**Taller 2 – Python**

**1.Numeros palíndromos:**

Escriba una función que reciba un numero entero (positivo) o un String y entregue como respuesta si la entrada es o no un palíndromo.

def palindromo(cadena):

  tam=len(cadena)

  cont=0

for i in reversed(range(0, tam)):

  if (cadena[i].lower()!= cadena[cont].lower()):

    cont=cont+1

    print("Intente nuevamente:cadena, No es palindromo")

  else:

      print("El texto ingresado, si es palindromo")

cadena=input("Digite un numero o una palabra: ")

palindromo(cadena)

resultado=palindromo(cadena)

print(resultado)

**2.- Números primos:**

Escriba una función que reciba como entrada un número y determine si es o no un numero primo.

num=int (input("Ingrese un numero: "))

if num>1:

  cont=0

  for i in range(2,num):

      resultado=num%i

      #print("{}%{}={}".format(num,i,resultado))

      if resultado==0:

          cont+=1

  if cont==0:

         print("El {} es un numero primo".format(num))

  else:

    print("El {} no es un numero primo".format(num))

else:

  print("El {} no es un numero primo".format(num))

**3.- Numeros primos 2:**

Escriba una función que devuelva una lista de numeros primos con base a la cantidad que el usuario le solicite, Ejemplo si el usuario ingresa 3 debe retornar los 3 primeros numeros primos.

def visualizar\_primo():

  num=2

  yield num

  while True:

    aux=num

    while True:

      aux+=1

      cont=1

      cont\_div=0

      while cont<=aux:

          if aux%cont==0:

           cont\_div += 1

          if cont\_div >2:

           break

          cont += 1

      if cont\_div ==2:

        yield aux

        num=aux

#Visualizador de numeros primos de forma secuencial

numero=int (input("Ingrese un numero: "))

while numero!=0:

  pr=visualizar\_primo()

  primos=[next(pr) for \_ in range(numero)]

  print(primos)

  break

else:

  print("El numero debe ser diferente a 0")

**4- Numeros primos 3:**

El primo de Mersenne es un numero primo de la forma 2𝑝−1, una de las propiedades de los primos de Mersenne es que p debe ser también un número primo, escriba una función que imprima la cantidad de numeros que el usuario solicite, Ejemplo: si el usuario ingreso 3 los primeros primos de Mersenne deberían ser 3, 7 y 31.

def visualizar\_primo():

  num=2

  yield num

  while True:

    aux=num

    while True:

      aux+=1

      cont=1

      cont\_div=0

      while cont<=aux:

          if aux%cont==0:

           cont\_div += 1

          if cont\_div >2:

           break

          cont += 1

      if cont\_div ==2:

        yield aux

        num=aux

#Mersenee

def mersenee(num):

    primo=True

    for i in range(2,numero):

      if primo==True:

        for i in range(2,numero+1):

            potencia=(math.pow(2,i))

            if (potencia-1) == numero:

                mer=True

            break

numero=int(input('Introduce un número: '))

mersenee(numero)

#Visualizador de numeros primos de forma secuencial

mer=True

while numero!=0:

    mersenee(numero)

    if [mersenee(True)]:

      pr=visualizar\_primo()

      primos=[next(pr) for \_ in range(numero)]

      print(primos)

      break

    else:

     print("El numero debe ser diferente a 0")

**5- Máximo común divisor:**

Escriba una función que reciba dos numeros y retorne el cálculo de su máximo común divisor.

**6.Numeros Romanos**

Escriba una función que reciba un numero y retorne como resultado el numero romano de dicho número.