        Print ("Duplicates Found")
        break
    else:
        seen.add(char)
else:
    Print ("NO duplicates")
```

Lists, For loop, Sum

28.
```
def is_prime(n):
    if n< 2:
        return False
    for i in range (2,n):
        if n%i ==0
            return False
    return True

list = [2,3,4,5,6,7,8,9]
sum_of_primes = 0

for num in list:
    if is_prime(num):
        sum_of_primes += num
print (sum_of_primes)
```

Dictionary, Keys, Values

29. data = {'a': 5, 'b': 7, 'c': 3}

max_key = None

max_value = float('-inf')

for key, value in data.items():

    if value > max_value:

        max_value = value

        max_key = key

Print (max_key)

Tuple, immutability, for loop

30. data = ([1,2], [3,4], [5])

append_value = 99

for inner_list in data:

    inner_list.append(append_value)

Print (data)

## Strings, If-Else

31.
```
text = "12345"
if text.isdigit():
    Print ("Digits only")
else:
    Print ("Contains other characters")
```

## List, Insert, Remove

32.
```
numbers = [10, 20, 30]
numbers.insert (1, 15)
numbers.remove (30)
Print (numbers)
```

index → object

## Dictionary, popitem

33.
```
data = {'a':1, 'b':2, 'c':3}
last_item = data.popitem()
Print (last_item)
```

CamScanner

## Set, For loop

34.
```
set1 = {1,2,3,4}
set2 = {2,4}
common_elements = set1 & set2

for num in common_elements:
    if num%2 ==0
        Print(num, end= " ")
```

## Strings, Count Method

35.
```
text = "Mississippi"
char_to_count = 'S'

# convert both to lowercase for case insensitive counting
count = text.lower().count(char_to_count.lower())
output (count)
```

## List, For loop, If-else

36.
```
numbers = [1, -2, 3, -4, 5]
# Loop through the list with indices
for i in range(len(numbers)):
    if numbers[i] < 0: # check if the number is negative
        print(i, end=" ")
```

## Tuple, length

37.
```
numbers = (1, 2, 2, 3, 4, 4, 4)
# total elements
total_elements = len(numbers)
# unique elements [convert tuple to set]
unique_elements = len(set(numbers))

print("Total_elements:", total_elements)
print("unique_elements:", unique_elements)
```

## Dictionary, for loop

38. data = {'a': 1, 'b': 'Hello', 'c': 3.5, 'd': 'World'}

```
# loop through dictionary
for key in data
    if isinstance(data[key], str)   #check if the value is a string
        print [key]
```

## Sets, clear

39. numbers = [1, 2, 3, 4]

```
my_set = set(numbers)
#clear the set
my_set.clear()
print(my_set)
```

## For loop, Range

40. total = 0

```
for i in range (1, 50)
    if i % 2 != 0
        total += i
print (total)
```

CamScanner

Strings, Replace

41.
```
text = "Hello world"
vowels = "aeiouAEIOU"
for v in vowels:
    text = text.replace(v, '*')
Print(text)
```

List, Reverse, For loop

42.
```
numbers = [1,2,3,4,5]
reversed_list = []
#loop through the list from the end to the start
for i in range(len(numbers)-1,-1,-1):
    reversed_list.append(numbers[i])
Print(reversed_list)
```

Dictionary, Update

43.
```
dict1 = {'a':1}
dict2 = {'b':2, 'a':3}
dict1.update(dict2)
Print(dict1)
```

## Sets, Symmetric difference

44.
```
set1 = {1,2,3}
set2 = {2,4,6}
result = set1.symmetric_difference(set2)
print(result)
```

## Tuple, For Loop, if-else

45.
```
numbers = (2,4,6,8)
all_even = True
for num in numbers:
    if num % 2 != 0:
        all_even = False
        break
if all_even:
    print("All even")
else:
    print("Not all even")
```

String, For loop

46.
```
text = "Python"
for i in range(len(text)):
    if i % 2 == 0
        print(text[i], end = "")
```

List, For loop, If-else

47.
```
words = ['madam', 'apple', 'racecar', 'python']
for word in words:
    if word == word[::-1]
        print(word)
```

Dictionary, Keys and values

48.
```
data = {'a': 1, 'b': 2}
swapped = {value: key for key, value in data.items()}
print(swapped)
```

Set, For loop

49.
```
words = {'apple', 'bat', 'balloon', 'cat'}
for word in words:
    if len(word) > 4
        print(word)
```

CamScanner

If-else, For loop, lists

50.
```
numbers = [4, -3, 2, -1, 0]
for i in range(len(numbers)):
    if numbers[i] < 0:
        numbers[i] = 0
print(numbers)
```

# Section-B (DJM)

```
words = ("this", "is", "fine")
result = " "
for i in range(len(words)):
    result += words[i]
    if i != len(words) - 1:
        result += "-"
print(result)
```

Output:
this-is-fine.

Ejaj Ahmed
Zemon