

Universidad Simon Bolivar.

Organización del Computador.

Jesus Wahrman 15-11540, Cesar Rosario 15-11295.

Informe proyecto 1

Manejador de memoria:

El manejador de memoria funciona usando una lista llamada freeList, esta lista representa el estado del bloque de memoria y ocupa 4 veces menos bytes que el bloque, ya que cada byte de freeList representa 4 bytes del bloque de memoria a manejar. Los elementos de freeList tienen 3 posibles valores, 0 si esa posición no está siendo usada, 1 si está siendo usada y 2 si está siendo usada y además es el inicio de un bloque de memoria solicitado por malloc.

Init es la primera de las funciones del manejador, esta función recibe una cantidad de bytes a manejar y luego solicita el espacio para el bloque de memoria y para la freeList haciendo syscall 9 y llena la freeList de ceros para marcar todo el espacio como disponible.

Malloc recibe una cantidad de bytes para reservar, luego de esto revisa en la freeList si está el espacio disponible para reservar, si hay espacio suficiente, coloca un 2 en la primera casilla que vaya a usar de freeList y sigue colocando 1 después hasta que tenga el espacio pedido completo y luego devuelve la dirección del bloque de memoria solicitado. Si no hay espacio, devuelve un código de error.

Free recibe una dirección de memoria de un bloque solicitado con malloc y lo libera para que se pueda usar, se colocó el 2 al principio de cada espacio de memoria usado en la freeList para poder distinguir bloques de memoria que estuvieran continuos en freeList, así freeList primero libera el 2 del bloque y luego libera todos los 1 que encuentre después hasta encontrar un 2 o 0. Si se le da una dirección de memoria que no es de un bloque solicitado por malloc devuelve un código de error.

Error recibe un código de error e imprime un mensaje que explica ese código.

Lista enlazada:

La lista enlazada consta de una cabecera y cada nodo que se inserte a la misma. La cabecera está conformada por 4 atributos: first, last, size, id. Cada atributo mencionado ocupa una palabra de memoria desde la primera dirección de la lista. Los nodos, ocuparán dos palabras de memoria consecutivas cada vez que se inserten.

Create es la función encargada de inicializar la lista. Establece los valores iniciales de los atributos mencionados. Esto lo consigue haciendo una llamada a init, y otra a malloc. Se reservaron 400000 bytes para la lista, para que hubiese espacio suficiente para hacer todas las pruebas, pero no tanto como para usar demasiada memoria y que no se puedan hacer otras cosas en ella mientras este la lista. El atributo adicional "id" se utiliza para reconocer, en cada llamada a una función que maneje la lista, que la dirección dada es la de una lista válida. Para identificar a listas vacías, los atributos "first" y "last" se inicializan en -1.

Insert se ocupa de introducir un nodo a la lista cada vez que es llamada. Recibe la dirección de una lista y de un elemento. Para insertar, primero se solicita el espacio y se crea el nodo. Luego, se actualiza el tamaño y finalmente se actualiza el "next" del último nodo y se actualiza el atributo "last".

Delete se encarga de liberar el espacio correspondiente al nodo que está en una posición dada. Recibe la dirección de una lista, y la posición del nodo a eliminar. En condiciones normales, devuelve 0. Para eliminar, se busca el nodo en dicha posición, y si existe, se libera el espacio correspondiente al mismo. En caso contrario, devuelve un estatus de error.

Print es la función para imprimir a los elementos almacenados en la lista y que sólo imprime enteros. Recibirá en esta implementación solamente la dirección de la cabecera de la lista, esto porque no logramos usar la dirección de otra función para hacer print dentro de la misma. Se utilizaron macros para definir a la función. Un primer macro debe recibir la dirección de la cabecera de la lista, y en este se llama a otro macro, que imprime a un elemento por vez. En caso de que la lista esté vacía se imprime. " ".

Tabla de códigos de error

Código de error.	Mensaje correspondiente al código.
-1	No hay suficiente espacio para hacer malloc con el espacio pedido.
-2	La dirección dada no es el inicio de un bloque de memoria dado por malloc.
-3	La dirección dada al insert no es la de una lista válida.
-4	El nodo no puede ser insertado porque no hay suficiente espacio.
-5	La dirección dada para eliminar en delete no referencia a una lista.
-6	La posición dada no referencia a un nodo.