Homework XII

1. [20 points]
Find a termination measure (i.e., an expression involving program variables that is never negative, but decreases each time either of the loop bodies in the program is executed) for the program fragment below -- explain your answer.

```
        {N≥8}
   J:= 0;  K:= N/5;
   while J*3+K*5 <> N do
      K:= 0;  J:= J+1;
      while J*3+K*5 < N do K:= K+1 end while
   end while
{N=J*3+K*5 □ J≥0 □ K≥0}
```

2. [45 points] corrected
Write a program fragment using the Wren language augmented with arrays (as per the class Web page). This program should set program variable L to the length of a longest "up-sequence" beginning at B[J] of the integer array B[1..N] (and of course, not change the array). An *up-sequence* starting at B[J] is a series of contiguous values, B[J]≤B[J+1]≤B[J+2]≤ … For instance, if B[1]=6, B[2]=7, B[3]=7, B[4]=4, B[5]=8 (i.e., N=5), B has longest up-sequence at index 1 of length 3, and longest up-sequence at index 4 of length 2. Note that there is always an up-sequence of length at least one.

Hence the pre-condition is 1≤J≤N, and the post-condition is J+L-1≤N □ 0≤n<L-1 B[J+n]≤B[J+n+1] □ (J+L-1=N B[J+L-1]>B[J+L]). Supply a proof of partial correctness for your program, including an explanation of your loop invariant and its reflection of your solution strategy. Note that since no assignments to subscripted variables need be used, the standard proof rules (i.e., Chap. 11 of our text) are sound and sufficient for this proof.

3 . [25 points]
Consider the straight-line (Wren+arrays) code segment below using only assignment statements with simple and subscripted variables. Assuming that A and B are declared as integer arrays and that all the index values are within bounds (the pre-condition), determine the value assigned to the program variable N by the execution of P, and show that it does not depend on the prior values of the variables. Give a formal axiomatic proof of your answer using the array proof rules provided in the class handout.

```
I:= J;
K:= L;
A[J] := J;
B[K]:= A[I];
M:= B[L];
N:= A[M]-B[K]
```