**Apellido y nombre:**

**Ejercicio 1 (2.5 puntos)**:

Hacer un programa, utilizando funciones, que solicite al usuario un número e indique si es un **número especial**.

Un número especial es un natural que es primo y la suma de sus divisores es divisible por 3.

Por ejemplo, el 5 es primo ya que sus únicos divisores son el 1 y el 5. La suma de sus divisores da 1+5 = 6. 6 es divisible por 3, entonces el 5 es un número especial.

**Ejercicio 2 (2.5 puntos)**:

Nuestra Universidad tiene almacenadas en 2 listas distintas ***que se corresponden,*** *la primera almacena* los números de legajos de los alumnos *y la segunda*  la cantidad de materias aprobadas,

Realizar un programa utilizando la función **dondeesta(valor,lista)** estudiada en la práctica; que ingresado un legajo cualquiera, muestre si el estudiante califica a una beca o no como lo muestran los ejemplos.

**Para calificar a una beca debe tener más de 15 materias aprobadas.**

***Ejemplos:***

LdeLegajo= [ 12701 , 13456 , 17899 , 19821 , 20451 , 21980 ]

LdeMateriasAp= [ 5 , 21 ,17 ,3 , 5 ,17 ]

Si el legajo Ingresado es: 17899

El programa debe informar ***Califica*** a beca pues tiene 17 materias aprobadas

Si el legajo ingresado es: 12550

El programa debe informar: ***Matrícula inexistente***

Si el legajo ingresado es: 20451

El programa debe informar ***NO Califica*** a Beca pues tiene 5 materias aprobadas

**Ejercicio 3 (2.5 puntos)**:

Hacer una función que reciba una lista de números naturales y que devuelva otra lista con los mismos números pero algunos repetidos: si el número es primo y **no** termina en 7 lo debe repetir una vez y si es primo y termina en 7 lo debe repetir una vez más.

Ejemplo:

lista = [3, 9, 2, 7, 8, 12, 17, 43, 57] debe devolver esta otra lista:

lista2 = [3, 3, 9, 2, 2, 7, 7, 7, 8, 12, 17, 17, 17, 43, 43, 57]

**Ejercicio 4 (2.5 puntos)**:

Se desea implementar un controlador de PRODE, un juego donde se pronostican los resultados de cada fecha del fútbol argentino. Cada fecha del campeonato consta de 10 partidos; tomamos como fecha a un entero desde el 1 hasta el 19. En cada fecha los apostadores han pronosticado los resultados de cada partido (local, visitante o empate). Usted debe implementar una función que tome como parámetro una fecha y retorne la lista de ganadores. Se considera ganador a aquel que acierta al menos 7 partidos. Contamos con las siguientes funciones ya implementadas:

**Resultado(fecha)**: recibe un entero y devuelve la lista de resultados de dicha fecha donde 1 significa que ganó el local, 0 empate y -1 que ganó el visitante. La longitud de la lista es igual a la cantidad de partidos jugados (10).

**Nombres(fecha)**: recibe un entero y devuelve la lista con los DNI de los apostadores.

**Apuesta(apostador, fecha)**: para cada apostador devuelve la lista de resultados deportivos elegidos por el apostador para dicha fecha.

Sugerencia: implementarla función

**CantAciertos(listaResultado, listaApostada)**: recorre las listas y devuelve la cantidad de aciertos alcanzado.

**POSIBLES SOLUCIONES:**

**EJ 1**

def esNumeroEspecial(numero):

cont = 0

suma = 0

for i in range(1,numero+1):

if(numero%i == 0):

cont += 1

suma += i

if (cont == 2 and suma % 3 == 0):

return True

else:

return False

#Programa Principal

n = int(input("Ingrese un número"))

if (esNumeroEspecial(n)):

print(n, "es un número especial")

else:

print(n, "no es un número especial")

**EJ 2**

def dondeesta(elemento,lista):

for i in range (len(lista)):

if elemento==lista[i]:

return i

return -1

def beca (materias):

estado="NO CALIFICA "

if materias>15:

estado="CALIFICA "

return estado+"a beca pues tiene "+str(materias)+" materias aprobadas"

LdeMatriculas= [12701 , 13456 , 17899 , 19821 , 20451 , 21980]

LdeMateriasAp= [5 , 21 ,17 ,3 , 5 ,17]

mi=int(input("ingrese matrícula"))

j=dondeesta(mi,LdeMatriculas)

if j==-1:

print ("MATRICULA INEXISTENTE")

else:

print (beca(LdeMateriasAp[j]))

**EJ 3**

def repetirPrimos(lista):

nueva = []

for x in lista:

nueva.append(x)

if esPrimo(x):

nueva.append(x)

if x % 10 == 7:

nueva.append(x)

return nueva

**Eje4**

Def Prode(fecha):

Ganadores=[]

For apostador in Nombres(fecha):

If(CantAciertos(Resultado(fecha),Apuesta(apostador,fecha))>=7):

Ganadores.append(apostador)

Return ganadores