Functional Specification Template

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Student** | | | Juan Alberto Gutierrez Canto | | **Date** | 23/06/2016 |
| **Program** | | | List Sort | | **Program #** | 8A |
| **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** |
|  | Printf() | | Imprimir los resultados. |

|  |  |  |  |
| --- | --- | --- | --- |
| **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. |