Estructuras de Datos: EVALUACIÓN 1

Nombre:_____**Fecha**:_____**Calificación**: _____

Se sugiere leer primero cada problema antes de hacerlo, ya que muchas veces no se leen bien antes de comenzar, lo que origina errores. Una vez analizados los dos problemas son relativamente sencillos de resolver.

I. Se le pide que tomando como entrada un archivo de texto que contenga clases en Java, por ejemplo, algo así (50%).

Ejemplo de entrada 1:

```
package entradasalida.archivos;


import entradasalida.*;
import
estructurasdlineales.*;
import java.io.*;


public class ArchivoTexto {

    public static ArregloListaInfoEstatica leer(String archivo){
        FileReader input=null;
        int  registro=0;
        ArregloListaInfoEstatica datos=null;
        BufferedReader buffer = null;


        try {
            String cadena=null;
            input = new FileReader(archivo);
            buffer = new BufferedReader(input);
            datos=new ArregloListaInfoEstatica ((int)buffer.lines().count());
            buffer.close();
            input.close();
            input = new FileReader(archivo);
            buffer = new BufferedReader(input);
            while((cadena = buffer.readLine())!=null) {
                datos.nuevo(cadena);
            }
        } catch (FileNotFoundException e) {
            e.printStackTrace();
        } catch (IOException e) {
            e.printStackTrace();
        }
      finally {
          try{
              input.close();
              buffer.close();
            }catch(IOException e){
              e.printStackTrace();
            }
        }
        return datos;
    }
    public static void escribir(ArregloListaInfoEstatica arreglo, String archivo){
        FileWriter output=null;
        try {
            output = new FileWriter(archivo);
            for(int posicion=0;posicion<arreglo.numDatos() -1 ;posicion++) {
                output.write((String)arreglo.dato(posicion)+ "\n");
            }
            output.write((String)arreglo.dato(arreglo.numDatos() - 1));
        } catch (FileNotFoundException e) {
            e.printStackTrace();
        } catch (IOException e) {
```

```
            e.printStackTrace();
        }
        finally {
            try{
              output.close();
            }catch(IOException e){
              e.printStackTrace();
            }
        }
    }
}
```

Genere un programa en Java que:

a) El programa recibirá como entrada el archivo a analizar.
b) **Usando TDAs pilas**, validar que el archivo o la cadena de texto de la expresión aritmética cumpla con un balanceo adecuado de (...), […], {…} y de /* … */, /** …**/. Es decir, por cada uno de estos símbolos de apertura, debe existir un símbolo de cierre.
c) El programa generará como salida si el archivo o la cadena de texto cumplen o no con el balanceo adecuado de símbolos.**Se debe indicar la línea de código y el carácter en donde está el problema**. La salida deseada debe ser tal cual se pide en el ejemplo. Se pide que **cada error encontrado se vaya metiendo en un TDA cola**, para al final del proceso de análisis, se pueda hacer todo el proceso de impresiones.

Ejemplo 1 (balanceo en archivo):

…

línea 45 `try {`

`^ Faltó cerrar la llave {`

…

Ejemplo 2 (balanceo en archivo):

…..

línea 50 `} catch (FileNotFoundException e {`

`^ Faltó cerrar el paréntesis (`

….

Utilice la clase que lee un archivo de texto y lo va guardando en un arreglo. Ese le va a ir indicando la línea del error, ya al procesar cada arreglo (quiere decir que se tendrá una arreglo con arreglos dentro, por ejemplo) podrá saber el número de caracter en cuestión usando el índice del arreglo.

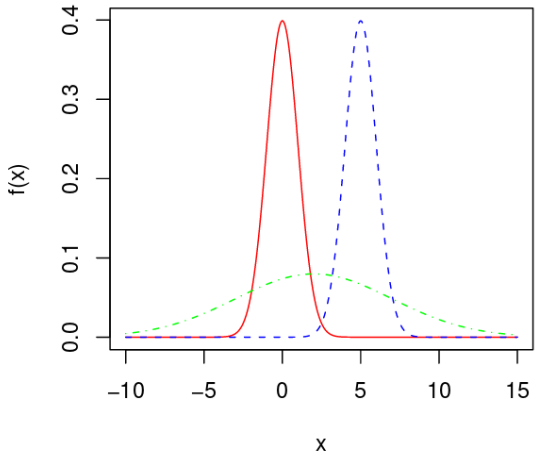
NO haga simplemente conteos de símbolos para solucionar el problema, debe usar forzosamente pilas. Si hace solo conteos de caracteres, automáticamente el programa no cumple el propósito del parcial.

Recuerde que en este examen está permitido solo el uso de arreglo (de una dimensión o varias, registros, pilas y colas).

II. La distribución normal o gaussiana es la distribución continua más importante. Se dice que una variable X se distribuye como normal con parámetros μ=2.06643303 y σ= 1.523718943 si **(50%).**

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{1}{2\sigma^2}(x-\mu)^2\right)$$

En este caso, se escribe X∼N(μ,σ). La media de la distribución normal es μ y la desviación típica es σ. El siguiente gráfico muestra la función de densidad de tres distribuciones normales con distintas medias y desviaciones típicas. Se ve que la densidad es simétrica en torno de la media.



Se le pide que:

- Almacene todo lo que se ocupe usando en arreglo numérico.
- Leyendo el archivo que se proporciona con nombre x.txt, genere los valores de f(x).
- Grafique en Excel los datos como “gráfica de dispersión” usando “x” y “f(x)”, observe la curva normal obtenida con los datos de su programa en java. Añada la gráfica obtenida como imagen en el entregable.
- Tipifique el valor de X para que ahora se comporte/transforme en una distribución normal estándar (es aquella con μ=0, σ=1) De tal forma que ahora los datos X∼N(0,1). Esto se realiza de manera sencilla calculando z=(x - μ) / σ.
- Con los datos de X transformados en Z. Recalcule el nuevo valor de la función f(x), pero ahora como f(z).
- Grafique en Excel los datos como “gráfica de dispersión” usando “z” y “f(z)”, observe la curva normal obtenida con los datos de su programa en java. Añada la gráfica obtenida como imagen en el entregable.