AIR UNIVERSITY

Department of Electrical and Computer Engineering

Lab #1: Introduction to Python

Lab Instructor: Muhammad Awais

Objective

The Introduction to Programming using Python Practice Lab will provide you with necessary skills with programming using Python.

By completing the lab tasks, you will improve your practical skills in performing operations using data types and operators, controlling flow with decisions and loops, performing input and output operations, documenting and structuring code, performing troubleshooting and error handling and performing operations using modules and tools.

For Loop

The for loop in Python is a control flow statement that is used for iterating over a sequence.
Syntax:
for **variable** in **sequence**:

```
for i in range (1,11):
    print(f"{5} X {i} = {5*i}")
```

```
 5 X 1 = 5
 5 X 2 = 10
```

```
5 X 3 = 15
5 X 4 = 20
5 X 5 = 25
5 X 6 = 30
5 X 7 = 35
5 X 8 = 40
5 X 9 = 45
5 X 10 = 50
```

Conditional Statement

The conditional statement in Python allows to control the flow of program based on certain condition.

Syntax:

if **condition**:

```
x = 3

if x > 5:
    print("x is greater than 5")
else:
    print("x is not greater than 5")
```

```
 x is not greater than 5
```

Lab Tasks

Task 1: Write a code that takes the input from User, calculate Factorial and print the answer.

```
def factorial(n):
    if n == 0:
        return 1
    else:
        return n * factorial(n-1)

try:
    num = int(input("Enter a number to calculate its factorial: "))
    if num < 0:
        print("Factorial is not defined for negative numbers.")
    else:
        result = factorial(num)
        print(f"The factorial of {num} is: {result}")
except ValueError:
    print("Please enter a valid integer.")
```

```
 Enter a number to calculate its factorial: 3
 The factorial of 3 is: 6
```

Task 2: Write a code that takes the input from User, check the value is even positive, Even Negative, Odd Positive or Odd Negative.

```
try:
    num = int(input("Enter a number: "))

    if num > 0:
        if num % 2 == 0:
            print("Even Positive")
        else:
            print("Odd Positive")
    elif num < 0:
        if num % 2 == 0:
            print("Even Negative")
        else:
            print("Odd Negative")
    else:
        print("Zero is neither positive nor negative.")
except ValueError:
    print("Please enter a valid integer.")
```

```
 Enter a number: 4
 Even Positive
```

Task 3:Create a program that calculates the average grade for a list of students. The program should perform the following tasks:

Ask the user for the number of students they want to calculate grades for at least 3.

For each student:

1. Prompt the user to enter the student's name.
2. Use a for loop to iterate through a predefined set of subjects (e.g., Math, Science, English).
3. For each subject, ask the user to input the grade (assume grades are on a scale of 0 to 100).
4. Calculate the average grade for each student.

Use a conditional statement to determine the overall grade for each student based on the average:

1. If the average is 90 or above, assign the grade 'A'.

2. If the average is between 80 and 89, assign the grade 'B'.
3. If the average is between 70 and 79, assign the grade 'C'.
4. If the average is between 60 and 69, assign the grade 'D'.
5. If the average is below 60, assign the grade 'F'.

Store the student names, their average grades, and corresponding letter grades in a list of dictionaries.

Create a function named display_results that takes the list of dictionaries as an argument and prints out the results in a readable format, including each student's name, average grade, and letter grade.

```
def calculate_average_grade(num_students):

  subjects = ["Math", "Science", "English"]
  student_data = []

  for i in range(num_students):
    student_name = input(f"Enter student {i + 1} name: ")
    student_grades = {}
    for subject in subjects:
      while True:
        try:
          grade = int(input(f"Enter {student_name}'s grade in {subject}: "))
          if 0 <= grade <= 100:
            student_grades[subject] = grade
            break
          else:
            print("Invalid grade. Please enter a number between 0 and 100.")
        except ValueError:
          print("Invalid input. Please enter a valid number.")

    total_grade = sum(student_grades.values())
    average_grade = total_grade / len(subjects)

    if average_grade >= 90:
      letter_grade = "A"
    elif average_grade >= 80:
      letter_grade = "B"
    elif average_grade >= 70:
      letter_grade = "C"
    elif average_grade >= 60:
      letter_grade = "D"
    else:
      letter_grade = "F"

    student_data.append({
      "name": student_name,
      "average_grade": average_grade,
      "letter_grade": letter_grade
    })
```

```
    return student_data

def display_results(student_data):

    print("\nStudent Results:")
    for student in student_data:
        print(f"Name: {student['name']}")
        print(f"Average Grade: {student['average_grade']:.2f}")
        print(f"Letter Grade: {student['letter_grade']}")
        print("---")

# Get the number of students from the user
while True:
    try:
        num_students = int(input("Enter the number of students (at least 3): "))
        if num_students >= 3:
            break
        else:
            print("Please enter a number greater than or equal to 3.")
    except ValueError:
        print("Invalid input. Please enter a valid number.")

# Calculate and display the results
student_data = calculate_average_grade(num_students)
display_results(student_data)
```

```
Enter the number of students (at least 3): 3
Enter student 1 name: Ahmed
Enter Ahmed's grade in Math: 95
Enter Ahmed's grade in Science: 65
Enter Ahmed's grade in English: 80
Enter student 2 name: Majid
Enter Majid's grade in Math: 50
Enter Majid's grade in Science: 45
Enter Majid's grade in English: 30
Enter student 3 name: Aqib
Enter Aqib's grade in Math: 70
```

```
Enter Aqib's grade in Science: 90
```

Conclusions

```
Name: Ahmed
```

In this AI lab, we implemented three Python programs to demonstrate essential programming concepts:

Factorial Calculation: Developed a program to compute the factorial of a user-input number using recursion, ensuring input validation.

Number Classification: Created a program to classify user-entered numbers as even positive, even negative, odd positive, or odd negative, employing conditional statements for classification.

Student Grade Calculation: Designed a program to calculate and classify average grades for a list of students, utilizing loops, conditional statements, and data structures to organize and present the results effectively.