EX NO:1 PROGRAM USING OPERATORS AND EXPRESSIONS

1(a): Write a Python program that accepts an integer n and computes the value of n+nn+nnn

AIM

To write and execute a c program program that accepts an integer n and computes the value of n+nn+nnn

ALGORITHM

Step 1: Start the program

Step 2: Get the Input .

Step 3: Compute the n+nn+nnn.

Step 4: Display the result.

Step 5: Stop the program

PROGRAM

```
 n- int(input())
x=int("%s"%n)
y=int("%s%s"%(n,n))
z=int("%s%s%s"%(n,n,n))
print(x+y+z)
```

OUTPUT

```
5

615
```

1(b): Write a program to convert seconds into hours, minutes, and seconds.

AIM

To write and execute a program to convert seconds into hours, minutes, and seconds.

ALGORITHM

Step 1: Start the program

Step 2: Get the Inputs

Step 3: Convert seconds into hours, minutes, and seconds.

Step 4: Display the result.

Step 5: Stop the program

PROGRAM

```
n=int(input())
h=n//3600
m=(n%3600)//60
s=n%60
print(h, "hours, ", m, "minutes,", "and",s, "seconds")
```

OUTPUT

```
3600
1    hours,0 minutes and, 0 seconds
```

1(c): Write a program that takes the distance in kilometers as input and outputs the number of fullmiles (1 mile = 1 kilometer beyond the initial 3 kilometers) and the total cost of delivery.

AIM

To write and execute a program that takes the distance in kilometers as input and outputs the number of full miles (1 mile = 1 kilometer beyond the initial 3 kilometers) and the total cost of delivery.

ALGORITHM

Step 1: Start the program

Step 2: Get Input .
Step 3: Calculate the cost

Step 4: Display result.

Step 5: Stop the program

PROGRAM

```
d=int(input())

 x=d-3

 cost=400+(x*80)

  print(“Number of fullmiles:”,x)

 print(“Total cost of delivery:Rs.”,cost)
```

OUTPUT:

```
4

Number of full miles:1

Total cost of delivery: Rs.4
```

1(d): Write a program to count the number of apples and oranges bought.

AIM:

To write a program to count the number of apples and oranges bought.

ALGORITHM:

```
Step 1: Start the program
Step 2: Get the inputs
Step 3: Count the number of apples and oranges bought
Step 4: Print the output
Step 5: Stop the program
```

PROGRAM:

```
n=input().split()

 lis=[int(i) for I in n]

apple=lis[0]+lis[0]+lis[2]-lis[4]

orange=lis[1]+lis[1]+lis[3]-lis[5]

print(apple)

print(orange)
```

OUTPUT:

```
4 3 2 8 3 2
7
12
```

RESULT:

Thus, the above code using operators and expression has been executed and the output is verified successfully.

EX NO:2 PROGRAM USING CONDITIONAL & LOOPING

2(a):Write a program to calculate the bonus, tax amount and salary of an employees.

AIM:

To write a program to calculate the bonus, tax amount and salary of an employees.

ALGORITHM:

```
Step 1: Start
Step 2: Get the input
Step 3: Calculate the bonus, tax amount
and salary.
Step 4: Print the output.
Step 5: Stop
```

PROGRAM:

```
salary=float(input())

years_of_service=int(input())

bonus_percentage=float(input())

tax_percentage=float(input())

netbonus=bonus_percentage/100)*salary

 if years_of_service<5:
        print("Sorry, you are not eligible for a bonus.")
        netbonus=0
         tax_amount=(tax_percentage/100)*(salary+net_bonus)
         net_salary=salary+netbonus-tax_amount
 else:
        print("You have earned a bonus of %.1funits."%net_bonus)

        tax_amount=(tax_percentage/100)*(salary+net_bonus)

        net_salary=salary+net_bonus-tax_amount

print("Tax Amount:%.1f units"%tax_amount)

print("Net Salary:%.1f units"%net_salary)
```

OUTPUT:

```
50000.00
6
10.00
15.00
You have earned a bonus of
5000.0 unitsTax Amount:8250.0
units
Net Salary:46750.0 units
```

2(b): Write a program to check the secret code entered by students.

AIM:

To write a program to check the secret code entered by students.

ALGORITHM:

Step 1: Start the program
Step 2: Get the inputs
Step 3: Check the secret code entered
Step 4: Print the output

Step 5: Stop the program

PROGRAM:

```
secret_code=int(input())

first=int(str(secret_code)[0])

last=int(str(secret_code)[-3:])

if(last%first)==0:

      print(“Logged In”)

else:

    print(“Incorrect code”)
```

OUTPUT:

```
5345
```

2(c): Write a program to understand the behavior of a loop with an else clause.

AIM:

To write a program to understand the behavior of a loop with an else clause.

ALGORITHM:

```
Step 1: Start the program
Step 2: Get the inputs
Step 3: Check the secret code entered
Step 4: Print the output

Step 5: Stop the program
```

PROGRAM:

```
n=int(input())

m=0

while(m<n):

    print(“Inside loop”)

     m+=1

else:

     print(“Inside else”)
```

OUTPUT:

```
4
Inside loop
Inside loop
Inside loop
Inside loop
Inside else
```

2(d): Write a program to find the coordinate geometry for the given input.

AIM:

To write a program to find the coordinate geometry for the given input.

ALGORITHM:

Step 1: Start
Step 2: Get the inputs
Step 3: Find the coordinate
geometry.
Step 4: Print the output

Step 5: Stop the program

PROGRAM:

```
x=int(input())

y=int(input())

if x>0 andy>0:

    print(“The coordinate point lies in the first quandrant.”)

     elif x<0 andy>0:

    print(“The coordinate point lies in the second quandrant.”)

     elif x<0 andy<0:

    print(“The coordinate point lies in the third quandrant.”)

    elif x>0 andy<0:

    print(“The coordinate point lies in the fourth quandrant.”)

    else:

                print(“The coordinate point lies at the origin.”)
```

OUTPUT:

```
1
2
The coordinate point lies in the first quandrant.
```

RESULT:

Thus, the above code using condition and looping has been executed and the output is verified successfully

EX NO:3 PROGRAMS USING FUNCTION

3(a): Write a program that calculates the length of the hypotenuse of a right triangle

AIM:

To write a program that calculates and displays the length of the hypotenuse of right triangle.

ALGORITHM:

```
Step 1: Start the program ,import math module.
Step 2: Get two inputs
Step 3: calculate the hypotenuse using sqrt(a**2+b**2).
Step 4: Print the output.

Step 5: End.
```

PROGRAM:

```
import math
 def     calculate_hypotenuse(a,

b):


  hypotenuse = math.sqrt(math.pow(a, 2) + math.pow(b, 2))
  return hypotenuse

def main():
    a = float(input())
    b = float(input())
    hypotenuse = calculate_hypotenuse(a, b)
    print(f"The length of the hypotenuse is: {hypotenuse:.2f}")

if _name_ == "_main_":
    main()
```

OUTPUT

```
The length of hypotenuse us:5.26
```

3(b): Write a program to generate the Fibonacci sequence for a given number of terms. Calculate the factorial of a user-provided number."

AIM:

To write a program that calculates and displays the Fibonacci sequence for a given number of terms.Calculate the factorial of a user-provided number.

ALGORITHM

Step 1: Start the program ,
Step 2: Get the inputs
Step 3: calculate the Fibonacci sequence and the factorial of a user-provided number
Step 4: Print the output.

Step 5: End.

PROGRAM:

```
  def fibonacci_sequence(n):
      fib_sequence = []
      a, b = 0, 1
     for _ in range(n):
       fib_sequence.append(a)
       a, b = b, a + b
   return fib_sequence

def factorial(num):
   if num < 0:
     return None
 elif num == 0:
    return 1
else:
   return num * factorial(num - 1)

def main():
  n = int(input())
 if n <= 0:
    print("Please enter a positive integer")
else:
   print("Fibonacci sequence:")
  fib_seq = fibonacci_sequence(n)
  for term in fib_seq:
      print(term)

 m = int(input())
 if m < 0:
    print("Sorry, factorial does not exist for negative numbers")
else:
   print(f"The factorial of {m} is {factorial(m)}")

if _name_ == "_main_":
main()
```

OUTPUT

```
5
4
Fibonacci sequence:
0
1
1
2
3
The factorial of 4 is 24
```

3(c):

Write a program that determines the maximum discount a customer can receive based on their purchase amount. The program takes the purchase amount as input, applies a discount of 10% of the purchase amount (up to a maximum of $100), and prints the maximum discount amount.

AIM:

To write a program that determines the maximum discount a customer can receive.

ALGORITHM

Step 1: Start the program ,
Step 2: Get the inputs
Step 3: calculate the maximum discount .
Step 4: Print the output.

Step 5: End.

PROGRAM:

```
    def calculate_discount(purchase_amount):
      discount = min(purchase_amount * 0.1, 100)
      return discount

  def main():
       purchase_amount = float(input())
      max_discount = calculate_discount(purchase_amount)
     print(f"{max_discount:.1f}")

 if _name_ == "_main_":
main()
```

OUTPUT

```
50.57


5.1
```

3(d): Write a program to calculating the last digit of a Fibonacci number at a specific position.

Fibonacci numbers are a ,typically starting with 0 and 1.

AIM:

To write a to calculating the last digit of a Fibonacci number at a specific position.

ALGORITHM

Step 1: Start the program ,
Step 2: Get the inputs
Step 3: calculate the last digit of Fibonacci series.
Step 4: Print the output.

Step 5: End.

PROGRAM:

```
    def fibo_last_digit(n):
    fib = [0, 1]
   for i in range(2, n + 1):
       fib.append((fib[i - 1] + fib[i - 2]) % 10)
      return fib[n]

 def main():
     n = int(input())
    result = fibo_last_digit(n)
    print(result)

if _name_ == "_main_":
    main()
```

OUTPUT

```
    Input 1:

     1
    Output 1 :
     1
    Input 2 :
    9
   Output 2 :
    4
```

RESULT:

Thus, the above code using functions has been executed and the output is verified successfully

EX NO:4 PROGRAM USING OPERATORS AND EXPRESSIONS

4(a). Write a Python program to find the longest word in a given sentence and return the longest word. If multiple words are of the same length, return the first occurring word of maximum length.

AIM

To write a program to find the longest word in a given sentence and return the longest word..

ALGORITHM

Step 1: Start..

Step 2: Get the Input .

Step 3: compute the longest word using code.

Step4: Display result.

Step 5: Stop.

PROGRAM

```
sentence = input()
words = sentence.split()
max_length = 0
longest_word = ""
for word in words:
    if len(word) > max_length:
        max_length = len(word)
        longest_word = word
print(longest_word)
```

Input :

```
This is a sentence with
```

Output :

```
sentence
```

4(b): Write a program to slice a given string based on user-defined start and end positions.

AIM

To write program to slice a given string based on user-defined start and end positions.

ALGORITHM

Step 1: Start..

Step 2: Get the Input .

Step 3: Slice the string.

Step 4: Display the result.

Step 5: Stop.

PROGRAM

```
input_string = input()
start = int(input())
end = int(input())

if 0 <= start <= end < len(input_string):
   sliced_string = input_string[start:end+1]
   print(sliced_string)
else:
   print("Invalid start and end positions")
```

INPUT

```
Pythonprogramming

0

5
```

OUTPUT

```
Python
```

4 (c): Write a Python program that takes a sentence as Input and returns a new sentence with the words reversed

AIM

To write a program that takes a sentence as Input and returns a new sentence with the words reversed

ALGORITHM

Step 1: Start the program

Step 2: Get Input .

Step 3: Reverse the string.

Step 4: Display result.

Step 5: Stop the program .

PROGRAM

```
 sentence = input()
words = sentence.split()
reversed_sentence = ' '.join(word[::-1] for word in words)
print(reversed_sentence)
```

OUTPUT:

```
Racecar is palindrrome

racecaR si emordnilap
```

4(d) Write a Python program that concatenates two strings using string formatting and replaces all instances of a certain character with another character..

AIM:

To write a program that concatenates two strings using string formatting and replaces all instances of a certain character with another character..

ALGORITHM:

Step 1: Start the program
Step 2: Get the inputs
Step 3: Concatenate two strings.
Step 4: Print the output

Step 5: Stop the program

PROGRAM:

```
 string1 = input()
string2 = input()

char_to_replace = input()
replacement_char = input()

concatenated_string = "{}{}".format(string1, string2)

updated_string = concatenated_string.replace(char_to_replace, replacement_char)

print("The updated string is:", updated_string)
```

OUTPUT:

```
Hello
World


Hello world
```

RESULT:

Thus, the above code using string functions has been executed and the output is verified successfully.

EX NO:5 PROGRAM USING LIST AND TUPLES

5(a) Write a program to identify if there are any duplicate elements in a given list..

AIM:

To write a program to identify if there are any duplicate elements in a given list..

ALGORITHM:

```
Step 1: Start the program
Step 2: Get the inputs
Step 3: identify the duplicate element in a list.
Step 4: Print the output

Step 5: Stop the program
```

PROGRAM:

```
def has_duplicate(1st):

          seen = set()

          for num in 1st:

                if num in seen:

                      return True

               seen.add(num)

            return False

     def main():

              1st = list(map(int, input().split()))

              print(has_duplicate(1st))

     if_name main()
```

Input :

```
3 5 7 9 11 2 4 6 8 10
```

Output :

```
False
```

5(b) Write a Python program that takes a space-separated string of elements as input and outputs tuple with its elements reversed.

AIM:

To write a program that takes a space-separated string of elements as input and outputs a tuple with its elements reversed.

ALGORITHM:

Step 1: Start the program
Step 2: Get the inputs
Step 3: Reverse the tuple element.
Step 4: Print the output
Step 5: Stop the program

PROGRAM:

```
def reverse_input(input_str):

        words = input_str.split()

        reversed_tuple = tuple(reversed (words))

        return reversed_tuple

def main():

        input_str = input()

        reversed_tuple = reverse_input(input_str)

         print(reversed_tuple)

if name = "_main_":

     main()
```

Input :

```
Enjoy every moment
```

Output :

```
('moment', 'every', 'Enjoy')
```

5(c) Write a Python program that takes an input string representing a tuple of elements and an integer 'n.' The program should remove the element at the specified position 'n' and output the updated tuple..

AIM:

To write a program to remove the element at the specified position 'n' and output the updated tuple..

ALGORITHM:

```
Step 1: Start the program
Step 2: Get the inputs
Step 3: Remove the element at the specified position 'n' .
Step 4: Print the output

Step 5: Stop the program
```

PROGRAM:

```
 defremove_element_from_tuple(input_str

, index):

    input_tuple = tuple(input_str.split())

    updated_tuple = input_tuple[:index 1]

 input_tuple[index:]

def main():

     input_str = input().strip()

     index = int(input().strip())

     updated_tuple =

 remove_element_from_tuple(input_str,

index)

 print(updated_tuple)
```

OUTPUT

```
qwerty poiu asdf lkj zxcv

mnbh g



('qwerty', 'poiu', 'asdf', 'zxcv', 'mnbh', 'g')
```

5(d) Write a Python program to remove all occurrences of a specified element from a list.

AIM:

To write a program to remove all occurrences of a specified element from a list.

ALGORITHM:

```
Step 1: Start the program ,import math module.
Step 2: Get two inputs
Step 3: remove all specified element.
Step 4: Print the output.

Step 5: End.
```

PROGRAM:

```
def remove_element(input_list, element):

        return [word for word in input_list if word != element]

def main():

      input_list = input().split()

      element = input().strip()

      updated_list = remove_element(input_list, element)

      print(updated_list)

if_name == "_main_":

   main()
```

Input :

```
a b c b d e a f
a
```

Output :

```
['b', 'c', 'b', 'd', 'e', 'f']
```

RESULT:

Thus, the above code using list and tuple has been executed and the output is verified successfully.