

Escuela Colombiana de Ingenieria Julio Garavito

**Laboratorio 2 Introducción a sistemas complejos,
Heroku, Java, Maven y GitHub**

Jimmy Armando Chirivi Nivia

Luis Daniel Benavides Navarro

Arquitectura Empresarial
Enero 2020

1 DESCRIPCION´

En este laboratorio se elaborará un aplicativo en Java el cual calculará la media y desviación estándar de unos datos numéricos ingresados por el usuario. En este caso se implementará una estructura de datos llamada lista enlazada el cual manipulará los datos para calcular dichas operaciones.

2 IMPLEMENTACION

Clase Nodo: En esta clase se crea el nodo con la información ingresada por el usuario el cual se almacena en el atributo dato además se deja espacio para almacenar el nodo siguiente.

Atributos

1. dato : float
2. next : Nodo

Metodos

1. public Nodo(float dato) : Creador de la clase
2. public void setDatoNodo(float dato) : Cambia dato del nodo
3. public float getDatoNodo() : Retorna el Dato del nodo
4. public void setNextNodo(Nodo next) : Cambia el nodo siguiente del Nodo5.
public Nodo getNextNodo() : Retorna el nodo siguiente del nodo

Clase LinkedList: En esta clase recrea una estructura de datos llamada lista encadena esta compuesta por una serie de nodos enlazados consecutivamente.

Atributos

1. nodofirst : Nodo
2. nodolast : Nodo
3. nodonew : Nodo
4. nodoremove : Nodo
5. nodoprev : Nodo6. nodoact : Nodo
7. nodonext : Nodo
8. longitud : int

Metodos

1. public LinkedList(): Creador de la clase LinkedList
2. public Nodo getFirst(): Clase encargada de retornar el primero de lalinkedlist

3. public Nodo getLast(): Clase encargada de retornar el ultimo de la linkedlist
4. public boolean add(float nodo): Clase encargada de agregar un nodo en la linkedlist
5. public boolean isEmpty(): Clase encargada de verificar si esta vacia o llenala linkedlist
6. public int size(): Clase encargada de verificar el tamaño de la linkedlist
7. public Nodo removeFirst(): Clase encargada de eliminar el primer nodo de la linkedlist
8. public Nodo removeLast(): Clase encargada de eliminar el ultimo nodo de la linkedlist
9. public Nodo remove(int index): Clase encargada de eliminar un nodo de la linkedlist

Clase principal: clase principal dedica a calcular la media y desviación de estandar de unos datos mediante la estructura de datos LinkedList

Atributos

1. list : LinkedList
2. actual : Nodo
3. siguiente : Nodo
4. Tmen : float
5. Tstdes : float
6. m : float

Metodos

1. public principal (): Creador de la clase Principal donde se crea una linkedlist vacia
2. public float mean (): Clase mean se encarga de calcular la media de los datos de la linkedlist
3. public float StandardDesviation (): Clase StandardDesviation se encarga de calcular la desviación estandar de los datos de la linkedlist
4. public LinkedList getLinkedList(): Clase getLinkedList se encarga de retornar la linkedlist actual
5. public static void contenidofile (String file): Clase contenidofile se encarga de leer los caracteres de un archivo
6. public static void main(String[] args): Clase main principal de la clase que depura la clase principal

Clase SparkWebApp: clase principal dedica a elaborar el proceso de mostrar el resultado de la media y desviación estándar mediante una pagina web

Metodos

1. `private static String inputDataPage(Request req, Response res):`
Método en cargado de crear la pagina principal y enlazar con la pagina de resultados
2. `private static String resultsPage(Request req, Response res):`
encargada de invocar la clase principal para la creacion de la linkedlist y calcular la media y la desviación estándar el cual muestra los resultados en una pagina web
3. `static int getPort():` encargada de definir el puerto lógico donde se desplegara la pagina web.