

## ▼ Unidad 01: Radiación de Cuerpo Negro

- **Nombre:** Yael Flores Ambrosio
- **email:** [veloesyael@gmail.com](mailto:veloesyael@gmail.com)
- **Matricula:** 201860558



## ▼ Ley de Planck

Planck generó una expresión teórica para la distribución de la longitud de onda

$$I(\lambda, T) = \frac{2\pi hc^2}{\lambda^5 (e^{hc/\lambda k_B T} - 1)}$$

Donde:

- $k_B$  = Constante de Boltzman
- $h$  = Constante de Planck
- $c = [m^2/s]$  Velocidad de la luz
- $T = [K]$  Temperatura
- $\lambda = [m]$  Longitud de onda

```
# Importar librerias (se usaran más adelante en el curso)
```

```
import numpy as np #Esta libreria hace calculo de matrices
import matplotlib.pyplot as plt # Esta liberaria nos permite graficar
```

```
# Declarar constantes
```

```
h          = 6.63e-34 # [J s] Constante de Planck
kB          = 1.38e-23 # [J/K] Constante de Boltzman
vel_luz     = 2.99e8   # [m/s] Velocidad de la luz en el vacío
T           = 6000     # [K]   Temperatura de la superficie
```

```
# Variable independite (Temperatura): P(T)
#URL: https://numpy.org/doc/stable/reference/generated/numpy.arange.html
```

```
x = np.arange(0.01,3.01,0.01) # [μm] np.arange(inicio,final,espacio)
Longitud_Onda = x*1e-6        # [m] Longitud de onda en metros
```

```

#x
#Longitud_Onda

#URL https://numpy.org/doc/stable/reference/generated/numpy.array.html

type(x) #Esta función muestra el tipo de objeto (Arreglo de Numpy)

        numpy.ndarray

# Programar la ecuación (Variable dependiente)

Intensidad_6000 = (2*np.pi*vel_luz**2*h)/((Longitud_Onda**5)*(np.exp((h*vel_luz)/(Longitud_Onda*
Intensidad_5000 = (2*np.pi*vel_luz**2*h)/((Longitud_Onda**5)*(np.exp((h*vel_luz)/(Longitud_Onda*
Intensidad_4000 = (2*np.pi*vel_luz**2*h)/((Longitud_Onda**5)*(np.exp((h*vel_luz)/(Longitud_Onda*
Intensidad_3000 = (2*np.pi*vel_luz**2*h)/((Longitud_Onda**5)*(np.exp((h*vel_luz)/(Longitud_Onda*
Intensidad_2000 = (2*np.pi*vel_luz**2*h)/((Longitud_Onda**5)*(np.exp((h*vel_luz)/(Longitud_Onda*

# Revisa las unidades de Intensidad de radiación de cuerpo negro

<ipython-input-5-fcaaa649f424>:7: RuntimeWarning: overflow encountered in exp
  Intensidad_2000 = (2*np.pi*vel_luz**2*h)/((Longitud_Onda**5)*(np.exp((h*vel_luz)/(

# Graficar Intensidad(Longitud_Onda , T)

plt.figure(figsize=(8,10))
plt.title('Ley de Planck')

# Graficar curvas
plt.plot(Longitud_Onda, Intensidad_6000, label = '6000 K')
plt.plot(Longitud_Onda, Intensidad_5000, label = '5000 K')
plt.plot(Longitud_Onda, Intensidad_4000, label = '4000 K')
plt.plot(Longitud_Onda, Intensidad_3000, label = '3000 K')
plt.plot(Longitud_Onda, Intensidad_2000, label = '2000 K')

# Muestra las legendas
plt.legend()

# Nombre de los ejes
plt.xlabel('$\lambda$ [m]', fontsize=14)
plt.ylabel('Intensidad $ (W/m^2)/m^{-1}$', fontsize=14)

# Define los limites de las graficas
plt.ylim(0,1.1e14)
plt.xlim(0,0.3e-5)

plt.show()

```



