***Functional Specification Template***

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Student** | |  | | | **Program #** | Proyecto Final |
|  | | | | | | | |
| **Class Name** | | | Main | | | | |
| **Parent Class** | | | - | | | | |
|  | | |  | | | | |
|  | | |  | | | | |
|  | | |  | | | | |
|  | | | | | | | |
| **Attributes** | | | | | | | |
|  | **Declaration** | | | **Description** | | | |
|  |  | | |  | | | |
|  |  | | |  | | | |
|  | | | | | | | |
| **Items** | | | | | | | |
|  | **Declaration** | | | **Description** | | | |
|  |  | | |  | | | |
|  |  | | |  | | | |
|  |  | | |  | | | |
|  |  | | |  | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Class Name** | | ManejadorDeMemoria | |
| **Parent Class** | | - | |
|  | | | |
| **Attributes** | | | |
|  | **Declaration** | | **Description** |
|  | HashMap< TablaDePaginacion > hsmTablasDePaginacion | | Guarda las tablas de paginación de los procesos. El índice del HashMap indica el ID del proceso. |
|  | MarcoDePagina[] mdpMemoria | | Representa los marcos de página en memoria RAM. |
|  | int iTamMemoria | | Tamaño de la memoria RAM. |
|  | MarcoDePagina[] mdpSwap | | Representa los marcos de página en memoria SWAP. |
|  | int iTamSwap | | Tamaño de la memoria SWAP. |
|  | int iTamPagina | | Tamaño de ambos, página y marco de página, del sistema. |
|  | | | |
| **Items** | | | |
|  | **Declaration** | | **Description** |
|  | + ManejadorDeMemoria( int iTamMemoria, int iTamSwap, int iTamPagina ) | | Inicializa los arreglos de objetos MarcoDePagina de memoria y de swap con los tamaños especificados, e inicializa el atributo iTamPagina. |
|  | + bool cargarProceso( int iIDProceso, int iTam ) | | Carga el proceso a Memoria y regresa true si tuvo éxito. Carga el proceso agregando una TablaDePaginacion a hsmTablasDePaginacion en la posicion iIDProceso.  Si el proceso ya existía, o su tamaño es mayor que iTamMemoria, regresa falso. |
|  | + bool accesarProceso( int iDireccion, int iIDProceso, bool bModificar ) | | Traduce la dirección del proceso iIDProceso en hsmTablasDePaginacion.  Regresa verdadero si se pudo hacer corréctamente. |
|  | + CajaTemporal liberarProceso( int iIDProceso ) | | Regresa un objeto que se creará en el método. El objeto contendrá dos vectores: vecMemo, que guarda las posiciones liberadas en memoria; y vecSwap, que guarda las posiciones liberadas en Swap. (Se utilizarán para desplegar en pantalla).  Utiliza los métodos de la tabla de paginación, del proceso iIDProceso, liberarMemoria y liberarSwap para conseguir una lista de todas las residencias en memoria y en swap de iIDProceso.Con estas listas, liberará los espacios correspondientes en memoria y swap. Tras terminar esto, remueve la TablaDePaginacion asignada a iIDProceso del hashmap hsmTablasDePaginacion. |

|  |  |  |  |
| --- | --- | --- | --- |
| **Class Name** | | TablaDePaginacion | |
| **Parent Class** | | - | |
|  | |  | |
|  | |  | |
|  | |  | |
|  | | | |
| **Attributes** | | | |
|  | **Declaration** | | **Description** |
|  | Vector <Pagina> vecPaginas | | Guarda la dirección en memoria de la página [n] del vector. |
|  | int iIDProceso | | ID del proceso al cual pertenece la tabla de paginación |
|  | | | |
| **Items** | | | |
|  | **Declaration** | | **Description** |
|  | + TablaDePaginacion( int iIDProceso, int iNumPaginas ) | | Inicializa iIDProceso, y el vector de Páginas. |
|  | + Vector <int> liberarMemoria() | | Regresa un vector de enteros que representa las direcciones en memoria a liberar.. |
|  | + Vector <int> liberarSwap() | | Regresa un vector de enteros que representa las direcciones en Swap a liberar. |

|  |  |  |  |
| --- | --- | --- | --- |
| **Class Name** | | Pagina | |
| **Parent Class** | | - | |
|  | | | |
| **Attributes** | | | |
|  | **Declaration** | | **Description** |
|  | int iDirFisica | | Dirección de la memoria física donde se encuentra esta página |
|  | bool bSwap | | Si esta página se encuentra en swap |
|  | int iDirSwap | | Dirección de swap donde se encuentra esta página |
|  | bool bModificacion | | Si esta página fue modificada después de ser leida del disco |
|  | bool bModificación | | Si la página ha sido modificada |
|  | | | |
| **Items** | | | |
|  | **Declaration** | | **Description** |
|  | + Pagina(int iDirMem) | | Inicializa la Página |
|  | + swapOut(int iDirSwap) | | Cambia bSwap a true y guarda la dirección dada en su iDirSwap |
|  | + swapIn(int iDirMem) | | Cambia bSwap a false y guarda la dirección dada en su iDirMem |
|  | + modificar() | | Cambiar el bit de modificación a true |

|  |  |  |  |
| --- | --- | --- | --- |
| **Class Name** | | MarcoDePagina | |
| **Parent Class** | | - | |
|  | | | |
| **Attributes** | | | |
|  | **Declaration** | | **Description** |
|  | int iIDProceso | | ID del proceso al cual pertenece la página que está cargada en este marco |
|  | int iAccesos | | Cuantas veces se ha accedido a la página que está en este marco de página |
|  | int timestamp | | El timestamp de cuando se cargó la página en este marco (para FIFO, en caso de empate por accesos) |
|  | int iNumDePagEnTabla | | Número de página de la tabla del proceso de la página cargada en este marco |
|  |  | |  |
|  |  | |  |
|  | | | |
| **Items** | | | |
|  | **Declaration** | | **Description** |
|  | MarcoDePagina() | | Inicializa el Marco de Página |
|  | + accesso() | | Incrementa el iAccesos por 1 |
|  | + cargar(int iIDProceso, int iNumDePagEnTabla) | | Carga un nuevo marco |