**Especificaciones Operacionales**

|  |  |  |  |
| --- | --- | --- | --- |
| **Estudiantes** | Sandra V. Londoño, Oscar C. Lopera, Yeifer A. Herrera | **Fecha** | 09/09/2017 |
| **Programa** | Programa 4 | **Programa #** | 4 |
| **Instructor** | Juan Carlos Marin | **Lenguaje** | Java |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Número del escenario** | 1 | **Objetivo de usuario** | Seleccionar la opción de menú | |
| **Objetivo del escenario** | | determinar el flujo de acción del programa | | |
| **Fuente** | **Paso** | **Acción** | | **Comentarios** |
| Sistema | 1 | El sistema solicita elegir la opción del menú | | 1. Ingresar las lista a promedia  2. Ingresar lista unica promedio  3. Salir |
| Usuario | 2 | El usuario ingresa una opción distinta a las del menú proporcionado | | Cualquier carácter alfanumérico o especial diferente a 1,2 ó 3 |
| Sistema | 3 | El sistema solicita elegir la opción del menú nuevamente | |  |
|  |  |  | |  |
| Sistema | 1 | El sistema solicita elegir la opción del menú | | 1. Ingresar las lista a promedia  2. Ingresar lista unica promedio  3. Salir |
| Usuario | 2 | El usuario ingresa la opción 3 del menú | |  |
| Sistema | 3 | El sistema termina el proceso y cierra el programa | |  |
|  |  |  | |  |
| Sistema | 1 | El sistema solicita elegir la opción del menú | | 1. Ingresar las lista a promedia  2. Ingresar lista unica promedio  3. Salir |
| Usuario | 2 | El usuario ingresa la opción 1 del menú | |  |
| Sistema | 3 | El sistema termina solicita al usuario el ingreso de la ruta del primer archivo que contiene el conjunto de datos a promediar y continúa con su operación | | se espera después del ingreso de la ruta correcta se ingrese la ruta del segundo archivo |
|  |  |  | |  |
| Sistema | 1 | El sistema solicita elegir la opción del menú | | 1. Ingresar las lista a promedia  2. Ingresar lista unica promedio  3. Salir |
| Usuario | 2 | El usuario ingresa la opción 2 del menú | |  |
| Sistema | 3 | El sistema termina solicita al usuario el ingreso de la ruta del archivo que contiene el conjunto de datos y continúa con su operación | | solo se espera el ingreso de este archivo correctamente |

|  |  |  |  |
| --- | --- | --- | --- |
| **Estudiantes** | Sandra V. Londoño, Oscar C. Lopera, Yeifer A. Herrera | **Fecha** | 21/09/2017 |
| **Programa** | Programa 4 | **Programa #** | 4 |
| **Instructor** | Juan Carlos Marin | **Lenguaje** | Java |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Número del escenario** | 2 | **Objetivo de usuario** | Ingresar los conjuntos de datos a procesar. | |
| **Objetivo del escenario** | | Validar los datos ingresados. | | |
| **Fuente** | **Paso** | **Acción** | | **Comentarios** |
| Sistema | 1 | El sistema solicita al usuario ingresar la ruta del primer archivo | | Se espera la ruta del archivo de datos a promediar (opción 1) |
| Usuario | 2 | El usuario ingresa la ruta del archivo que contiene el primer conjunto de datos | | La ruta ingresada corresponde a un archivo existente, con la extensión correcta y correcta construcción de los datos. |
| Sistema | 3 | El sistema valida el correcto ingreso de la ruta del archivo. | |  |
| Sistema | 4 | El sistema verifica que la extensión del archivo sea correcta. | |  |
| Sistema | 5 | El sistema verifica que el archivo exista. | |  |
| Sistema | 6 | El sistema valida el contenido del archivo. | |  |
| Sistema | 7 | El sistema confirma el correcto ingreso del archivo. | |  |
| Sistema | 8 | El sistema solicita ingresar la ruta del segundo archivo | |  |
|  |  |  | |  |
| Sistema | 1 | El sistema solicita al usuario ingresar la ruta del primer archivo | | Se espera la ruta del archivo de datos a promediar (opción 1) |
| Usuario | 2 | El usuario ingresa la ruta del archivo que contiene el primer conjunto de datos | | La ruta ingresada corresponde a un archivo existente, con la extensión incorrecta. |
| Sistema | 3 | El sistema valida el correcto ingreso de la ruta del archivo. | |  |
| Sistema | 4 | El sistema identifica que la extensión ingresada no es correcta. | |  |
| Sistema | 5 | El sistema muestra al usuario mensaje informando que la extensión ingresada no es correcta. | |  |
| Sistema | 6 | El sistema finaliza la ejecución del programa. | |  |
|  |  |  | |  |
| Sistema | 1 | El sistema solicita al usuario ingresar la ruta del primer archivo | | Se espera la ruta del archivo de datos a promediar (opción 1) |
| Usuario | 2 | El usuario ingresa la ruta del archivo que contiene el primer conjunto de datos | | La ruta ingresada corresponde a un archivo que no existe. |
| Sistema | 3 | El sistema valida el correcto ingreso de la ruta del archivo. | |  |
| Sistema | 4 | El sistema identifica que el archivo no existe. | |  |
| Sistema | 5 | El sistema muestra mensaje informando al usuario que el archivo no existe. | |  |
| Sistema | 6 | El sistema finaliza la ejecución del programa. | |  |
|  |  |  | |  |
| Sistema | 1 | El sistema solicita al usuario ingresar la ruta del primer archivo | | se espera la ruta del archivo de datos a promediar (opción 1) |
| Usuario | 2 | El usuario ingresa la ruta del archivo que contiene el primer conjunto de datos | | La ruta ingresada corresponde a un archivo existente, con la extensión correcta, pero el archivo se encuentra vacío. |
| Sistema | 3 | El sistema valida el correcto ingreso de la ruta del archivo. | |  |
| Sistema | 4 | El sistema verifica que el archivo existe. | |  |
| Sistema | 5 | El sistema valida el contenido del archivo. | |  |
| Sistema | 6 | El sistema identifica que el archivo se encuentra vacío. | |  |
| Sistema | 7 | El sistema muestra mensaje al usuario indicando que el archivo está vacío. | |  |
| Sistema | 8 | El sistema finaliza la ejecución del programa. | |  |
|  |  |  | |  |
| Sistema | 1 | El sistema solicita al usuario ingresar la ruta del primer archivo | | Se espera la ruta del archivo de datos a promediar (opción 1) |
| Usuario | 2 | El usuario ingresa la ruta del archivo que contiene el primer conjunto de datos. | | La ruta ingresada corresponde a un archivo existente, con la extensión correcta, pero con el contenido del archivo incorrecto. |
| Sistema | 3 | El sistema valida el correcto ingreso de la ruta del archivo. | |  |
| Sistema | 4 | El sistema verifica que el archivo existe. | |  |
| Sistema | 5 | El sistema valida el contenido del archivo. | |  |
| Sistema | 6 | El sistema identifica que el contenido del archivo no es correcto. | |  |
| Sistema | 7 | El sistema muestra mensaje al usuario indicando que el contenido del archivo no es correcto. | | El archivo incorrectamente construido |
| Sistema | 8 | El sistema finaliza la ejecución del programa. | |  |
|  |  |  | |  |
| Sistema | 1 | El sistema solicita al usuario ingresar la ruta del segundo archivo | | Se espera la ruta del archivo de datos sobre los cuales se debe promediar cada dato del primer archivo (opción 1--segundo archivo) |
| Usuario | 2 | El usuario ingresa la ruta del archivo que contiene el segundo conjunto de datos | | La ruta ingresada corresponde a un archivo existente, con la extensión correcta y correcta construcción de los datos. |
| Sistema | 3 | El sistema valida el correcto ingreso de la ruta del archivo. | |  |
| Sistema | 4 | El sistema verifica que la extensión del archivo sea correcta. | |  |
| Sistema | 5 | El sistema verifica que el archivo exista. | |  |
| Sistema | 6 | El sistema valida el contenido del archivo. | |  |
| Sistema | 7 | El sistema confirma el correcto ingreso del archivo. | |  |
| Sistema | 8 | El sistema solicita ingresar la ruta del segundo archivo | |  |
|  |  |  | |  |
| Sistema | 1 | El sistema solicita al usuario ingresar la ruta del segundo archivo | | Se espera la ruta del archivo de datos sobre los cuales se debe promediar cada dato del primer archivo (opción 1--segundo archivo) |
| Usuario | 2 | El usuario ingresa la ruta del archivo que contiene el segundo conjunto de datos | | La ruta ingresada corresponde a un archivo existente, con la extensión incorrecta. |
| Sistema | 3 | El sistema valida el correcto ingreso de la ruta del archivo. | |  |
| Sistema | 4 | El sistema identifica que la extensión ingresada no es correcta. | |  |
| Sistema | 5 | El sistema muestra al usuario mensaje informando que la extensión ingresada no es correcta. | |  |
| Sistema | 6 | El sistema finaliza la ejecución del programa. | |  |
|  |  |  | |  |
| Sistema | 1 | El sistema solicita al usuario ingresar la ruta del segundo archivo | | Se espera la ruta del archivo de datos sobre los cuales se debe promediar cada dato del primer archivo (opción 1--segundo archivo) |
| Usuario | 2 | El usuario ingresa la ruta del archivo que contiene el segundo conjunto de datos | | La ruta ingresada corresponde a un archivo que no existe. |
| Sistema | 3 | El sistema valida el correcto ingreso de la ruta del archivo. | |  |
| Sistema | 4 | El sistema identifica que el archivo no existe. | |  |
| Sistema | 5 | El sistema muestra mensaje informando al usuario que el archivo no existe. | |  |
| Sistema | 6 | El sistema finaliza la ejecución del programa. | |  |
|  |  |  | |  |
| Sistema | 1 | El sistema solicita al usuario ingresar la ruta del segundo archivo | | Se espera la ruta del archivo de datos sobre los cuales se debe promediar cada dato del primer archivo (opción 1--segundo archivo) |
| Usuario | 2 | El usuario ingresa la ruta del archivo que contiene el segundo conjunto de datos | | La ruta ingresada corresponde a un archivo existente, con la extensión correcta, pero el archivo se encuentra vacío. |
| Sistema | 3 | El sistema valida el correcto ingreso de la ruta del archivo. | |  |
| Sistema | 4 | El sistema verifica que el archivo existe. | |  |
| Sistema | 5 | El sistema valida el contenido del archivo. | |  |
| Sistema | 6 | El sistema identifica que el archivo se encuentra vacío. | |  |
| Sistema | 7 | El sistema muestra mensaje al usuario indicando que el archivo está vacío. | |  |
| Sistema | 8 | El sistema finaliza la ejecución del programa. | |  |
|  |  |  | |  |
| Sistema | 1 | El sistema solicita al usuario ingresar la ruta del segundo archivo | | Se espera la ruta del archivo de datos sobre los cuales se debe promediar cada dato del primer archivo (opción 1--segundo archivo) |
| Usuario | 2 | El usuario ingresa la ruta del archivo que contiene el segundo conjunto de datos | | La ruta ingresada corresponde a un archivo existente, con la extensión correcta, pero con el contenido del archivo incorrecto. |
| Sistema | 3 | El sistema valida el correcto ingreso de la ruta del archivo. | |  |
| Sistema | 4 | El sistema verifica que el archivo existe. | |  |
| Sistema | 5 | El sistema valida el contenido del archivo. | |  |
| Sistema | 6 | El sistema identifica que el contenido del archivo no es correcto. | |  |
| Sistema | 7 | El sistema muestra mensaje al usuario indicando que el contenido del archivo no es correcto. | | El archivo incorrectamente construido |
| Sistema | 8 | El sistema finaliza la ejecución del programa. | |  |
|  |  |  | |  |
| Sistema | 1 | El sistema solicita al usuario ingresar la ruta del archivo con el conjunto de datos a operar | | Se espera la ruta del archivo de datos único de datos a operar (opción 2) |
| Usuario | 2 | El usuario ingresa la ruta del archivo que contiene el primer conjunto de datos | | La ruta ingresada corresponde a un archivo existente, con la extensión correcta y correcta construcción de los datos. |
| Sistema | 3 | El sistema valida el correcto ingreso de la ruta del archivo. | |  |
| Sistema | 4 | El sistema verifica que la extensión del archivo sea correcta. | |  |
| Sistema | 5 | El sistema verifica que el archivo exista. | |  |
| Sistema | 6 | El sistema valida el contenido del archivo. | |  |
| Sistema | 7 | El sistema confirma el correcto ingreso del archivo. | |  |
| Sistema | 8 | El sistema solicita ingresar la ruta del segundo archivo | |  |
|  |  |  | |  |
| Sistema | 1 | El sistema solicita al usuario ingresar la ruta del archivo con el conjunto de datos a operar | | Se espera la ruta del archivo de datos único de datos a operar (opción 2) |
| Usuario | 2 | El usuario ingresa la ruta del archivo que contiene el primer conjunto de datos | | La ruta ingresada corresponde a un archivo existente, con la extensión incorrecta. |
| Sistema | 3 | El sistema valida el correcto ingreso de la ruta del archivo. | |  |
| Sistema | 4 | El sistema identifica que la extensión ingresada no es correcta. | |  |
| Sistema | 5 | El sistema muestra al usuario mensaje informando que la extensión ingresada no es correcta. | |  |
| Sistema | 6 | El sistema finaliza la ejecución del programa. | |  |
|  |  |  | |  |
| Sistema | 1 | El sistema solicita al usuario ingresar la ruta del archivo con el conjunto de datos a operar | | Se espera la ruta del archivo de datos único de datos a operar (opción 2) |
| Usuario | 2 | El usuario ingresa la ruta del archivo que contiene el primer conjunto de datos | | La ruta ingresada corresponde a un archivo que no existe. |
| Sistema | 3 | El sistema valida el correcto ingreso de la ruta del archivo. | |  |
| Sistema | 4 | El sistema identifica que el archivo no existe. | |  |
| Sistema | 5 | El sistema muestra mensaje informando al usuario que el archivo no existe. | |  |
| Sistema | 6 | El sistema finaliza la ejecución del programa. | |  |
|  |  |  | |  |
| Sistema | 1 | El sistema solicita al usuario ingresar la ruta del archivo con el conjunto de datos a operar | | Se espera la ruta del archivo de datos único de datos a operar (opción 2) |
| Usuario | 2 | El usuario ingresa la ruta del archivo que contiene el primer conjunto de datos | | La ruta ingresada corresponde a un archivo existente, con la extensión correcta, pero el archivo se encuentra vacío. |
| Sistema | 3 | El sistema valida el correcto ingreso de la ruta del archivo. | |  |
| Sistema | 4 | El sistema verifica que el archivo existe. | |  |
| Sistema | 5 | El sistema valida el contenido del archivo. | |  |
| Sistema | 6 | El sistema identifica que el archivo se encuentra vacío. | |  |
| Sistema | 7 | El sistema muestra mensaje al usuario indicando que el archivo está vacío. | |  |
| Sistema | 8 | El sistema finaliza la ejecución del programa. | |  |
|  |  |  | |  |
| Sistema | 1 | El sistema solicita al usuario ingresar la ruta del archivo con el conjunto de datos a operar | | Se espera la ruta del archivo de datos único de datos a operar (opción 2) |
| Usuario | 2 | El usuario ingresa la ruta del archivo que contiene el primer conjunto de datos. | | La ruta ingresada corresponde a un archivo existente, con la extensión correcta, pero con el contenido del archivo incorrecto. |
| Sistema | 3 | El sistema valida el correcto ingreso de la ruta del archivo. | |  |
| Sistema | 4 | El sistema verifica que el archivo existe. | |  |
| Sistema | 5 | El sistema valida el contenido del archivo. | |  |
| Sistema | 6 | El sistema identifica que el contenido del archivo no es correcto. | |  |
| Sistema | 7 | El sistema muestra mensaje al usuario indicando que el contenido del archivo no es correcto. | | El archivo incorrectamente construido |
| Sistema | 8 | El sistema finaliza la ejecución del programa. | |  |

**Especificaciones Funcionales**

|  |  |  |  |
| --- | --- | --- | --- |
| **Estudiantes** | Sandra V. Londoño, Oscar C. Lopera, Yeifer A. Herrera | **Fecha** | 09/09/2017 |
| **Programa** | Programa 4 | **Programa #** | 4 |
| **Instructor** | Juan Carlos Marin | **Lenguaje** | Java |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | | | | | | |
| Clase Programa4 | |  | | | | |
|  | | Esta es la clase principal del programa. En esta clase se utilizan objetos de la clase Nodo y de la clase ListaLigada con el objetivo de realizar los procesos administrativos de información y cálculo de datos para entregar al usuario la respuesta esperada si se ingresan los datos de manera adecuada. En esta clase se encuentra la interacción con el usuario para el ingreso de la información a ser leída. La utilización y los métodos que se requieren para encontrar los valores que se deben calcular y la forma como el sistema muestra los resultados obtenidos. | | | | |
|  | |  | | | | |
|  | | | | | | |
| Metodo | | | | | | |
|  | public static ListaLigada LeerArchivo (String archivo) | | | Este método se encarga de leer el archivo que se encuentra en la ruta ingresada por el usuario y llenar una lista ligada con la información contenido en el archivo si tiene la información adecuada para el proceso.  Atributos:  String archivo: En esta cadena de caracteres se encuentra la ruta del archivo que va a ser leído por el sistema | | |
|  | public static void main() | | | Este es el programa principal de ejecución en el cual se interactúa con el usuario y además se realizan las invocaciones de métodos para realizar las actividades que la aplicación cumple. | | |
|  | public static boolean verificarExtension(String ruta) | | | Este metodo verifica que la extensión del archivo ingresado por el usuario sea la adecuada para el proceso de ejecución.  Atributos:  String ruta: En esta cadena de caracteres se encuentra la ruta del archivo del cual se debe evaluar la extensión que tiene. | | |
|  | public static double desviacionEstandar(ListaLigada lista) | | | Este método realiza el cálculo de la desviación estándar con la información contenida en una lista ligada.  Atributos:  ListaLigada lista: En esta lista ligada se encuentran los datos con los cuales se va a calcular la desviación estándar. | | |
|  | public static double suma(ListaLigada lista) | | | Este método realiza la suma de todos los datos encontrados en una lista ligada.  Atributos:  ListaLigada lista:En esta lista ligada se encuentran los datos que se van a sumar para retornar el total. | | |
|  | public static ListaLigada promediar\_Listas(ListaLigada x,ListaLigada y) | | | En este método se realiza el promedio entre 2 listas ligadas termino a termino y guarda la información en una nueva lista ligada que se retornara en el final.  Atributos:  ListaLigada x: Segunda lista de datos con la cual se calculara el prmedio.  ListaLigada y:Primera lista de datos con la cual se calculara el prmedio. | | |
|  | public static ListaLigada calcular\_ln\_cada\_dato(ListaLigada lista) | | | En este método se calcula el logaritmo natural de los datos contenidos en una lista ligada ingresa como parámetro y se retorna una nueva lista ligada con los valores calculados.  Atribitos:  ListaLigada lista: Lista ligada con los datos a los cuales se les debe calcular el logaritmo natural. | | |
|  | public static void mostrar\_datos(double VS ,double S, double M ,double L, double VL) | | | En este método se realiza la impresión en pantalla para el usuario de la información que calculo el sistema y es la esperada como respuesta.  Atributos:  double VS: Dato calculado por el sistema que se debe mostrar en pantalla y representa el valor VS.  double S: Dato calculado por el sistema que se debe mostrar en pantalla y representa el valor S.  double M: Dato calculado por el sistema que se debe mostrar en pantalla y representa el valor M.  double L: Dato calculado por el sistema que se debe mostrar en pantalla y representa el valor L.  double VL: Dato calculado por el sistema que se debe mostrar en pantalla y representa el valor VL. | | |

**Especificación de Estados**

|  |  |  |  |
| --- | --- | --- | --- |
| **Estudiantes** | Sandra V. Londoño, Oscar C. Lopera, Yeifer A. Herrera | **Fecha** | 21/09/2017 |
| **Programa** | Programa 4 | **Programa #** | 4 |
| **Instructor** | Juan Carlos Marin | **Lenguaje** | Java |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | | |  | | | | |
| **Nombre del estado** | | | | **Descripción** | | | |
|  | | | |  | | | |
| Nuevo nodo | | | | Se crea un nuevo objeto de la clase nodo con dato en cero (0) y siguiente en nulo. | | | |
| Dato modificado | | | | Se modifica el valor del dato, el valor de siguiente continúa nulo. | | | |
| Siguiente modificado | | | | Se modifica el valor de siguiente. | | | |
|  | | | |  | | | |
| **Función/Parámetros** | | | | **Descripción** | | | |
|  | | | |  | | | |
| setSiguiente(Nodo X) | | | | Modifica el valor del siguiente en un objeto tipo Nodo. | | | |
| SetDato(double dato) | | | | Modifica el valor del dato en un objeto tipo Nodo. | | | |
|  | | | |  | | | |
|  | | | |  | | | |
|  | | | |  | | | |
| **Estados/Próximos estados** | | | | **Condición de transisción** | | **Acción** | |
| Nuevo nodo/Dato modificado | | | | Se llama la función setDato y se le envía como parámetro el nuevo dato. | | El dato es modificado. | |
| Nuevo nodo/Siguiente modificado | | | | Se llama la función setSiguiente y se le envía como parámetro el nodo al que hará referencia. | | El siguiente es modificado. | |
| Dato modificado/Siguiente modificado | | | | Se llama la función setSiguiente y se le envía como parámetro el nodo al que hará referencia. | | El siguiente es modificado. | |
| Siguiente modificado/Dato modificado | | | | Se llama la función setDato y se le envía como parámetro el nuevo dato. | | El dato es modificado. | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Estudiantes** | Sandra V. Londoño, Oscar C. Lopera, Yeifer A. Herrera | **Fecha** | 21/09/2017 |
| **Programa** | Programa 4 | **Programa #** | 4 |
| **Instructor** | Juan Carlos Marin | **Lenguaje** | Java |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | | |  | | | | |
| **Nombre de Estados** | | | | **Descripción** | | | |
|  | | | |  | | | |
| Lista vacía | | | | Se crea un nuevo objeto de la clase lista con cabeza[0] en nulo (null) y cabeza[1] en nulo (null). | | | |
| Lista modificada | | | | Se modifica la clase lista con cabeza[0] en nodo y cabeza[1] en nodo. | | | |
|  | | | |  | | | |
| **Función/Parámetros** | | | | **Descripción** | | | |
|  | | | |  | | | |
| insertarNodo(double dato) | | | | Modifica el valor cabeza[0] en nodo de valor dato y cabeza[1] en nodo de valor dato.  . | | | |
| Delete(double dato) | | | | Modifica el valor cabeza[0] en nulo (null) y cabeza[1] en nulo (null).. | | | |
|  | | | |  | | | |
|  | | | |  | | | |
|  | | | |  | | | |
| **Estados/Próximos estados** | | | | **Estados/Próximos estados** | | **Acción** | |
| Lista vacia/Lista modificada | | | | Se llama la función insertarNodo(double dato) y se le envía como parámetro el nuevo dato. | | la lista pasa a estado cabeza[0] = nodo de valor dato y cabeza[1] = nodo de valor dato | |
| Lista modificada  /Lista vacia/ | | | | Se llama la función Delete(double dato) y se le envía como parámetro el dato a eliminar | | la lista con un solo dato pasa a estado cabeza[0] =nulo (null) y cabeza[1] = nulo (null) | |

**Especificación Lógica**

|  |  |  |  |
| --- | --- | --- | --- |
| **Estudiantes** | Sandra V. Londoño, Oscar C. Lopera, Yeifer A. Herrera | **Fecha** | 21/09/2017 |
| **Programa** | Programa 4 | **Programa #** | 4 |
| **Instructor** | Juan Carlos Marin | **Lenguaje** | Java |

|  |  |
| --- | --- |
| **Referencias de**  **Diseño** | Método Suma: |
|  | En este método se recibe una lista ligada con los datos que se sean sumar y se retorna un |
|  | valor de tipo double con la suma de los datos en la lista ligada. |

|  |  |
| --- | --- |
| **Parámetros** | Lista Ligada lista |

Metodo double Suma(listaLigada lista)

{

entero tamaño = lista.tamaño();

double suma = 0;

para (entero i = 1 hasta i <= tamaño con incremento 1) haga

{

suma = suma + lista [i];

}

retorne (suma) ;

}

|  |  |
| --- | --- |
| **Referencias de**  **Diseño** | Método Calcular el promedio entre 2 Listas Ligadas |
|  | En este método se recibe como parametro 2 listas ligadas para realizar la división |
|  | términos a término de los elementos de las listas ligadas. |

|  |  |
| --- | --- |
| **Parámetros** | Lista Ligada x, Lista Ligada Y |

Metodo listaLigada Calc\_promedio(listaLigada x, listaLigada y)

{

listaLigada resultado = nuevo listaLigada();

Si (x.cant\_datos() == y.cant\_datos()) entonces

{

nodo nodo\_x = x.primerNodo();

nodo nodo\_y = y.primerNodo();

Mientras (nodo\_y != null) haga

{

resultado.agregar( nodo\_x.dato()/nodo\_y.dato());

nodo\_x=nodo\_x.siguiente();

nodo\_y=nodo\_y.siguiente();

}

}

Si No

{

Muestre: “Los tamaños de las listas NO son iguales”;

}

retorne (resultado);

}

|  |  |
| --- | --- |
| **Referencias de**  **Diseño** | Método Calcular el logaritmo natural de una Listas Ligadas |
|  | En este método se recibe como parametro 1 lista ligada para realizar el calculo del |
|  | logaritmo natural de cada dato en la lista. |

|  |  |
| --- | --- |
| **Parámetros** | Lista Ligada x |

Metodo listaLigada Calc\_loga\_N(listaLigada x)

{

listaLigada resultado = nuevo listaLigada();

nodo nodo\_aux = x.primero();

Mientras (nodo\_aux != null) haga

{

resultado.agregar(LogNatural(nodo\_aux.dato());

}

retorne(nodo\_aux);

}

|  |  |
| --- | --- |
| **Referencias de**  **Diseño** | Método Calcular Media |
|  | En este método se recibe como parametro 1 lista ligada para realizar el calculo de la |
|  | media entre todos los datos que contiene la lista. |

|  |  |
| --- | --- |
| **Parámetros** | Lista Ligada x |

Metodo doble Media(listaLigada x)

{

doble resultado = 0;

resultado = suma(x)

\_\_\_\_\_\_\_\_\_\_\_\_ ;

x.cant\_datos()

retorne (resultado);

}

|  |  |
| --- | --- |
| **Referencias de**  **Diseño** | Método Calcular Desviación Estándar |
|  | En este método se recibe como parametro 1 lista ligada para realizar el cálculo de la |
|  | desviación estándar con la información contenida en la lista. |

|  |  |
| --- | --- |
| **Parámetros** | Lista Ligada x |

Método doble Desviacion\_Estandar(listaLigada x)

{

doble media = Media(x);

doble suma = 0;

nodo nodo = x.primero();

Mientras (nodo != nulo)

{

suma = suma + (Potencia((nodo.dato() - media), 2));

nodo = nodo.siguiente();

}

suma = suma

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ ;

(lista.getCant\_datos()-1)

suma = Raiz(suma,2);

retorne (suma);

}

|  |  |
| --- | --- |
| **Referencias de**  **Diseño** | Método Mostrar Resultados |
|  | En este método se imprimen en pantalla los resultados calculados por el sistema. |
|  |  |

|  |  |
| --- | --- |
| **Parámetros** | Doble VS, Doble S, Doble M, Double L, Doble VL |

Método Mostrar\_resultados(doble VS ,doble S, doble M ,doble L, doble VL)

{

Mostrar. “ El valor calculado de VS es: “ + VS;

Mostrar. “ El valor calculado de S es: “ + S;

Mostrar. “ El valor calculado de M es: “ + M;

Mostrar. “ El valor calculado de L es: “ + L;

Mostrar. “ El valor calculado de VL es: “ + VL;

}

|  |  |
| --- | --- |
| **Referencias de**  **Diseño** | Metodo Main: |
|  | En este método se realiza la interacción con el usuario para el ingreso de los datos |
|  | necesarios por el sistema para brindar el resultado esperado. En este método se realizan |
|  | los proceso, invocación de métodos y cálculos necesarios para verificar la entrada del |
|  | usuario y brindar la respuesta de cálculos esperados por el cliente. |

|  |  |
| --- | --- |
| **Parámetros** | Este método no tiene parámetros para su ejecución. |
|  |  |

Método Principal()

{

listaLigada lista\_x = nuevo listaLigada();

listaLigada lista\_y = nuevo listaLigada();

entero menu = 0;

Mientras (menu == 0)

{

entero respuestas: Muestre: "Selecciones una obcion:

1. Ingresar las lista a promediar

2. Ingresar lista unica promedio

3. Salir";

Switch (respuesta)

{

Caso "1":

{

texto entrada\_x= Muestre: "Ingrese la ruta del archivo 1: ";

lista\_x = LeerArchivo(entrada\_x);

texto entrada\_y = Muestre: "Ingrese la ruta del archivo 2: ";

lista\_y = LeerArchivo(entrada\_y);

menu = 1;

}

Caso "2":

{

texto entrada\_x = Muestre: "Ingrese la ruta del archivo 1: ";

lista\_x = LeerArchivo(entrada\_x);

menu = 2;

}

Caso "3":

{

Muestre:”Usted ha salido del programa”;

Salir();

menu = 3;

}

}

listaLigada datos = nuevo listaLigada();

Si (menu == 1) entonces

{

datos = Calc\_promedio(lista\_x, lista\_y);

}

Si No

{

datos = lista\_x;

}

datos = Calc\_loga\_N(datos);

doble AVG = Media(datos);

doble desviacion = Desviacion\_Estandar(datos);

doble log\_N\_VS = AVG - (2 \* desviacion);

doble log\_N\_S = AVG - desviacion;

doble log\_N\_M = AVG;

doble log\_N\_L = AVG + desviacion;

doble log\_N\_VL = AVG + (2 \* desviacion);

doble VS = Potencia(Math.E, lnVS);

doble S = Potencia(Math.E, lnS);

doble M = Potencia(Math.E, lnM);

doble L = Potencia(Math.E, lnL);

doble VL = Potencia(Math.E, lnVL);

Mostrar\_resultados(VS, S, M, L, VL);

}

}