| Ex No.02 | USES OF CONTROL | Reg.No. URK24CS9068 |
| 10.01.25 | STATEMENTS IN PYTHON | |

1. Write a Python program that takes a student's score as input and assigns a grade based on the following:

   • **Score >= 90: Grade A**

   • **Score >= 75: Grade B**

   • **Score >= 60: Grade C**

   • **Score < 60: Grade D**

   Use if, elif, and else statements to implement this.

Aim: To develop a Python program that accepts a student's score as input and assigns a grade based on predefined criteria using conditional statements.

Objective: To implement a Python program using conditional statements to classify student scores into grades based on predefined criteria.

Algorithm:

   Step 1: Start

   Step 2: Input the student's score.

   Step 3: Check the score using conditional statements:

   • If the score is 90 or above, assign Grade A.

   • Else if the score is 75 or above, assign Grade B.

   • Else if the score is 60 or above, assign Grade C.

   • Else, assign Grade D.

   Step 4: Display the assigned grade.

   Step 5: End

Program:

```python
#Grading System
print("D. Brian Gabriel")
print("URK24CS9068")
print("------------------")
score=int(input("Enter your score in digits between 0 to 100"))
if(score>100 or score<0):
    print("Invalid Score")
if(score>=90):
    print("Your Grade is : A")
elif(score>=80):
    print("Your Grade is : B")
elif(score>=70):
    print("Your Grade is : C")
else:
    print("Your Grade is : D")
```

✓ 16.3s

Output:

```
D. Brian Gabriel
URK24CS9068
------------------
Your Grade is : C
```

Result:

The Program correctly assigns grades based on the student's score using conditional statements and displays the appropriate grade.

2. Write a Python program that takes a number from the user and checks if it is even or odd. If the number is even, print "Even", otherwise print "Odd".

Aim: To develop a Python program that takes a number as input and determines whether it is even or odd using conditional statements.

Objective: To implement a Python program using conditional statements to check whether a given number is even or odd.

Algorithm:

  Step 1: Start

  Step 2: Input a number from the user.

  Step 3: Check if the number is divisible by 2:

   • If number % 2 == 0, print "Even".

   • Else, print "Odd".

  Step 4: End

Program:

```python
#odd or even
print("D. Brian Gabriel")
print("URK24CS9068")
print("-----------------")
n=int(input("Enter a no."))
if(n%2==0):
    print(n,"is an even no.")
else:
    print(n,"is an odd no.")
```

Output:

```
D. Brian Gabriel
URK24CS9068
------------------
Enter a no. 43
43 is an odd no.
```

Result: The program correctly determines whether a given number is even or odd and displays the appropriate output based on the user's input.

3. Write a Python program that takes an integer input and checks if it is a prime number. A prime number is a number that is greater than 1 and has no divisors other than 1 and itself.

Ain: To develop a Python program that takes an integer as input and checks whether it is a prime number using conditional statements and loops**.**

Objective: To implement a Python program that determines whether a given integer is a prime number by checking its divisibility, using loops and conditional statements.

Algorithm:

Step 1: Start

Step 2: Input an integer from the user.

Step 3: If the number is less than or equal to 1, print "Not a prime number" and stop.

Step 4: Initialize a loop from 2 to (number - 1):

- If the number is divisible by any value in this range, print "Not a prime number" and stop.

Step 5: If no divisors are found, print "Prime number".

Step 6: End

Program:

```python
#odd or even
print("D. Brian Gabriel")
print("URK24CS9068")
print("-----------------")
n=int(input("Enter a no."))
flag=0
if (n>1):
    for i in range(2,n):
        if (n%i==0):
            print(n,"is not a prime no.")
            flag=1
            break
    if(flag==0):
        print(n,"is a prime no.")
else:
    print(n,"is not a prime no.")
```

Output:

```
D. Brian Gabriel
URK24CS9068
------------------
Enter a no. 547
547 is a prime no.
```

Result: The program correctly determines whether the given integer is a prime number by checking its divisibility and displays the appropriate output based on the user's input.

4. Write a Python program that prints the Fibonacci sequence up to a given number n. The Fibonacci sequence is defined as:

• **0, 1, 1, 2, 3, 5, 8, 13, 21, ... Use a for or while loop to generate the sequence.**

Aim: To develop a Python program that generates and prints the Fibonacci sequence up to a given number n using a loop.

Objective: To implement a Python program that generates the Fibonacci sequence up to a given number **n** using iterative loops, demonstrating the use of loops and sequence generation.

Algorithm:

Step 1: Start

Step 2: Input the value of n (the limit for the Fibonacci sequence).

Step 3: Initialize two variables:

- a = 0 (first term)
- b = 1 (second term)

Step 4: Print the first two terms (0 and 1).

Step 5: Use a loop to generate the next terms:

- Calculate the next term as next_term = a + b.
- Update a = b and b = next_term.
- Repeat until the next term exceeds n.

Step 6: End

Program:

```python
#Fibonacci series
print("D. Brian Gabriel")
print("URK24CS9068")
print("------------------")
n=int(input("Enter a no."))
a,b=0,1
print(f"The fibonacci series for {n} is:")
while a<=n:
    print(a,end=" ")
    a,b=b,a+b
```

Output:

```
D. Brian Gabriel
URK24CS9068
------------------
Enter a no. 7
The fibonacci series for 7 is:
0 1 1 2 3 5
```

Result: The program successfully generates and prints the Fibonacci sequence up to the given number n, demonstrating the use of loops for sequence generation.

5. Write a program that generates the multiplication table for a given number n. The table should be displayed for the numbers 1 through 10.

Aim: To develop a Python program that generates and displays the multiplication table for a given number n from 1 to 10 using a loop.

Objective: To implement a Python program that takes an integer input and prints its multiplication table using a loop, demonstrating iteration and arithmetic operations.

Algorithm:

Step 1: Start

Step 2Input the number n from the user.

Step 3: Use a loop to iterate from 1 to 10:

- Multiply n by the loop variable.

- Print the result in the format: n × i = result.

Step 4: End

Program:

```python
#odd or even
print("D. Brian Gabriel")
print("URK24CS9068")
print("------------------")
n=int(input("Enter a number"))
print(f"Multiplication table of {n} is:")
for i in range(0,11,1):
    print(n,"x",i,"=",n*i)
```

Output:

```
D. Brian Gabriel
URK24CS9068
------------------
Enter a number 7
Multiplication table of 7 is:
7 x 0 = 0
7 x 1 = 7
7 x 2 = 14
7 x 3 = 21
7 x 4 = 28
7 x 5 = 35
7 x 6 = 42
7 x 7 = 49
7 x 8 = 56
7 x 9 = 63
7 x 10 = 70
```

Result: The program correctly generates and displays the multiplication table for the given number n, ensuring accurate calculations using a loop.

6. **Write a Python program that takes three numbers as input from the user and finds the largest among them using if statements.**

Aim: To develop a Python program that takes three numbers as input and determines the largest among them using if statements.

Objective: To develop a Python program that takes three numbers as input and determines the largest among them using if statements.

Algorithm:

    Step 1: Start

    Step 2: Input three numbers from the user as num1, num2, and num3.

    Step 3: Use if statements to compare the numbers:

- If num1 is greater than both num2 and num3, print num1 as the largest.

- Else if num2 is greater than both num1 and num3, print num2 as the largest.

- Else, print num3 as the largest.

    Step 4: End

Program:

```python
#odd or even
print("D. Brian Gabriel")
print("URK24CS9068")
print("------------------")
a=int(input("Enter 1st no."))
b=int(input("Enter 2nd no."))
c=int(input("Enter 3rd no."))
if a>=b and a>=c:
    print(f"{a} is the largest no.")
elif b>=a and b>=c:
    print(f"{b} is the largest no.")
else:
    print(f"{c} is the largest no.")
```

Output:

```
D. Brian Gabriel
URK24CS9068
------------------
Enter 1st no. 4
Enter 2nd no. 5
Enter 3rd no. 2
5 is the largest no.
```

Result: The program correctly identifies and displays the largest number among the three user inputs using conditional statements.