

TAREA 1

Nombre: Karen Gabriela Rodríguez Corona

Instrucciones:

Desarrolla en Python un código que permita dar solución a las siguientes ecuaciones. Es necesario mostrar el procedimiento; algoritmo, código y resultado final con unidades correspondientes.

1. Ley de los Gases Ideales

La ecuación de los gases ideales relaciona la presión, el volumen, la cantidad de

$$PV = nRT$$

sustancia y la temperatura mediante la expresión:

Datos:

P= 1.2 atm

R= 0.0821 L * atm/ mol* K (*busqueda hecha*)

T= 273 K

V= 0.7 L

Realice una búsqueda para obtener el valor de la constante de los gases (R) en las unidades requeridas. Determina el valor de los moles n del sistema.

Solución:

Lo primero que debemos hacer es despejar la ecuación (gases ideales) de la siguiente manera para determinar los moles del sistema dado:

$$PV = nRT$$

Pasamos dividiendo RT del otro lado:

$$PV/ RT = n$$

*Por lo tanto queda de la siguiente manera: *

$$n=PV/RT$$

Ahora solo sustituimos cada termino:

```
'Datos'
P = 1.2 #atm
R = 0.0821 #L*atm/mol* K # Este dato fue investigado para las unidades de los c
T = 273 #K
V = 0.7 #L

n = (P*V)/(R*T)
print(n) #En este caso pedimos que se nos imprima el resultado, pero no aparece

0.03747774758736999
```

Haremos un paso más, en este caso le pediremos a Phyton que nos de los decimales necesarios, de esta forma.

```
'Datos'
P = 1.2
R = 0.0821
T = 273
V = 0.7
#mismo código, pero en esta parte agregaremos el valor solicitado con unidades.

n = (P*V)/(R*T)
print(f'Número de moles: {n: .2} mol') #utilizamos f (f-string), que sirve para

Número de moles: 0.037 mol
```

2.Ecuación de Arrhenius (Cinética Química)

Esta ecuación describe la dependencia de la constante de velocidad k con la temperatura:

$$k = A e^{\frac{-E_a}{RT}}$$

Calcula la constante de velocidad k.

Solución:

Para poder resolver este ejercicio es importante primeramente enfocarnos en los datos dados, en este caso podemos observar que tenemos un número con notación científica.

```
'Datos'
A = 5*10**7 #s^-1
Ea = 75000 #J* mol^-1
R = 8.31 #J*mol^-1 * K^-1
T = 350 #K

k = A *np.exp(-Ea / (R * T)) #NumPy nos ayuda a calcular funciones matemáticas.
print(k)
```

0.00032020535930733963

```
'Datos'
A = 5*10**7
Ea = 75000
R = 8.314
T = 350

k = A *np.exp(-Ea / (R * T))
print(f'Constante de velocidad:{k:.1}, 1/s') #En este caso pedimos que se nos i
```

Constante de velocidad:0.0003, 1/s

3. Ecuación cuadrática

Dada una ecuación cuadrática

$$ax^2 + bx + c = 0,$$

El discriminante ($\Delta = b - 4ac$) determina el tipo de raíces:

- $\Delta > 0 \rightarrow$ Dos raíces reales distintas
- $\Delta = 0 \rightarrow$ Una raíz real doble
- $\Delta < 0 \rightarrow$ Dos raíces complejas conjugadas

Realice el código que indique el tipo de raíces e emprima el valor si son reales.

Solución:

```
# Importar funciones matemáticas
import math
```

```
#Coeficientes de la raíz cuadrática
a=22
```

```
b=-45
```

```
c=46
```

```
delta=(b**2)-(4*a*c)
# Si delta es positivo, hay dos raíces reales distintas
if delta> 0:
    x1= (-b + math.sqrt(delta))/(2*a)
    x2= (-b-math.sqrt(delta))/(2*a)
    print('Dos raíces reales distintas:', x1, x2)
# Si delta es cero, hay una raíz real doble
elif delta == 0:
    X= -b/(2*a)
    print('Una raíz real doble:',x)
# Si delta es negativo, las raíces no son reales: son complejas conjugadas
else:
    print('Raíces complejas conjugadas')
```

Raíces complejas conjugadas

4. Notas finales de clases

Escribe el código usando estructuras de control if, elif y else para determinar la nota de clase de acuerdo con la calificación numérica final obtenida.

Menos de 6, es NA

De 6 a menos de 7.6, es S

De 7.6 a menos de 8.6, es B

De 8.6 a 10, es MB

Solución:

```
# Solicitamos al usuario una nota calificación y la convertimos a decimal (float)
nota = float(input("Ingresa la nota (0-10):"))
if nota>= 8.6 and nota <=10:
    #Si está entre 8.6 y 10
    print ('MB')
elif nota>= 7.6 and nota <= 8.5:
    # Entre 7.6 y menor que 8.6
    print('B')
elif nota>=6 and nota <=7.5:
    # Intervalo entre 6.0 y 7.6
    print('S')
else:
    # Si es menor que 6 (reprobado)
    print('NA')
```

Ingresá la nota (0-10):5
NA

5. Número de Reynolds (Flujo de fluidos)

El número de Reynolds (Re) se utiliza para caracterizar el régimen de flujo (laminar, de transición o turbulento):

$$Re = \frac{\rho v D}{\mu}$$

Datos:

$$\rho = 1000 \text{ kg}\cdot\text{m}^{-3}$$

$$v = 1,2 \text{ m}\cdot\text{s}^{-1}$$

$$D = 0,05 \text{ m}$$

$$\mu = 0,001 \text{ Pa}\cdot\text{s}$$

Calculá el valor del número de Reynolds.

Solución:

```
rho = 1000
v = 1.2
D = 0.05
mu = 0.001
```

```
Re = rho*v*D/mu
print(f'El número de Reynolds es: {Re:.0f}') #En este caso en particular, el númer
```

```
El número de Reynolds es: 60000
```

6. Uso de condicionales: Determinación del régimen de flujo

Aplicar estructuras de control `if`, `elif` y `else` para determinar el tipo de flujo a partir del número de Reynolds calculado.

El programa deberá solicitar al usuario los valores necesarios mediante la función `input()`, correspondientes a:

- Densidad del fluido, ρ [kg/m³]
- Velocidad promedio del fluido, v [m/s]
- Diámetro interno de la tubería, D [m]
- Viscosidad dinámica, μ [Pa·s]

Con los valores introducidos, el código deberá calcular el número de Reynolds:

$$Re = \frac{\rho v D}{\mu}$$

Posteriormente, el programa determinará el régimen de flujo de acuerdo con los siguientes criterios:

$Re < 2300 \rightarrow$ Flujo laminar

$2300 \leq Re \leq 4000 \rightarrow$ Flujo de transición

$Re > 4000 \rightarrow$ Flujo turbulento

Solución

```

rho = float(input('Ingresa la densidad del fluido en kg/m3: ')) #float es utili
v = float(input('Ingresa la velocidad promedio del fluido en m/s: ')) #input se
D = float(input('Ingresa el diámetro interno de la tubería en m: '))
mu = float(input('Ingresa la viscosidad dinámica del fluido en Pa/s: '))

Re = rho*v*D/mu

if Re < 2300:
    print('El flujo se mantiene en régimen laminar')
elif Re >= 2300 and Re <= 400:
    print('El sistema opera en flujo transitorio')
else:
    print('El flujo del sistema es turbulento')

```

```

Ingresa la densidad del fluido en kg/m3: 0.999
Ingresa la velocidad promedio del fluido en m/s: 23
Ingresa el diámetro interno de la tubería en m: 0.76
Ingresa la viscosidad dinámica del fluido en Pa/s: 1.93

```

El flujo se mantiene en régimen laminar

7. Estado del agua respecto al punto crítico

Aplicar estructuras condicionales para determinar el estado del agua en función de su temperatura y presión, comparándolas con las condiciones críticas.

El *punto crítico* del agua se define por una temperatura y una presión a las cuales la fase líquida y la fase vapor se vuelven indistinguibles. Los valores son:

$$T_c = 647,1 \text{ K}, \quad P_c = 22,06 \text{ MPa}$$

El programa deberá solicitar al usuario la temperatura y presión del sistema mediante la función `input()`, y posteriormente determinar el estado del agua utilizando estructuras condicionales `if`, `elif` y `else` de acuerdo con los siguientes criterios:

- Si $T < T_c$ y $P < P_c$: el agua se encuentra en **fase líquida o vapor**.

Si $T > T_c$ y $P > P_c$: el agua está en estado supercrítico.

Si $T < T_c$ y $P > P_c$: el agua se encuentra en condiciones de líquido comprimido.

En cualquier otro caso: las condiciones son cercanas al punto crítico.

Solución:

```

Tc = 647 #[=]K
Pc = 22.06 #[=] Mpa

T = float(input('Ingrese la temperatura del sistema: '))
P = float(input('Ingrese la presion del sistema: '))

if T < Tc and P < Pc:
    print('El agua se encuentra en estado líquido o vapor.')
elif T > Tc and P > Pc:
    print('El agua se presenta en estado supercrítico.')
elif T < Tc and P > Pc:
    print('El agua se presenta en condiciones de líquido comprimido.') #Podemos c
else:
    print('Bajo las condiciones de operación, estas son cercanas al punto crítico')

```

```

Ingrese la temperatura del sistema: 456
Ingrese la presion del sistema: 13.45
El agua se encuentra en estado líquido o vapor.

```

