

5.4 Recursive Algorithms

5.4 pg 370 # 3

Trace Algorithm 3 when it finds $\text{gcd}(8,13)$. That is, show all the steps used by Algorithm 3 to find $\text{gcd}(8,13)$.

Algorithm 3 1 $\text{gcd}(a, b : \text{nonnegative integers with } a < b)$

```
1: if  $a = 0$  then  
2:   return  $b$   
3: else  
4:   return  $\text{gcd}(b \bmod a, a)$   
5: end if  
   {output is  $\text{gcd}(a, b)$ }
```

5.4 pg 370 # 7

Give a recursive algorithm for computing nx whenever n is a positive integer and x is an integer, using just addition.

5.4 pg 370 # 9

Give a recursive algorithm for finding the sum of the first n odd positive integers.

5.4 pg 370 # 11

Give a recursive algorithm for finding the minimum of a finite set of integers, making use of the fact that the minimum of n integers is the smaller of the last integer in the list and the minimum of the first $n - 1$ integers in the list.

5.4 pg 371 # 45

Use a merge sort to sort $b, d, a, f, g, h, z, p, o, k$ into alphabetic order. Show all the steps used by the algorithm

Procedure 2 $\text{mergesort}(L = a_1, \dots, a_n)$

```
1: if  $n > 1$  then  
2:    $m := \lceil n/2 \rceil$   
3:    $L_1 := a_1, a_2, \dots, a_m$   
4:    $L_2 := a_{m+1}, a_{m+2}, \dots, a_n$   
5:    $L := \text{merge}(\text{mergesort}(L_1), \text{mergesort}(L_2))$   
6: end if  
   { $L$  is now sorted into elements in nondecreasing order}
```
