

**Note: Submit Softcopy of Each program with output**

1. Given an integer, n, perform the following conditional actions:  
If n is odd, print Weird  
If n is even and in the inclusive range of 2 to 5, print Not Weird  
If n is even and in the inclusive range of 6 to 20, print Weird  
If n is even and greater than 20, print Not Weird
2. Read two integers from user input and print three lines where:  
The first line contains the sum of the two numbers.  
The second line contains the difference of the two numbers (first - second).  
The third line contains the product of the two numbers.
3. Read two integers and print two lines. The first line should contain integer division,  $a // b$ . The second line should contain float division,  $a / b$ .  
You don't need to perform any rounding or formatting operations.
4. We add a Leap Day on February 29, almost every four years. The leap day is an extra, or intercalary day and we add it to the shortest month of the year, February.  
In the Gregorian calendar three criteria must be considered to identify leap years:  
The year can be evenly divided by 4, is a leap year, unless:  
The year can be evenly divided by 100, it is NOT a leap year, unless:  
The year is also evenly divisible by 400. Then it is a leap year.  
This means that in the Gregorian calendar, the years 2000 and 2400 are leap years, while 1800, 1900, 2100, 2200, 2300 and 2500 are NOT leap years.  
  
You are given the year, and you have to write a function to check if the year is leap or not.
5. Using a for loop, write a program that prints out the decimal equivalents of  $1/2$ ,  $1/3$ ,  $1/4$ , . . . ,  $1/10$ .
6. Write a program using a while loop that asks the user for a number, and prints a countdown from that number to zero. What should your program do if the user inputs a negative number? As a programmer, you should always consider “edge conditions” like these when you program! (Another way to put it- always assume the users of your program will be trying to find a way

to break it! If you don't include a condition that catches negative numbers, what will your program do?)

7. Write a program using a while loop that asks the user to enter a number that is divisible by 2. Give the user a witty message if they enter something that is not divisible by 2- and make them enter a new number. Don't let them stop until they enter an even number! Print a congratulatory message when they \*finally\* get it right.

8. Buggy loop

Consider the following program

```
n = 10
i = 10
while i > 0:
    print i
    if i % 2 == 0:
        i = i / 2
    else:
        i = i + 1
```

Draw a table that shows the value of the variables n and i during the execution of the program. Your table should contain two columns (one for each variable) and one row for each iteration. For each row in the table, write down the values of the variables as they would be at the line containing the print statement.

9. Factorial of given number (Take input from user)  
10. Prime Number between range 1 to 100  
11. Fibonacci series between range 0 to 100  
12. Reverse of a number (Take input from user)  
13. Number is Palindrome or not (Take input from user)  
14. Number is Armstrong or not (Take input from user)  
15. Print table of given number (Take input from user)

2 X 1 = 2

2 X 2 = 4

.

.

.

2 X 10 = 20

16. Print table from 1 to 10 in given format

1 2 3 4 .....10

2 4 6 8 .....20

.

.

.

.

9 18 27 36 .....90

10 20 30 40 .....100