CS2223
HW#6 SOLUTIONS

1. m :=0;

```
do m++; while ((T[m]<=T[m+1]) && (m<n-1));   /* Find m */
if m <n -1
    MERGE(T[0..m],T[m+1..n-1 ],T[1..n-1])
```

2. (*a*) The stack can contain n-1 elements, where n is the number of elements to be *quicksort*ed.
(*b*) Always process (*quicksort*) the shorter list while pushing the longer list onto the stack.

3. S := $(x_1, x_2, \ldots, x_n)$

```
if !BlackBox (S, q) return ("failure")                       O(1)
for (i = 1; i<= n; i++)                                      n times
   S = S - x_i
     if !BlackBox (S, q)          /*we really need x_i ; put it back */
                     S = S + x_i;
return S ;                                                   O (1)
```

Since the body of the loop is executed in $O(1)$ time, the time to execute the loop (and the program) is $O(n)$.