SET A

1)Write a Python program that simulates a basic calculator, performing addition, subtraction, multiplication, and division.

```
print("CALCULATOR")
def add(x, y):
   return x + y

def subtract(x, y):
   return x - y

def multiply(x, y):
   return x * y

def divide(x, y):
   if y == 0:
      return "Error! Division by zero."
   return x / y

def calculator():
   print("Select operation:")
   print("1. Add")
   print("2. Subtract")
   print("3. Multiply")
   print("4. Divide")

   choice = input("Enter choice (1/2/3/4): ")

   if choice in ['1', '2', '3', '4']:
      num1 = float(input("Enter first number: "))
      num2 = float(input("Enter second number: "))

      if choice == '1':
         print(f"{num1} + {num2} = {add(num1, num2)}")
      elif choice == '2':
         print(f"{num1} - {num2} = {subtract(num1, num2)}")
      elif choice == '3':
         print(f"{num1} * {num2} = {multiply(num1, num2)}")
      elif choice == '4':
         print(f"{num1} / {num2} = {divide(num1, num2)}")
   else:
      print("Invalid input")
calculator()
```

2) **Write a Python program that converts a given decimal number to its binary equivalent.**

```
dec =int(input("Enter the decimal number:"))
binary =""
while dec>0:
    binary=str(dec%2)+binary
    dec=dec//2

print("binary equvalence of your number is:",binary)
```

3) **Write a Python program that asks for the user's age and then prints a message stating whether the user is a minor, an adult, or a senior.**

```
age =int(input("Enter your age :"))
if age < 18 :
   print("You are a minor")
elif age >= 18 and age < 60:
    print("You are a adult")
else :
   print("You are a senior")
```

4) **Write a Python program to swap the values of two variables without using a third variable.**

```
a =int(input("Enter your first number :"))
b =int(input("Enter your second number :"))
print(f"Before swapping the number first number ={a}, second number = {b}")
a = a+b
b = a-b
a = a-b
print(f"After swapping the number first number = {a}, second number = {b}")
```

5) **Write a Python program to print the first 10 numbers of the Fibonacci series.**

```
a = 1
b = 1
next_term = a+b
print(f"The numbers of series are :" )
for i in range(0,10):
   print(a,",",end ="")
   a = b
   b = next_term
   next_term = a+b
```

6)**Write a Python program to check if a given number is prime or not.**

```
n = int(input("Enter your positive number: "))
flag = 0

if n <= 0 or n == 1:
   flag = 1
else:
   for i in range(2, n- 1):
      if n % i == 0:
         flag = 1
         break

if flag == 0:
   print("It is a prime number")
else:
   print("It is not a prime number")
```

7)**Write a Python program that takes three numbers as input and checks if the third number is the sum of the first two numbers using logical operators.**

```
num1 = int(input("Enter first number: "))
num2 = int(input("Enter second number: "))
num3 = int(input("Enter third number: "))

if (num1 + num2 == num3) and (num3-num2==num1 or num3-num1==num2):
   print("The third number is the sum of the first two numbers.")
else:
   print("The third number is not the sum of the first two numbers.")
```

8)**Write a Python program that imports a custom module you created with a function that returns the factorial of a number.**

```
#File1(fact.py)

def fact(n):
   x = 1
   for i in range (1, n+1):
      x = x * i
   return x

#File 2(question8.py)

from fact import fact
n = int(input("Enter the number :"))
result = fact(n)
print("Factorial of your number is :",result)
```

9)Write a Python program that takes two numbers as input and performs division, handling the case where the divisor is zero.

```
a = int(input("Enter your first number: "))
b = int(input("Enter your second number: "))
if (b==0):
    print("Can't be divisible by zero")
else :
    c = a/b
    print(f"The division of your number is : {c}")
```

10) Write a Python function that takes a list of numbers and returns the maximum value in the list.

```
numbers = list(map(int, input("Enter numbers separated by spaces: ").split()))

if len(numbers) == 0:
    max_value = None
    else:
    max_value = numbers[0]
        for number in numbers:
        if number > max_value:
            max_value = number

print("The maximum value in the list is:", max_value)
```

11) **Write a Python function that takes a name and an optional age parameter and prints a greeting. If the age is not provided, it should default to 25.**

```
    name = input("Please enter your name: ")
    age_input = input("Please enter your age (press Enter to skip): ")
    age = int(age_input) if age_input else 25
    print(f"Hello, {name}! You are {age} years old.")
```

12) **Write a Python program to count the number of vowels in a given string.**

```
s = input("Please enter a string: ")
vowels = "aeiouAEIOU"
count = 0
for char in s:
    if char in vowels:
        count += 1
print(f"The number of vowels in the string is: {count}")
```

13) Write a Python program that prints a multiplication table up to (numberx10).

```
num =int(input("Enter the number which multiplication table you want :"))
for i in range (1,11):
    print(num,"X",i,"=",num*i)
```

14)Write a Python program to print a right-angled triangle of '*' with a given number of rows.

```
num_rows = int(input("Enter the number of rows: "))
for i in range(1, num_rows + 1):
   print('*' * i)
```

15)Write a Python program to print a pyramid of '*' with a given number of rows.

```
num_rows = int(input("Enter the number of rows: "))
for i in range(num_rows):
    print(' ' * (num_rows - i - 1), end='')
    print('*' * (2 * i + 1))
```

SET-B

1)Given an integer x, return true if x is a palindrome, and false otherwise. (LeetCode: Palindrome Number)

```
x = int(input("Enter an integer: "))
str_x = str(x)
if str_x == str_x[::-1]:
  print(f"{x} is a palindrome.")
else:
   print(f"{x} is not a palindrome.")
```

2)Given a non-empty array of integers nums, every element appears twice except for one. Find that single one. (LeetCode: Single Number)

```
nums = list(map(int, input("Enter numbers separated by spaces: ").split()))
single_number = 0
for num in nums:
   single_number ^= num
print(f"The single number in the array is: {single_number}")
```

3)**Given an array of integers nums and an integer target return indices of the two numbers such**

that they add up to target. You may assume that each input would have exactly one solution, and you may not use the same element twice. You can return the answer in any order. (LeetCode: Two Sum)

```
def two_sum(nums, target):
    num_map = {}
    for i, num in enumerate(nums):
        complement = target - num
        if complement in num_map:
            return [num_map[complement], i]

        num_map[num] = i
    return []

input_nums = input("Enter numbers separated by spaces: ").split()
nums = list(map(int, input_nums))
target = int(input("Enter the target sum: "))
result = two_sum(nums, target)
if result:
    print(f"Indices of the two numbers that add up to {target}: {result}")
else:
    print("No solution found.")
```

4) Write an algorithm to determine if a number n is happy. (LeetCode: Happy Number)

```
def is_happy(n):
    seen = set()
    while n != 1 and n not in seen:
        seen.add(n)
        n = sum(int(digit) ** 2 for digit in str(n))
    return n == 1
num = int(input("Enter a number to check if it's happy: "))
if is_happy(num):
    print(f"{num} is a happy number.")
else:
    print(f"{num} is not a happy number.")
```

5)Given an integer array nums, return true if any value appears at least twice in the array, and return false if every element is distinct. (LeetCode: Contains Duplicate)

```
def contains_duplicate(nums):
    seen = set()
    for num in nums:
        if num in seen:
            return True
        seen.add(num)
    return False

input_nums = input("Enter numbers separated by spaces: ").split()
nums = list(map(int, input_nums))
if contains_duplicate(nums):
    print("True.")
else:
    print("False.")
```