Shihang Zhang

####Big-O Analysis

##SLCreate O(1)
This creates an empty doubly linked list.
the initial status is that there is one base node in the empty list and both the next and prev pointers of base node point to itself.

##SLDestroy O(n)
free all the allocated space including the nodes and list

##SLInsert O(n)
find the position to insert that meets the rule that all of the right items smaller or equal to it.

##SLRemove O(n)
This frees the node indicated by argument passed in the function unless it is pointed by iterators. Only free the node when there is only one iterator points to it and the program is going to destroy that iterator. This will help the function SLNextItem to handle a special situation that a sorted list encapsulated within an iterator is modified while that iterator is active.

##SLCreateIterator O(1)
This creates an iterator that allows the caller to iterate through the list. It starts the iterator at the first element.

##SLDestroyIterator O(1)
This will not only destroy the iterator but also the node it points to when this node is removed from the list and only referenced by this iterator.

##SLGetItem O(1)
Get the element.

##SLNextItem O(1)
Get the next element if the node the iterator points to is still in the list, otherwise, return NULL