

Silvio Orozco Vizquerra  
Carné 18282  
Fecha 09 de agosto, 2018

#### Ejercicio 4

##### Clases:

##### Clase DriverAtomos:

Su objetivo es ser la clase principal o driver program para ejecutar las funcionalidades de las demás clases. En ella se pide la información al usuario y se muestra simplemente la información necesaria de retorno para el usuario.

##### Clase DecimalFormat:

Su objetivo es ser la clase que provee métodos de conversión de strings a números, separar números y dimensionales, al igual que proveer formatos para mostrar datos numéricos.

##### Métodos:

Decimalformatsymbols(): Para cambiar el símbolo de E a  $10^{\wedge}$ .

setExponentSeparator(): Lo mismo que lo anterior.

Decimalformat(patrón,símbolos): Con el patrón que deseamos y los símbolos genera un decimal format como deseamos.

##### Clase Scanner:

Su objetivo es ser la clase para instanciar un scanner y que se puedan leer las entradas del usuario.

Utilizaremos el método next(). Que lee el siguiente string.

##### Clase Math:

Su objetivo es ser una clase que provea métodos necesarios para ejecutar valores de operaciones, tales como la ecuación de decaimiento radiactivo. Al igual que proveerá random para ejecutar las probabilidades de que cierta cantidad de átomos decaigan por hora.

##### Métodos:

Math.abs() Los parámetros son un valor para sacar valor absoluto.

Math.pow(): Los parámetros son la base y un exponente para la ecuación.

Math.floor()Calcular el entero menor de un número con parámetro el número..

Math.ceil()Calcular el entero mayor de un número con parámetro el número..

Math.random(). Para generar un número aleatorio con la fórmula multiplicada por el número mayor+1.

### Clase IsotopoSimulador:

Su objetivo es ser la clase de cada tipo de isotopo con información necesaria para realizar todos los cálculos, desde la simulación y de igual forma en la ecuación, comparando si la simulación fue exitosa o no. En ella suceden, la mayor parte de los métodos a los que se llama en el main y hace uso de la Clases Math y DecimalFormat para ejecutar sus métodos.

### Propiedades:

- Private float VidaMedia; Esto es una constante con la vida media del elemento.
- Private float MasaMolar; Esto es una constante con la masa molar del elemento.
- Private porcentajePorhora; Esto es una constante con el porcentaje que decae por hora.
- Private float Cantatomos; Esto es una variable que será importante para reportar la cantidad de átomos pedidos por la persona para conocer el tiempo de su decaimiento radiactivo. Sera pedida en el main.
- Private float cantidadendim; Es la cantidad en la dimensional dada.
- Private string Dimensional. Es la dimensional en la que fueron los datos brindados.
- Private float HorasSimulador; Esto será una variable asignada a partir de que se ejecute el simulador.
- Private float ValorEcuacion. Esto será el valor de una variable asignada a partir de ejecutar la ecuación de decaimiento radiactivo.
- Private string NotacionCientifica; Esto será el valor de convertir la cantidad de notación científica computación a un string que muestre lo mismo.
- Private boolean exitosa; Este será el valor de saber si la operación fue exitosa o no.

### Métodos:

- Public String Simulador(): Ejecuta todos los métodos necesarios para calcular el valor por medio del simulador, al igual que la ecuacion y compara el resultado, enviando resultados al main. Ademas simula los contadores de las horas y los atomos.
- Private float ecuacion(): Recibirá como parámetros el tiempo en horas cuando termine el ciclo del simulador, y retornara el valor de la ecuación de decaimiento radiactivo.
- Private boolean comparar():Recibirá como parámetros el valor de la ecuación para devolver un valor boolean que nos indicara si el valor es menor a 1 y por tanto será el valor de si la simulación fue exitosa o no.
- Public String convertir(): En esta tarea se recibe un parámetro de tipo string para asignar los valores de horas y luego convertir la dimensional en la se encuentra a número de átomos y en otra variable su cantidad respectiva. Devuelvo un mensaje con la conversión hecha.
- Public IsotopoSimulador(): Recibe como parámetros el nombre, la vida media y la masa molar para ser el constructor de esta clase y asi poder ejecutar todos sus métodos.