

1. **(3 ptos)** Escriba un procedimiento que muestre ondas de números enteros, es decir, una serie de enteros consecutivos que empiecen en un valor, lleguen a otro (centro de la onda), continúen hasta un tercero y se repita este proceso un número determinado de veces. La onda siempre terminara en el número inicial. Por ejemplo, si número inicial: 5, centro onda: 8, número final: -1 y repeticiones: 2, el resultado será:

5 6 7 8 7 6 5 4 3 2 1 0 -1 0 1 2 3 4 5 6 7 8 7 6 5 4 3 2 1 0 -1 0 1 2 3 4 5

```
def obtenerOndas(ini, centro, fin, nRep):
    """int, int, int, int --> nada
    OBJ: Imprime una onda de enteros
    PRE: nRep>=1 y ini<centro<fin"""
    termino=ini
    for i in range(nRep):
        while termino<=centro:
            print (termino, end=' ')
            termino+=1
        termino=centro-1
        while termino>fin:
            print (termino, end=' ')
            termino-=1
        #Imprimimos lo que reste de onda hasta el término de inicio
        for i in range (termino, ini+1):
            print (i, end=' ')
    return

obtenerOndas(5, 8, -1, 2))
```

2. **(2.5 ptos)** Implementar un subprograma que dado un entero “premiado”, pida al usuario una serie de números enteros (use un centinela para finalizar la entrada) y devolver si en la serie está el número “premiado” y en qué posición salió (-1 si no estuviera en la serie). También debe devolver el número más cercano al “premiado” y su posición. Pruébalo con el siguiente supuesto: si premiado =6 y la serie es 3 8 5 37 14 37 6 22 -999, debe devolver que sí se ha encontrado el premiado, en la posición 7, y que el más cercano es 5, que salió en la posición 3.

```
def averiguar(premiado):
    n=int(input('Número (-999 para terminar): '))
    enc=False
    posi=-1
    i=1
    distMenor=999999
    while n!=-999:
        if n==premiado:
            enc=True
            posi=i
        else:
            dist=abs(premiado-n)
            if dist<distMenor:
                distMenor=dist
                masCercano=n
                posiMasCerca=i
        i+=1
    n=int(input('Número (-999 para terminar): '))
```

```

        return (enc, posi, masCercano, posiMasCerca)

    encontrado, posicion, masCercano, posiMasCercano= averiguar(6)
    if encontrado:
        print('Encontrado en posición: ', posicion)
    else:
        print('No ha salido el premiado :(')
    print('El más cercano es: ', masCercano)
    print('y salió en la posición: ', posiMasCercano)

```

3. **(2 ptos)** Programa que muestra en pantalla todos los términos de la siguiente serie numérica entre 1 y un término n dado: 1/2, 3/3, 6/12, 18/78, etc. donde como se ve cada numerador es la suma del numerado y denominador del término anterior y cada denominador es la suma del numerador anterior más la multiplicación de numerador anterior por el denominador anterior.

```

def termino(n):
    """ int --> none
        OBJ: Imprime los n primeros términos de la serie 1/2, 3/3, 6/12,
        18/78, ..."""
    num_anterior = 1
    den_anterior = 2
    print(num_anterior, '/', den_anterior)
    i = 2
    while i<=n:
        numerador = num_anterior + den_anterior
        denominador = num_anterior+ num_anterior*den_anterior
        print(numerador, '/', denominador)
        num_anterior = numerador
        den_anterior = denominador
        i += 1

termino(5)

```

4. **(2.5 pto)** Realice un programa que muestre todos los números en la línea de distancia mínima de un número a sus límites naturales. El usuario introduce el número (por ejemplo 35) y los límites (20 y 40 por ejemplo). Se debe primero calcular si el número está entre dichos límites y si es así, la línea de distancia mínima es 36-37-38-39-40, ya que 35 se encuentra a 15 unidades del límite inferior (20) y a 5 del superior (40) y por tanto la distancia mínima a sus límites es la que va desde 35 a 40.

```

def dentroLimites(x, l, h):
    """int, int, int --> bool
        OBJ: Devuelve verdadero si l<=x<=h, falso en caso contrario
        PRE:l<h"""
    return (l<=x<=h)

def mostrarLineaDistanciaMin(x, l, h):
    """int, int, int --> nada
        OBJ: Muestra la línea de distancia mínima de un número a sus límites
        PRE: l<=x<=h"""
    distHastaL=x-l
    distHastaH= h-x
    if distHastaL==0 or distHastaH==0:
        print ('La distancia mínima es 0')

```

```

elif distHastaL<=distHastaH:
    for i in range(l, x+1):
        print (i, end=' ')
else:
    for i in range(x, h+1):
        print (i, end=' ')

x=35
l=20
h=40
if l<h and dentroLimites(x, l, h):
    mostrarLineaDistanciaMin(x, l, h)

```

### **EXTRA:**

1. Programar en Python la función discontinua  $f(x)$  definida como:

$$f(x) = x^2 - 3x, \text{ si } x < 1$$

$$f(x) = 10, \text{ si } x = 1$$

$$f(x) = 4x + 2, \text{ si } x > 1$$

```

import math
def funcion (x):
    if x<1:
        res=math.pow(x,2)-x*3
    elif x==1:
        res=10
    else:
        res=4*x+2
    return res

print(funcion(2))

```

2. Implementar en Python una calculadora que, dependiendo de la opción elegida, realice las siguientes tareas siempre y cuando no se haya elegido la opción 0:

Opción 0: no se realiza ninguna operación y finaliza el programa

Opción 1: realiza la suma entre dos números

Opción 2: realiza la resta entre dos números

Opción 3: realiza la multiplicación de dos números

Opción 4: realiza la división entre dos números

El programa debe ser robusto y estar protegido frente a fallos. El usuario puede estar realizando todas las operaciones que desee (opciones de 1 a 4) hasta que decide terminar pulsando la opción 0.

```

def calculadora(opcion,num1,num2):
    """ int --> float
    OBJ: según la opción elegida, se realiza una operación u otra
    """

    if opcion == 1:
        num = num1 + num2
    elif opcion == 2:

```

```

        num = num1 - num2
    elif opcion == 3:
        num = num1 * num2
    elif opcion == 4:
        if num2 != 0:
            num = num1 / num2

    return num

opcion=int(input("""Elija una opción de la calculadora:\n
1 -> suma\n
2 -> resta\n
3 -> multiplicación\n
4 -> división"""))
while opcion!=0:
    num1=float(input("Introduce el primer número!))
    num2=float(input("Introduce el segundo número!))
    print("El resultado de la operación realizadas es: ",
caculadora(opcion, num1, num2))
    opcion=int(input("""Elija una opción de la calculadora:\n
1 -> suma\n
2 -> resta\n
3 -> multiplicación\n
4 -> división"""))

```

3. (1 pto) Un video club necesita un programa informático para gestionar sus facturas. Por el alquiler de cualquier película cobra una fianza de 5€ que se añade al precio de alquiler de la misma. El precio de alquiler varía según el número de días y el género de la película:
- a. Alquiler inferior o igual a una semana:
    - i. Drama: 3€/día
    - ii. Comedia: 2€/día
    - iii. Terror: 4€/día
    - iv. Animación: 3€/día
    - v. Ciencia ficción: 2€/día
    - vi. Intriga y acción: 3€/día
  - b. Alquiler superior a una semana:
    - i. Drama: 2€ por cada semana completa y 1.5€ por cada día.
    - ii. Comedia: 1.5€ por cada semana completa y 0.5€ por cada día.
    - iii. Terror: 3€ por cada semana completa y 2€ por cada día.
    - iv. Animación: 2€ por cada semana completa y 2€ por cada día.
    - v. Ciencia ficción: 1.5€ por cada semana completa y 1€ por cada día.
    - vi. Intriga y acción: 2€ por cada semana completa y 1€ por cada día.

Implemente un programa modularizado que calcule el precio del alquiler de una película en función de su tipo y el número de días que se alquiló.

```

def gestiona_facturas(ndias,genero):
    if ndias<=7:
        if genero.upper() == 'D' or genero.upper() == 'A' or genero.upper() == 'I':
            precio=3*ndias
        elif genero.upper() == 'R' or genero.upper() == 'C':
            precio=2*ndias
        else:
            precio=4*ndias
    else:

```

```

    nsemanas=int(ndias/7)
    print(nsemanas)
    if genero.upper() == 'D':
        precio=2*nsemanas+1.5*(ndias-nsemanas*7)
    elif genero.upper() == 'R':
        precio=1.5*nsemanas+0.5*(ndias-nsemanas*7)
    elif genero.upper() == 'M':
        precio=3*nsemanas+2*(ndias-nsemanas*7)
    elif genero.upper() == 'A':
        precio=2*nsemanas+2*(ndias-nsemanas*7)
    elif genero.upper() == 'C':
        precio=1.5*nsemanas+1*(ndias-nsemanas*7)
    else:
        precio=2*nsemanas+1*(ndias-nsemanas*7)

    return 5+precio

ndias = int(input("¿Cuántos días desea alquilar la película? "))
if ndias>0:
    genero = input("""Indique el género de la película: \n
                    Drama: D\n
                    Comedia: R\n
                    Terror: M\n
                    Animación: A\n
                    Ciencia Ficción: C\n
                    Intriga y acción: I\n""")
    if genero.upper() == 'D' or genero.upper() == 'R' or genero.upper() == 'M'
    or genero.upper() == 'A' or genero.upper() == 'C' or genero.upper() == 'I':
        print("El precio de alquiler de la película es de: ",
              gestiona_facturas(ndias,genero))
    else:
        print("Género de película incorrecto")
else:
    print("Número de días incorrecto")

```