ASSIGNMENT

Submitted by,

Ananthalakshmi A G

Roll No.13

1. GCD of two numbers

```
a = int(input("Enter first number:"))
Enter first number:25
b = int(input("Enter second number:"))
Enter second number:75
for i in range(1,min(a,b)+1):
    if(a%i==0) and (b%i==0):
        gcd=i

print("GCD of ",a ," and ",b, " is ",gcd)

Output:
GCD of  25  and  75  is  25
```

2. Factorial

```
a = int(input("Enter number:"))
Enter number:6
f=1
for i in range(1, a+1):
    f=f*i
print("Factorial of ",a," is ",f)

Output:
Factorial of  6  is  720
```

3. Fibonacci Series

```
n = int(input("Enter number:"))
Enter number:8
n1,n2=0,1
n3=n1+n2
print("Fibonacci series of first ",n,":")
Fibonacci series of first  8 :
print(n1)
0
print(n2)
1
for i in range(3, n+1):
    print(n1+n2)
    n1=n2
    n2=n3
    n3=n1+n2

Output:
1
2
3
5
8
13
```

4. Count of vowels

```
s = input("Enter a string:")
Enter a string:Nightmare
```

```
c=0
for i in s:
    if i in 'aeiouAEIOU':
        c=c+1

print("No of vowels:",c)

Output:
No of vowels: 3
```

5. Sum of all items in list

```
li=input("Enter items for list:")
Enter items for list:5 6 9 6 3 2 8
new_list=map(int,li.split())
sum=0
for i in new_list:
    sum=sum+i

print("Sum of items in list:",sum)

Output:
Sum of items in list: 39
```

6. Pyramid

```
a = int(input("Enter no of lines:"))
Enter no of lines:8
for i in range(1,a+1):
    for j in range(1,i+1):
        print(i, end=' ')
    print()

Output:
1
2 2
3 3 3
4 4 4 4
5 5 5 5 5
6 6 6 6 6 6
7 7 7 7 7 7 7
8 8 8 8 8 8 8 8
```

7. Pythagorean Triads

```
>>> for i in range(1,50):
...     for j in range(1,i):
...         for k in range(1,j):
...             if k*k + j*j == i*i:
...                 flag=0
...                 for l in range(2,i):
...                     if i%l==0 and j%l==0 and k%l==0:
...                         flag=1
...                         break
...                 if flag: continue
...                 print("a=",k,"b=",j,"c=",i)
...
```

```
Output:
a= 3 b= 4 c= 5
a= 5 b= 12 c= 13
a= 8 b= 15 c= 17
a= 7 b= 24 c= 25
a= 20 b= 21 c= 29
a= 12 b= 35 c= 37
a= 9 b= 40 c= 41
```

8. Chess – Bishop movements

```
r = int(input("Enter number of rows:"))
Enter number of rows:4
c = int(input("Enter number of columns:"))
Enter number of columns:3
while(c in range(1,9) and r in range(1,9)):
    if(r==1 and c==1):
        print(r,c+1)
        print(r+1,c)
        break
    elif(r==1 and c==8):
        print(r,c-1)
        print(r+1,c)
        break
    elif(c==1 and r==8):
        print(r,c+1)
        print(r-1,c)
        break
    elif(c==8 and r==8):
        print(r,c-1)
        print(r-1,c)
        break
    elif(c==1 and r<8):
        print(r-1,c)
        print(r+1,c)
        print(r,c+1)
        break
    elif(c==8):
        print(r-1,c)
        print(r,c-1)
        print(r+1,c)
        break
    elif(r==1):
        print(r+1,c)
        print(r,c+1)
        print(r,c-1)
        break
    elif(r==8):
        print(r,c-1)
        print(r,c+1)
        print(r-1,c)
        break
    else:
        print(r,c-1)
        print(r,c+1)
        print(r+1,c)
```

```
            print(r-1,c)
            break

    Output:
    4 2
    4 4
    5 3
    3 3
```

9. Count a number in list

```
    l1=input("Enter a list of values:")
    Enter a list of values:4 5 6 7 7 8 4 6
    lis=list(map(int,l1.split()))
    print(lis)
    [4, 5, 6, 7, 7, 8, 4, 6]
    n=int(input("Enter values whose no of occurence is to be
    displayed:"))
    Enter values whose no of occurence is to be displayed:6
    print("No of occurences of ",n," is",lis.count(n))

    Output:
    No of occurences of  6  is 2
```

10. n copies of first 2 characters of string

```
    str = input("Enter a string:")
    Enter a string:Hello
    n = int(input("No of copies:"))
    No of copies:4
    for i in range(n):
        if len(str)>=2:
            print(str[0],str[1])
        else:
            print(str[0])
    Output:
    H e
    H e
    H e
    H e
```

11. Check whether value contained in list of values

```
    l1 = input("Enter list of values:")
    Enter list of values:1 2 3 4 5
    n = int(input("Enter no to be searched:"))
    Enter no to be searched:4
    l1=list(map(int,l1.split()))
    print(l1)
    [1, 2, 3, 4, 5]
    for i in l1:
        if n==i:
            flag = 1
            break
        else:
```

```
            flag =0
    if flag==1:
        print("Number found")
    else:
        print("Number not found")

    Output:
    Number found
```

12. Print even no upto 237

```
    numbers = [10, 15, 20, 30, 237, 40, 50, 60, 70]
    for num in numbers:
        if num == 237:
            break
        if num % 2 == 0:
            print(num)

    10
    20
    30
```

13. Write a python program to get the least common multiple of two positive integers

```
    def gcd(a, b):
        while b:
            a, b = b, a % b
        return a

    def lcm(a, b):
        return (a * b)

    num1 = int(input("Enter the first positive integer: "))
    num2 = int(input("Enter the second positive integer: "))

    if num1 <= 0 or num2 <= 0:
        print("Please enter positive integers.")
    else:

        result = lcm(num1, num2)
        print(f"The LCM of {num1} and {num2} is {result}")

    Output:
    Enter the first positive integer: 2
    Enter the second positive integer: 5
    The LCM of 2 and 5 is 10
```

14. Write a python program to count the number of characters (character frequency) in a string.

```
    def count_characters(string):
        char_count = {}
        for char in string:

            if char.isalnum():
```

```
            char_count[char] = char_count.get(char, 0) + 1

    return char_count
input_string = input("Enter a string: ")

result = count_characters(input_string)

for char, count in result.items():
    print(f"'{char}' occurs {count} times.")

Output:
Enter a string: Good Morning
‘g’ occurs 2 times.
‘o’ occurs 3 times.
‘d’ occurs 1 times.
‘m’ occurs 1 times.
‘r’ occurs 1 times.
‘n’ occurs 2 times.
‘i’ occurs 1 times.
```

15. Write a python program to get a string made of the first 2 and the last 2 chars from a given a string. if the string length is less than 2, return instead the empty string

```
def extract_chars(string):
    if len(string) < 2:
        return ""
    else:
        return string[:2] + string[-2:]
input_string = input("Enter a string: ")

result = extract_chars(input_string)
print(f"The modified string is: {result}")

Output:
Enter a string: hello sunshine
The modified string is: hene
```

16. Write a python program to add 'ing' at the end of a given string(length should be at least 3).if the given string is already ends with 'ing' then add 'ly' instead. if the string length of the given string is less than 3, leave it unchanged

```
def modify_string(string):
    if len(string) < 3:
        return string
    elif string[-3:] == "ing":
        return string + "ly"
    else:
        return string + "ing"
input_string = input("Enter a string: ")
result = modify_string(input_string)
print(f"The modified string is: {result}")
```

```
Output:
Enter a string: Good Morn
The modified string is: Good Morning
```

17. Write a python function that takes a list of words and return the length of the longest one

```
def find_longest_word_length(word_list):
    if not word_list:
        return 0
    longest_word = max(word_list, key=len)
    return len(longest_word)
words = ["apple", "banana", "cherry", "date", "elderberry"]
longest_length = find_longest_word_length(words)
print(f"The length of the longest word is: {longest_length}")

Output:
The length of longest word is: 10
```

18. Write a python program to remove the characters which have odd index values of a given string

```
def remove_odd_index_chars(input_string):
    result = ""
    for i in range(len(input_string)):
        if i % 2 == 0:
            result += input_string[i]
    return result
input_string = input("Enter a string: ")
result_string = remove_odd_index_chars(input_string)
print(f"The string with odd index characters removed is:
{result_string}")

Output:
Enter a string: Good Day
The string with odd index characters removed is: Go Dy
```

19. Write a python program that accepts a comma seperated sequence of words as input and prints the unique words in sorted form(alphanumerically)

```
def unique_sorted_words(input_sequence):
    words_list = input_sequence.split(',')
    words_list = [word.strip().lower() for word in
words_list]
    unique_words = set(words_list)
    sorted_unique_words = sorted(unique_words)
    return sorted_unique_words
input_sequence = input("Enter a comma-separated words: ")
result = unique_sorted_words(input_sequence)
print("Unique words in sorted order:")
for word in result:
    print(word)

Output:
Enter a comma-separated words: apple,orange,grapes,banana
Unique words in sorted order:
```

```
apple
banana
grapes
orange
```

20. Write a python program to count the number of strings where the string length is 2 or more and the first and last character are same from a given list of strings.

```
def count_strings_with_same_first_last(strings_list):
    count = 0
    for string in strings_list:
        if len(string) >= 2 and string[0] == string[-1]:
            count += 1
    return count
input_strings = input("Enter list of strings separated by
commas: ")
strings_list = input_strings.split(',')
result_count =
count_strings_with_same_first_last(strings_list)
print(f"The number of strings with the same first and last
character is: {result_count}")

Output:
Enter list of strings separated by commas: Happy Days
The number of strings with the same first and last character
is: 0
```

21. Write a python program to check a list is empty or not

```
l=input("Enter a list(space seperated):")
lis=list(l.split())
if not lis:
    print("List is empty")
else:
    print("List is not empty")
    print(lis)

Output:
Enter a list(space separated): 8 7 6 4 8
List is not empty
['8', '7', '6', '4', '8']
```

22. Write a python program to find the list of words that are longer than n from a given list of words

```
str=input("Enter a list of words(space seperated) :")
n=int(input("Enter length"))
txt=str.split()
wordlen=[]
for x in txt:
    if len(x)>n:
        wordlen.append(x)
print("Words with length greater than", n,"=",wordlen)

Output:
Enter a list(space separated): one two three four
Enter length: 4
```

```
Words with length greater than 4 = [‘three’]
```

23. Write a python program to generate a 3*4*6 3D array whose each element is *.

```
array=[[['*' for col in range (6)] for col in range (4)] for
row in range(3)]
print(array)

Output:
[[['*', '*', '*', '*', '*', '*'], ['*', '*', '*', '*', '*',
'*'], ['*', '*', '*', '*', '*', '*'], ['*', '*', '*', '*',
'*', '*']], [['*', '*', '*', '*', '*', '*'], ['*', '*', '*',
'*', '*', '*'], ['*', '*', '*', '*', '*', '*'], ['*', '*',
'*', '*', '*', '*']], [['*', '*', '*', '*', '*', '*'], ['*',
'*', '*', '*', '*', '*'], ['*', '*', '*', '*', '*', '*'],
['*', '*', '*', '*', '*', '*']]]
```

24. Write a python program to generate and print a list of first and last 5 elements where the values are squares of numbers between 1 and 30 (both included).

```
l=list()
for i in range(1,15):
    l.append(i**2)
print(l[:4])
print(l[-4:])

Output:
[1, 4, 9, 16]
[121, 144, 169, 196]
```

25. Write a python script to generate and print a dictionary that contains number (between 1 and n) in the form (x*x*X)

```
n=int(input("Enter the limit: "))
d=dict()
for x in range(l+n+1):
    d[x]=x*X
print(d)

Output:
Enter the limit: 8
1 2 3 4 5 6 7 8 9
```

26. Write a python program to convert temperatures to and from celsius ,Fahrenheit

```
def celsius_to_fahrenheit(celsius):
    # Formula to convert Celsius to Fahrenheit
    fahrenheit = (celsius * 9/5) + 32
    return fahrenheit

def fahrenheit_to_celsius(fahrenheit):
    # Formula to convert Fahrenheit to Celsius
    celsius = (fahrenheit - 32) * 5/9
    return celsius

# Menu to choose conversion direction
print("Choose conversion direction:")
```

```
print("1. Celsius to Fahrenheit")
print("2. Fahrenheit to Celsius")

choice = int(input("Enter your choice (1/2): "))

if choice == 1:
    celsius = float(input("Enter temperature in Celsius: "))
    fahrenheit = celsius_to_fahrenheit(celsius)
    print(f"{celsius} Celsius is equal to {fahrenheit}
Fahrenheit")
elif choice == 2:
    fahrenheit = float(input("Enter temperature in
Fahrenheit: "))
    celsius = fahrenheit_to_celsius(fahrenheit)
    print(f"{fahrenheit} Fahrenheit is equal to {celsius}
Celsius")
else:
    print("Invalid choice. Please enter 1 or 2 for
conversion.")

Output:
Choose conversion direction:
1. Celsius to Fahrenheit
2. Fahrenheit to Celsius
Enter your choice (1/2): 1
Enter temperature in Celsius: 25
25.0 Celsius is equal to 77.0 Fahrenheit
```

27. Write a python program that accept a word from the user and reverse it

```
word=input("Enter a word: ")
for char in range(len(word)-1,-1,-1):
    print(word[char],end="")

Output:
Enter a word: Happy
yppaH
```

28. Write a python program that counts odd and even numbers from a list

```
lis=input("Enter some positive integers (space separated):")
numbers=list(map(int,lis.split()))
count_odd=0
count_even=0
for x in numbers:
    if not x%2:
        count_even+=1
    else:
        count_odd+=1
print("Numbers of even numbers: ",count_even)
print("Numbers of odd numbers: ",count_odd)
Output:
Enter some positive integers (space separated): 2 3 6 9 12
Numbers of even numbers:  3
Numbers of odd numbers:  2
```

29. Write a python program which accepts a sequence of comma separated 4 digits binary numbers as its input and print the numbers that are divisible by 5 in a comma separated sequence

```
items=[]
num=input("Enter some binary numbers(comma separated):")
num1=list(num.split(','))
for p in num1:
    x=int(p,2)
    if not x%5:
        items.append(p)
print(','.join(items))

Output:
Enter some binary numbers(comma
separated):101,110,111,1001,1010
101,110,1010
```

30. Write a python program to find numbers between 100 and 400 (both includes) where each digit of a number is an even number . The numbers obtained should be printed in a comma-separated sequence

```
items=[]
for i in range(100,401):
    s=str(i)
    if(int(s[0])%2==0) and (int(s[1])%2==0) and
(int(s[2])%2==0):
        items.append(s)
print(",".join(items))

Output:
200,202,204,206,208,220,222,224,226,228,240,242,244,246,248,2
60,262,264,266,268,280,282,284,286,288,400
```

Functions

1. Write a python function to check whether a number is even or odd

```
def is_even_or_odd(number):
    if number % 2 == 0:
        return "Even"
    else:
        return "Odd"
num = int(input("Enter a number: "))
result = is_even_or_odd(num)
print(f"The number {num} is {result}.")

Output:
Enter a number: 7
The number 7 is Odd.
```

2. Write a python program to calculate the sum of three given numbers,if the values are equal then return thrice of their sum

```
def sum_of_three_numbers(a, b, c):
    if a == b == c:
        return 3 * (a + b + c)
    else:
        return a + b + c
num1 = float(input("Enter the first number: "))
num2 = float(input("Enter the second number: "))
num3 = float(input("Enter the third number: "))
result = sum_of_three_numbers(num1, num2, num3)
print(f"The result is: {result}")

Output:
Enter the first number: 3
Enter the second number: 3
Enter the third number: 3
The result is: 27
```

3. Write a python function to get a new string from a given string where “ls” has been added to the front.If the given string already begins with “ls” then return the string unchanged

```
def new_string(str):
 if len(str)>=2 and str[:2] == "is":
            return str
 return "is" +str
str1=input("Enter a string:")
Enter a string:Happy Day
print("New String ", new_string(str1))
New String isHappy Day
```

4. Write a python program to get a string which is n(non-negative integer) copies of a given string

```
def larger_str(str,n):
       result=""
```

```
            for i in range(n):
                  result=result+ “” + str
            return result

      str1 = input(“Enter a string:”)
      Enter a string: Morning
      N = int(input(“Enter no of copies: ”)
      Enter no of copies:5
      print(larger_str(str1,n))
      Morning Morning Morning Morning Morning
```

5. Write a python function that will return true if the two given integer values are equal or their sum or difference is 5

```
      def check_integer_values(num1, num2):
          if num1 == num2 or num1 + num2 == 5 or abs(num1 - num2)
      == 5:
               return True
          else:
               return False
      num1 = int(input("Enter the first integer: "))
      num2 = int(input("Enter the second integer: "))
      result = check_integer_values(num1, num2)
      if result:
          print("True")
      else:
          print("False")

      Output:
      Enter the first integer: 3
      Enter the second integer: 3
      True
```

6. Write a python program to display Fibonacci series using recursion

```
def fibonacci_recursive(n):
    if n <= 0:
        return []
    elif n == 1:
        return [0]
    elif n == 2:
        return [0, 1]
    else:
        # Recursive call to generate the Fibonacci series
        fib_series = fibonacci_recursive(n - 1)
        fib_series.append(fib_series[-1] + fib_series[-2])
        return fib_series
n = int(input("Enter the number of terms for Fibonacci series:
"))
fib_series = fibonacci_recursive(n)
print("Fibonacci Series (First", n, "terms):", fib_series)

Output:
Enter the number of terms for Fibonacci series: 5
Fibonacci Series (First 5 terms): [0, 1, 1, 2, 3]
```

7. Write a python function to find the sum of digits of a number.

```
def sum(n):
      num_str = str(n)
      digit_sum = 0
      for digit in num_str:
            digit_sum+=int(digit)
      return digit_sum
Output:
n=123456
result=sum(n)
print(result)
21
```

8. Write a python function to concatenate two strings.

```
def concatenate_strings(str1, str2):
    return str1 + str2
string1 = input("Enter the first string: ")
string2 = input("Enter the second string: ")
result = concatenate_strings(string1, string2)
print("Concatenated string:", result)

Output:
Enter the first string: Hello
Enter the second string: World
Concatenated string: HelloWorld
```

9. Write a python function called compare which takes two strings s1 and s2 and an integer n as arguments. The function should return True if first n characters of both the strings are same else the function should return False.

```
def compare(s1,s2,n):
      return s1[:n] == s2[:n]
s1=”exam”
s2=”example”
n=3
result=compare(s1,s2,n)
print(result)
True
```

10. Write a python program to display Fibonacci series using recursion

```
def fibonacci(n):
    if n <= 0:
        return []
    elif n == 1:
        return [0]
    elif n == 2:
        return [0, 1]
    else:
        fib_series = fibonacci(n - 1)
        fib_series.append(fib_series[-1] + fib_series[-2])
        return fib_series
n = int(input("Enter the number of terms in the Fibonacci
series: "))
```

```
fib_series = fibonacci(n)
print("Fibonacci Series:")
print(fib_series)

Output:
Enter the number of terms in the Fibonacci series: 10
Fibonacci Series:
[0, 1, 1, 2, 3, 5, 8, 13, 21, 34]
```