Functional Specification Template

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Student** | | | Juan Alberto Gutierrez Canto | | **Date** | 13/07/2016 |
| **Program** | | | Normalizacion | | **Program #** | 9A |
| **Instructor** | | | Jorge Rafael Aguilar Cisneros | | **Language** | C++ |
|  | | | | | | |
| **Class Name** | | Int ISEMPTY() | | | | |
| **Parent Class** | | N/A | | | | |
|  | | | | | | |
| **Attributes** | | | | | | |
|  | **Declaration** | | | **Description** | | |
|  | Struc nodo Raiz | | | El nodo de el comienzo o el raiz. | | |
|  | | | | | | |
| **Items** | | | | | | |
|  | **Declaration** | | | **Description** | | |
|  | Return | | | Regresa un valor declarado. | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Class Name** | | Struct nodo | |
| **Parent Class** | | N/A | |
|  | | | |
| **Attributes** | | | |
|  | **Declaration** | | **Description** |
|  | Double numerox; | | Numero de cada nodo llamado x. |
|  | Double numeroy; | | Numero de cada nodo llamado y. |
|  | Struct nodo sig; | | Apuntador a la siguiente lista; |
|  | | | |
| **Items** | | | |
|  | **Declaration** | | **Description** |
|  | N/A | |  |

|  |  |  |  |
| --- | --- | --- | --- |
| **Class Name** | | Void INSERT() | |
| **Parent Class** | | N/A | |
|  | | | |
| **Attributes** | | | |
|  | **Declaration** | | **Description** |
|  | Struct nodo nuevo; | | Nodo para nuevos datos; |
|  | Double datox; | | Dato x, mandado del usuario. |
|  | Double datoy; | | Dato y, mandado del usuario. |
|  | | | |
| **Items** | | | |
|  | **Declaration** | | **Description** |
|  | ISEMPTY() | | Para revisar si la lista esta vacía. |
|  | Malloc() | | Genera espacio en memoria. |
|  | Sizeof() | | Regresa el tamaño de un dato. |

|  |  |  |  |
| --- | --- | --- | --- |
| **Class Name** | | Void REMUEVE() | |
| **Parent Class** | | N/A | |
|  | | | |
| **Attributes** | | | |
|  | **Declaration** | | **Description** |
|  | Struct nodo \*elimina; | | Nodo a eliminar de la lista. |
|  | | | |
| **Items** | | | |
|  | **Declaration** | | **Description** |
|  | ISEMPTY() | | Regresa si la lista esta vacía =1, o si no =0. |
|  | Free() | | Limpia un espacio en memoria. |

|  |  |  |  |
| --- | --- | --- | --- |
| **Class Name** | | Void PEDIRDATOS() | |
| **Parent Class** | | N/A | |
|  | | | |
| **Attributes** | | | |
|  | **Declaration** | | **Description** |
|  | Doublé datox; | | Dato x, introducido por el usuario. |
|  | Double datoy; | | Dato y, introducido por el usuario. |
|  | Long Long n; | | Numero de datos que introducirá el usuario. |
|  | Long Long i; | | Contador de datos. |
|  | | | |
| **Items** | | | |
|  | **Declaration** | | **Description** |
|  | INSERT() | | Inserta los datos en la lista ligada. |

|  |  |  |  |
| --- | --- | --- | --- |
| **Class Name** | | Void BURBUJALTST() | |
| **Parent Class** | | N/A | |
|  | | | |
| **Attributes** | | | |
|  | **Declaration** | | **Description** |
|  | Struct nodo \*inicio; | | Nodo para el comienzo de la lista. |
|  | Struct nodo \*siguiente; | | Nodo para comparar la lista. |
|  | Doublé datox; | | Dato auxiliar para hacer el cambio de elementos en x. |
|  | Long i; | | Contador de números dentro de la lista. |
|  | Long j; | | Contador de números dentro de la lista. |
|  | Doublé datoy; | | Dato auxiliar para hacer el cambio de elementos en y. |
|  | | | |
| **Items** | | | |
|  | **Declaration** | | **Description** |
|  | N/A | |  |

|  |  |  |  |
| --- | --- | --- | --- |
| **Class Name** | | double NORMALIZACION() | |
| **Parent Class** | | N/A | |
|  | | | |
| **Attributes** | | | |
|  | **Declaration** | | **Description** |
|  | double xavg | | Variable para el average de los datos x. |
|  | double varianza | | Variable para almacenar la varianza. |
|  | double dest | | Variable para almacenar la desviación estandar de la función. |
|  | double Q | | Calidad de los datos normalizados. |
|  | double simx2 | | Regla de Simpson con x^2 |
|  | double p | | Probabilidad tail. |
|  | Nodo inicio | | Tabla de valores |
|  | | | |
| **Items** | | | |
|  | **Declaration** | | **Description** |
|  | AVG() | | Saca el average de una serie de datos. |
|  | VARIANZA() | | Calcula la varianza de una serie de datos. |
|  | SECNORMAL() | | Calcula los segmentos y la calidad de los datos. |
|  | REGLASMPX2() | | Regal de Simpson con x^2 |
|  | PROBABILIDAD() | | Probilidad de una serie de datos. |
|  | Sqrt() | | Calcula la raíz cuadrada de una serie de datos. |
|  | Return | | Regresa un dato de tipo float. |

|  |  |  |  |
| --- | --- | --- | --- |
| **Class Name** | | double AVG | |
| **Parent Class** | | N/A | |
|  | | | |
| **Attributes** | | | |
|  | **Declaration** | | **Description** |
|  | Struct nodo \*inicio; | | Nodo para el comienzo de la lista. |
|  | Doublé avgdata; | | Contador para el dato. |
|  | long i | | Utilizado para los ciclos for |
|  |  | |  |
|  |  | |  |
|  |  | |  |
|  | | | |
| **Items** | | | |
|  | **Declaration** | | **Description** |
|  | Return | | Regresa un dato en especifico |

|  |  |  |  |
| --- | --- | --- | --- |
| **Class Name** | | double VARIANZA | |
| **Parent Class** | | N/A | |
|  | | | |
| **Attributes** | | | |
|  | **Declaration** | | **Description** |
|  | Struct nodo \*inicio; | | Nodo para el comienzo de la lista. |
|  | Doublé varianza; | | Contador para el dato, y formulas. |
|  | long i | | Utilizado para los ciclos for |
|  | Doublé avgdata | | Utilizado para el average |
|  |  | |  |
|  |  | |  |
|  | | | |
| **Items** | | | |
|  | **Declaration** | | **Description** |
|  | Return | | Regresa un dato en especifico |
|  | Pow | | Para elevar un termino a un exponente dado |

|  |  |  |  |
| --- | --- | --- | --- |
| **Class Name** | | double SECNORMAL() | |
| **Parent Class** | | N/A | |
|  | | | |
| **Attributes** | | | |
|  | **Declaration** | | **Description** |
|  | Struct nodo \*inicio; | | Nodo para el comienzo de la lista. |
|  | Doublé tablavaloresnor | | Valores para hacer la tabla de valores. |
|  | Doublé sumterm. | | Suma de valores de los términos. |
|  | Doublé rglsmps | | Valores para los rangos comensando en -4 hasta 4 |
|  | long segmentos | | Numero de segmentos |
|  |  | |  |
|  | | | |
| **Items** | | | |
|  | **Declaration** | | **Description** |
|  | Return | | Regresa un dato en especifico |
|  | CALRANGOS() | | Calcula los rangos para S segementos |
|  | REGLASMPNOR() | | Calcula la regla de Simpson conun valor determinado. |

|  |  |  |  |
| --- | --- | --- | --- |
| **Class Name** | | int CALRANGOS() | |
| **Parent Class** | | N/A | |
|  | | | |
| **Attributes** | | | |
|  | **Declaration** | | **Description** |
|  | Int n | | Numero de datos. |
|  | Int s | | Segmentos de la lista |
|  | Doublé tablavaloresnor | | Se utiliza la tabla. |
|  | Int i | | Se utiliza para los ciclos |
|  |  | |  |
|  |  | |  |
|  | | | |
| **Items** | | | |
|  | **Declaration** | | **Description** |
|  | Return | | Regresa un dato en especifico |

|  |  |  |  |
| --- | --- | --- | --- |
| **Class Name** | | Double REGLASMPNOR () | |
| **Parent Class** | | N/A | |
|  | | | |
| **Attributes** | | | |
|  | **Declaration** | | **Description** |
|  | Doublé xlow | | Valor de x bajo |
|  | Double xhigh | | Valor de x alto |
|  | Double N | | Numero de bloques |
|  | Double W | | Pesos de incremento. |
|  | Doublé fija | | Varible fija de la función que corresponde a |
|  | Double x2 | | Valor para la ecuacion |
|  | Double ex | | Valor de e para la ecuación |
|  | Doublé fx | | Representa el valor de toda la formula |
|  | Double term | | Termino de la función |
|  | Double sumterm | | Sumatoria de términos |
|  | Bool negative | | Utilizado para saber que variable entre si positiva o negativa |
|  | Int i | | Utilizado para los ciclos |
|  | | | |
| **Items** | | | |
|  | **Declaration** | | **Description** |
|  | Return | | Regresa un dato en especifico |
|  | Sqrt() | | Regresa la raíz cuadrada de un determinado numero. |
|  | Pow() | | Regresa el cuadrado de un numero. |
|  | Exp() | | Regresa el valor de e a un exponente dado. |
|  |  | |  |

|  |  |  |  |
| --- | --- | --- | --- |
| **Class Name** | | Double REGLASPMX2() | |
| **Parent Class** | | N/A | |
|  | | | |
| **Attributes** | | | |
|  | **Declaration** | | **Description** |
|  | Doublé xlow | | Valor de x bajo |
|  | Double xhigh | | Valor de x alto |
|  | Double N | | Numero de bloques |
|  | Double n | | Grados de libertad |
|  | Double tfija | | Valor de t fijo con la condición de |
|  | Doublé fijo | | Valor fijo de 2^n/2 |
|  | Double xan | | Valor de x^n/2-1 |
|  | Double exa | | Valor de e a el exponente –(x/2) |
|  | Double fx | | Valor de la ecuacion |
|  | Double term | | Valor del termino de la ecuacion |
|  | Bool negative | | Valor para saber si es negativo o positovo xhigh |
|  | Int i | | Utilizado para los ciclos |
|  | Double sumter | | Valor que suma los términos |
|  | | | |
| **Items** | | | |
|  | **Declaration** | | **Description** |
|  | Return | | Regresa un dato en especifico |
|  | Pow() | | Eleva un valor a el exponente dado |
|  | Exp(( | | Eleva la variable e a un exponente determinado. |

|  |  |  |  |
| --- | --- | --- | --- |
| **Class Name** | | double PROBABILIDAD() | |
| **Parent Class** | | N/A | |
|  | | | |
| **Attributes** | | | |
|  | **Declaration** | | **Description** |
|  | Double prob | | Data para calcular la probabilidad |
|  |  | |  |
|  |  | |  |
|  |  | |  |
|  |  | |  |
|  |  | |  |
|  | | | |
| **Items** | | | |
|  | **Declaration** | | **Description** |
|  | Return | | Regresa un dato en especifico |

|  |  |  |  |
| --- | --- | --- | --- |
| **Class Name** | | int main() | |
| **Parent Class** | | N/A | |
|  | | | |
| **Attributes** | | | |
|  | **Declaration** | | **Description** |
|  |  | |  |
|  | | | |
| **Items** | | | |
|  | **Declaration** | | **Description** |
|  | Freeopen() | | Leer un archivo de texto. |
|  | PEDIRDATOS() | | Pedir los datos al usuario. |
|  | BURBUJALST() | | Ordenar la lista. |
|  | NORMALIZACION() | | Hace la normalización de los datos ingresados. |
|  | Printf() | | Imprime datos a pantalla. |