

# Prueba 1

## Programación

II-2024

---

Tiempo Total: 2 Horas

---

No de vuelta esta página de la prueba hasta que se lo indiquen.

Instrucciones generales:

- La prueba dura 120 minutos desde el momento en que el profesor/a indica que se puede comenzar.
- La prueba es INDIVIDUAL, sin apuntes ni calculadora.
- Es importante aclarar que esta prueba será escaneada, por lo cual **no escriba nada** en la misma área horizontal que los códigos QR y utilice un lápiz cargado. La utilización de lápiz grafito no es recomendada y queda bajo su propia responsabilidad.
- Se puede escribir por ambas caras de cada hoja, recuerde no escribir nada en la misma área horizontal que los códigos QR.
- Ud. es el responsable de que lo que hizo en el código se entienda, por lo tanto, es altamente recomendable que explique sus acciones comentando el código de tal forma que el corrector entienda de primera fuente lo que Ud. intentó realizar.
- Esta evaluación se rige bajo las normas de honestidad y probidad mencionadas en el Código de Honor que el estudiante acepta al matricularse en esta universidad. El no cumplir con esta disposición puede constituir hasta la expulsión del alumno de la universidad.
- El uso de dispositivos electrónicos durante la prueba está prohibido.

## Pregunta 1: (1 punto)

Cree un programa que solicite el número de boletas a ingresar por el usuario. Posteriormente, deberá solicitar el monto de cada boleta. Una vez que se ingresa el número de boletas mencionadas, se deberá mostrar por pantalla la suma total de las boletas, el monto de IVA pagado por los usuarios (el 19% de la suma de las boletas) y el dinero que ingresó al comercio sin contar el IVA (el 81% del valor de la boleta).

Ejemplo de ejecución:

¿Cuántas boletas ingresará en el sistema?: 5

Ingrese valor boleta: 10000

Ingrese valor boleta: 20000

Ingrese valor boleta: 30000

Ingrese valor boleta: 20000

Ingrese valor boleta: 20000

La suma total de las boletas es: 100000

El IVA ingresado es: 19000.0

El monto ingresado al negocio es: 81000.0

Solución sin listas:

```
print("¿Cuántas boletas ingresará en el sistema?:")
numBoletas=int(input())
suma=0
for i in range(numBoletas):
    print("Ingrese valor boleta:")
    valorBoleta=int(input())
    suma=suma+valorBoleta
print("La suma total de las boletas es:",suma)
print("El IVA ingresado es:",suma*0.19)
print("El monto ingresado al negocio es:",suma*0.81)
```

Solución con listas:

```
boletas=[]
print("¿Cuántas boletas ingresará en el sistema?:")
numBoletas=int(input())
for i in range(numBoletas):
    print("Ingrese valor boleta:")
    boletas.append(int(input()))
print("La suma total de las boletas es:",sum(boletas))
print("El IVA ingresado es:",sum(boletas)*0.19)
print("El monto ingresado al negocio es:",sum(boletas)*0.81)
```

Solución chat gpt

# Solicitar la cantidad de boletas a ingresar

```
num_boletas = int(input("¿Cuántas boletas ingresará en el sistema?: "))
```

```
# Inicializar variable para la suma total
suma_total = 0

# Solicitar el valor de cada boleta e ir sumando los valores
for i in range(num_boletas):
    monto = float(input("Ingrese valor boleta: "))
    suma_total += monto

# Calcular IVA (19%) y el monto ingresado al negocio (81%)
iva = suma_total * 0.19
monto_negocio = suma_total * 0.81

# Mostrar los resultados
print(f"La suma total de las boletas es: {suma_total}")
print(f"El IVA ingresado es: {iva}")
print(f"El monto ingresado al negocio es: {monto_negocio}")
```

## Pregunta 2: Depuración de Código (2 puntos)

El siguiente programa debería determinar si el número ingresado por un usuario es primo. Para ello, la idea implementada trata de revisar si el número es divisible por algún número entre 2 y numPrimo-1 y tomar una decisión en base a ello. Sin embargo, el código tiene problemas y no realiza lo solicitado:

```
numPrimo=int(input("Ingrese un número para revisar si es primo: "))
verificarPrimo = False
i=1
while (i<numPrimo):
    verificarPrimo = numPrimo%i==0
    i=i+1
if (verificarPrimo == True):
    print("NO es primo", end=""):
else:
    print("Es primo")
```

1) ¿Qué mostraría por pantalla el algoritmo si se ingresa el valor 1? Justifique su respuesta

Es primo, en este caso no se entra al ciclo, por lo cual el valor es False

2) ¿Qué mostraría por pantalla el algoritmo si se ingresa el valor 4? Justifique su respuesta

Es primo, en este caso se entra al ciclo, pero en el último intento  $i=3$  y  $\text{numPrimo}\%i==0$  será False.

3) Reescriba el código para determinar si un número es primo

```
numPrimo=int(input("Ingrese un número para revisar si es primo: "))
verificarPrimo=True
i=2
while (i<numPrimo):
    if (numPrimo%i==0):
        verificarPrimo = False
        i=i+1
if (verificarPrimo == True):
    print("Es primo")
else:
    print("No es primo")
```

### Pregunta 3: (3 puntos)

Uno de los algoritmos más básicos de encriptación y descifrado es simplemente utilizar números en vez de las letras del alfabeto. Por ejemplo, la palabra "hola" corresponde a la secuencia de números 7,14,11,0, ya que la primera letra del alfabeto (valor 0) es la letra a, la octava letra (valor 7) es la letra h, y así sucesivamente.

Teniendo en cuenta esta idea de encriptación, considere que a usted le han dado dos listas de datos:

1. "alfabeto" con todas las letras minúsculas del abecedario (27 caracteres desde 'a' [índice 0] hasta la letra z [índice 25], pero sin incluir la letra ñ); y
2. "msje" que contiene números con el mensaje encriptado.

Usted debe crear un programa para mostrar en pantalla el mensaje encriptado que contiene "msje", usando la información contenida en "alfabeto". Por lo tanto, el programa deberá leer cada índice de la lista "msje" y mostrar por pantalla la letra correspondiente según la lista "alfabeto". En caso que el número a mostrar esté fuera del rango de la lista alfabeto, el programa deberá terminar su ejecución y mostrar el mensaje generado hasta ese momento.

Ejemplo de ejecución:

#Si las listas fueran

alfabeto= ['a', 'b', 'c', ..., 'x', 'y', 'z', '']

msje = [21, 0, 12, 14, 18, 26, 15, 14, 17, 50, 4, 11, 26, 18, 8, 4, 19, 4]

#El programa debería mostrar por pantalla: "Vamos por"

#Correspondiente a los números 21, 0, 12, 14, 18, 26, 15, 14, 17, pero finaliza por el valor 50

#HINT1: Asuma que alfabeto y msje ya están definidos

#HINT2: recuerde que print(...,end='') permite mostrar algo por pantalla sin hacer salto de línea.

```

import string

# Lista de caracteres minúsculos
alfabeto = list(string.ascii_lowercase)
alfabeto.append(" ")
msje = [21, 0, 12, 14, 18, 26, 15, 14, 17, 50, 4, 11, 26, 18, 8, 4, 19, 4]
flag=True
index=0
msjeFinal=""
while(flag):
    if (msje[index]<0 or msje[index]>len(alfabeto)):
        flag=False
    else:
        msjeFinal=msjeFinal+alfabeto[msje[index]]
        index=index+1
        if index>=len(msje):
            flag=False
print(msjeFinal)

```

Definición	Ejemplo	Qué muestra el Ejemplo
<b>Asignar un valor a una variable</b>		
<code>variable = valor</code>	<code>x='0'</code>	
<b>Mostrar un texto por pantalla</b>		
<code>print(texto)</code>	<code>x='0'</code> <code>print('x es', x)</code>	x es 0
<b>Capturar un dato ingresado por teclado por el usuario</b>		
<code>nombre = input("mensaje")</code>	<code>x=input('ingrese un dato: ')</code> <code>print('x es', x)</code>	ingrese un dato: 12 x es 12
<b>Convertir entre tipos de datos</b>		
<code>texto = '123'</code> <code>numero = int(texto)</code> <code>decimal = float(texto)</code> <code>texto = str(numero)</code>	<code>texto = '123'</code> <code>numero = int(texto)</code> <code>print(numero)</code>	123
<b>Operaciones</b>		
<b>comparación:</b> <, <=, ==, !=, >=, > <b>matemáticas:</b> /, *, +, - <b>resto división:</b> % <b>potencia:</b> ** <b>división entera:</b> //	<code>print(12 &lt; 23)</code> <code>print(2 + 5)</code> <code>print(5 % 2)</code> <code>print(2 ** 3)</code> <code>print(7 // 2)</code>	True 7 1 8 3
<b>Condicional:</b> Se verifica la condición. Si es verdadera/falsa se ejecutan las instrucciones_if/instrucciones_elif/instrucciones_else		
<code>if (condición):</code> <instrucciones_if> <code>elif (condición):</code> <instrucciones_elif> <code>else:</code> <instrucciones_else>	<code>x=input('ingrese saludo: ')</code> <code>if x == 'hola':</code> <code>print('buen día')</code> <code>elif x == 'chao':</code> <code>print('hasta luego')</code> <code>else:</code> <code>print(x)</code>	ingrese saludo: chao hasta luego
<b>Ciclos condicionales:</b> Se ejecutan las instrucciones del ciclo mientras la condición sea verdadera		
<code>while (condición):</code> <instrucciones>	<code>x=0</code> <code>while x &lt; 3:</code>	1 2

	<pre>print(x+1) x = x + 1</pre>	3
<b>Ciclo for: Repetir instrucciones un número de veces</b>		
<pre>for var in range(val1,val2,val3):     &lt;instrucciones&gt;</pre>	<pre>for x in range(0,6,2):     print(x)</pre>	0 2 4
<b>Definir una lista de valores</b>		
<pre>lista = [&lt;val0&gt;,...,&lt;val_n-1&gt;]</pre>	<pre>lista = [1,3,7,13] print(lista)</pre>	[1,3,7,13]
<b>Acceder a un elemento de la lista</b>		
<pre>lista[i]</pre>	<pre>lista = [1,3,7,13] print(lista[2])</pre>	7
<b>Agregar nuevo elementos a una lista</b>		
<pre>lista.append(&lt;valor&gt;) lista.insert(&lt;posicion&gt;,&lt;valor&gt;)</pre>	<pre>lista = [1,3,7,13] lista.append(21) lista.insert(2,12) print(lista) print(lista[2:len(lista)-1])</pre>	[1,13,12,7,13,21] [12,7,13]
<b>Funciones para listas</b>		
<pre>max(lista) min(lista) sum(lista)</pre>	Retorna el máximo número de la lista. Retorna el mínimo número de la lista. Retorna la suma de los número de la lista.	