01 /\*@ modifies this.theArray[\*], this.topOfStack, this.theArray[this.topOfStack], this.theArray[this.topOfStack-1]; \*/  
02 /\*@ ensures (\result != null) == (\old(this.topOfStack) >= 0); \*/  
03 /\*@ ensures (\result != null) ==> (\old(this.theArray[this.topOfStack]) != null); \*/  
04 /\*@ ensures (\result != null) ==> (!(\forall int i; (0 <= i && i <= \old(this.theArray.length-1)) ==> (\old(\typeof(this.theArray[i])) != \typeof(\result)))); \*/  
05 /\*@ ensures (\result != null) ==> (\typeof(this.theArray) != \typeof(\result)); \*/  
06 /\*@ ensures (\result != null) ==> (this.theArray[\old(this.topOfStack)] == null); \*/  
07 /\*@ ensures (\result != null) ==> (this.topOfStack - \old(this.topOfStack) + 1 == 0); \*/  
08 /\*@ ensures (\result != null) ==> (this.topOfStack < this.theArray.length-1); \*/  
09 /\*@ ensures (\result == null) == (\old(this.topOfStack) == -1); \*/  
10 /\*@ ensures (\result == null) == (this.topOfStack == \old(this.topOfStack)); \*/  
11 /\*@ ensures (\result == null) ==> ((\forall int i; (0 <= i && i <= \old(this.theArray.length-1)) ==> (\old(this.theArray[i]) == null))); \*/  
12 /\*@ ensures (\result == null) ==> ((\forall int i; (0 <= i && i <= \old(this.theArray.length-1)) ==> (\old(\typeof(this.theArray[i])) == \typeof(null)))); \*/  
13 /\*@ ensures (\result == null) ==> ((\forall int i; (0 <= i && i <= this.theArray.length-1) ==> (this.theArray[i] == null))); \*/  
14 /\*@ ensures (\result == null) ==> ((\forall int i; (0 <= i && i <= this.theArray.length-1) ==> (\typeof(this.theArray[i]) == \typeof(null)))); \*/  
15 /\*@ ensures (\result == null) ==> (this.topOfStack == -1); \*/  
16 /\*@ ensures \typeof(this.theArray) != \typeof(\result); \*/  
17 /\*@ ensures this.topOfStack <= \old(this.topOfStack); \*/  
18 /\*@ ensures !(\forall int i; (0 <= i && i <= \old(this.theArray.length-1)) ==> (\old(\typeof(this.theArray[i])) != \typeof(\result))); \*/  
19 /\*@ ensures \old(this.topOfStack) <= this.theArray.length-1; \*/  
20 /\*\*  
21 \* Return and remove most recently inserted item from the stack.  
22 \* @return most recently inserted item, or null, if stack is empty.  
23 \*/  
24 public Object topAndPop( )  
25 {  
26 if( isEmpty( ) )  
27 return null;  
28 Object topItem = top( );  
29 theArray[ topOfStack-- ] = null;  
30 return topItem;  
31 }