COMSATS University Islamabad, Wah Campus
Electrical & Computer Engineering Department

Lab Rubrics Evaluation sheet

Fall 2023

Program: BCS **Section: 7C**

Subject: Artificial Intelligence **Reg #: FA20-BCS-073, FA20-BCS-157**

Lab No: 03 **Name: Aleena Khan, Fatima Zamir**

Lab Instructor: Engr.Adnan Saleem Mughal **Date: September 27, 2023**

Title of experiment: Control Flow Statements and Functions in Python

- **Objectives:**
  - The objective of these lab tasks is to help us practice specific aspect of Python programming, such as loops, conditionals, functions, and string manipulation.
  - Showcase practical application through completed lab tasks.
- **Tools:**

  I have utilized Spyder.

LAB TASKS

Question No: 01

Write a Python program to find those numbers which are divisible by 7 and multiple of 5, between 1500 and 2700 (both included).

Code:

```
result = []

for number in range(1500, 2701):
```

```
    if number % 7 == 0 and number % 5 == 0:
        # If both conditions are met, add the number to the result list
        result.append(number)
print("Numbers divisible by 7 and multiples of 5 between 1500 and 2700 are: ")
print(result)
```

```
Numbers divisible by 7 and multiples of 5 between 1500 and 2700 are:
[1505, 1540, 1575, 1610, 1645, 1680, 1715, 1750, 1785, 1820, 1855, 1890, 1925, 1960,
    1995, 2030, 2065, 2100, 2135, 2170, 2205, 2240, 2275, 2310, 2345, 2380, 2415, 2450
    , 2485, 2520, 2555, 2590, 2625, 2660, 2695]
```

Question No: 02

Write a Python program to convert temperatures to and from Celsius, Fahrenheit. [Formula: c/5 = f-32/9 [where c = temperature in Celsius and f = temperature in Fahrenheit]

Code:

```
def celsius_to_fahrenheit(celsius):
    fahrenheit = (celsius * 9/5) + 32
    return fahrenheit
def fahrenheit_to_celsius(fahrenheit):
    celsius = (fahrenheit - 32) * 5/9
    return celsius
celsius_temp = float(input("Enter temperature in Celsius: "))
fahrenheit_result = celsius_to_fahrenheit(celsius_temp)
print(f"{celsius_temp}°C is {fahrenheit_result}°F")
```

```
fahrenheit_temp = float(input("Enter temperature in Fahrenheit: "))

celsius_result = fahrenheit_to_celsius(fahrenheit_temp)

print(f"{fahrenheit_temp}°F is {celsius_result}°C")
```

```
Enter temperature in Celsius: 60
60°C is 140.0 in Fahrenheit
Enter temperature in Fahrenheit: 45
45°F is 7 in Celsius
```

Question No: 03

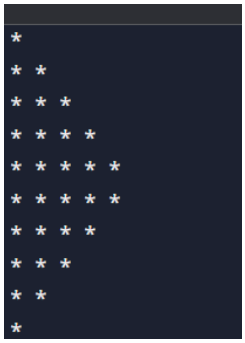
Write a Python program to construct the following pattern, using a nested for loop.

```
*
* *
* * *
* * * *
* * * * *
 * * * *
* * *
* *
*
```

Code:

```
rows = 5

for i in range(rows):

    for j in range(i+1):

        print("*" , end = " ")

    print("")

for i in range(rows):

    for j in range(rows-i):
```

```
    print("*" , end = " ")
print("")
```



Question No: 04

Write a Python program to count the number of even and odd numbers from a series of numbers.

Code:

```
even_count = 0
odd_count = 0
numbers = input("Enter a series of numbers (comma-separated): ")
number_list = [int(num) for num in numbers.split(',')]
for num in number_list:
    if num % 2 == 0:
        even_count += 1
    else:
        odd_count += 1
```

```
print(f"Even numbers: {even_count}")

print(f"Odd numbers: {odd_count}")
```

```
Enter a series of numbers (comma-separated): 11, 2, 19, 23, 8, 90, 17
Even numbers: 3
Odd numbers: 4
```

Question No: 05

Write a Python program that prints each item and its corresponding type from the following list.

Code:

```
datalist = [23, 11.23, 8+9j, True, 'string', (0, -1), [7, 31], {"class": 'V', "section": 'A'}]

for item in datalist:

    print(f"{item} is of type {type(item).__name__}")
```

```
23 is of type int
11.23 is of type float
(8+9j) is of type complex
True is of type bool
string is of type str
(0, -1) is of type tuple
[7, 31] is of type list
{'class': 'V', 'section': 'A'} is of type dict
```

Question No: 06

Write a Python program to get the Fibonacci series between 0 to 50.

Code:

```
fibonacci_sequence = [0, 1]

a, b = 0, 1
```

```
while b <= 50:
    a, b = b, a + b
    if b <= 50:
        fibonacci_sequence.append(b)
print("Fibonacci sequence between 0 and 50:")
print(fibonacci_sequence)
```

```
Fibonacci sequence between 0 and 50:
[0, 1, 1, 2, 3, 5, 8, 13, 21, 34]
```

Question No: 07

Write a Python program to check the validity of password input by users. Validation : a. At least 1 letter between [a-z] and 1 letter between [A-Z]. b. At least 1 number between [0-9]. c. At least 1 character from [$#@]. d. Minimum length 6 characters. e. Maximum length 16 characters.

Code:

```
import re
def is_valid_password(password):
    if 6 <= len(password) <= 16:
        if re.search(r'[a-z]', password) and re.search(r'[A-Z]', password) and re.search(r'[0-9]',
password):
            if re.search(r'[$#@]', password):
                return True
    return False
password = input("Enter a password: ")
```

```
if is_valid_password(password):
    print("Password is valid.")
else:
    print("Password is invalid.")
```

```
Enter a password: lostLight@126789
Password is valid.
```

Question No: 08

Write a Python program to check a string represent an integer or not.

Code:

```
string = input("Input a string: ")
if string.isdigit() or (string[0] == '-' and string[1:].isdigit()):
    print("The string represents an integer.")
else:
    print("The string is not an integer.")
```

```
Input a string: Autumn23
The string is not an integer.
```

Question No: 09

Write a Python function to find the Max of three numbers.

Code:

```
def find_max_of_three(num1, num2, num3):
    if num1 >= num2 and num1 >= num3:
        return num1
```

```
    elif num2 >= num1 and num2 >= num3:
        return num2
    else:
        return num3
num1 = int(input("Enter the first integer: "))
num2 = int(input("Enter the second integer: "))
num3 = int(input("Enter the third integer: "))
maximum = find_max_of_three(num1, num2, num3)
print(f"The maximum of {num1}, {num2}, and {num3} is:  {maximum}")
```

```
Enter the first integer: 23
Enter the second integer: 14
Enter the third integer: 31
The maximum of 23, 14, and 31 is:  31
```

Question No: 10

Write a Python program to reverse a string.

Sample String : "1234abcd" Expected Output : "dcba4321"

Code:

```
input_string = input("Enter a string: ")
reversed_string = input_string[::-1]
print("Reversed string:", reversed_string)
```

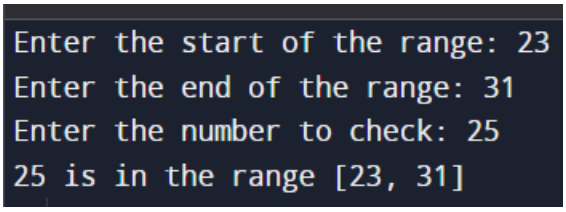
```
Enter a string: lostlight
Reversed string: thgiltsol
```

Question No: 11

Write a Python function to check whether a number is in a given range.

Code:

```
def is_number_in_range(number, start, end):
    return start <= number <= end


start_range = int(input("Enter the start of the range: "))
end_range = int(input("Enter the end of the range: "))
number_to_check = int(input("Enter the number to check: "))
if is_number_in_range(number_to_check, start_range, end_range):
    print(f"{number_to_check} is in the range [{start_range}, {end_range}]")
else:
    print(f"{number_to_check} is not in the range [{start_range}, {end_range}]")
```



Question No: 12

Write a Python function that accepts a string and calculate the number of upper case letters and lower case letters.

Code:

```
def count_upper_lower(string):
    upper_count = 0
    lower_count = 0
```

```
    for char in string:
        if char.isupper():
            upper_count += 1
        elif char.islower():
            lower_count += 1
    return upper_count, lower_count
input_string = input("Enter a string: ")
upper_count, lower_count = count_upper_lower(input_string)
print(f"No. of Upper case characters: {upper_count}")
print(f"No. of Lower case characters: {lower_count}")
```

```
Enter a string: Violets are Blue
No. of Upper case characters: 2
No. of Lower case characters: 12
```

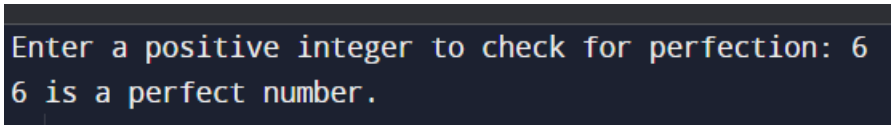
Question No: 13

Write a Python function to check whether a number is perfect or not. According to Wikipedia : In number theory, a perfect number is a positive integer that is equal to the sum of its proper positive divisors, that is, the sum of its positive divisors excluding the number itself (also known as its aliquot sum). Equivalently, a perfect number is a number that is half the sum of all of its positive divisors (including itself).

Code:

```
def is_perfect_number(number):
```

```
    if number <= 0:
        return False

    divisors_sum = 0
    for i in range(1, number):
        if number % i == 0:
            divisors_sum += i
    return divisors_sum == number
num = int(input("Enter a positive integer to check for perfection: "))
if is_perfect_number(num):
    print(f"{num} is a perfect number.")
else:
    print(f"{num} is not a perfect number.")
```



Question No: 14

Write a Python function that checks whether a passed string is palindrome or not.

Code:

```
def is_palindrome(string):
    cleaned_string = string.replace(" ", "").lower()
    return cleaned_string == cleaned_string[::-1]
input_string = input("Enter a string to check for palindrome: ")
```

```
if is_palindrome(input_string):
    print(f"'{input_string}' is a palindrome.")
else:
    print(f"'{input_string}' is not a palindrome.")
```

```
Enter a string to check for palindrome: madam
'madam' is a palindrome.
```

Question No: 15

Write a Python program to access a function inside a function.

Code:

```
def outer_function():
    print("This is the outer function.")
    def inner_function():
        print("This is the inner function.")
    inner_function()
outer_function()
```

```
This is the outer function.
This is the inner function.
```

Question No: 16

Write a recursive function to calculate the sum of numbers from 0 to 10.

Code:

```
def recursive_sum(n):
    if n == 0:
        return 0
    else:
        return n + recursive_sum(n - 1)
result = recursive_sum(10)
print("The sum of numbers from 0 to 10 is:", result)
```