

4. The purpose of the algorithms in this question is to swap the values of array b and to store a value in k so that the postcondition given below is true. Array b is not necessarily sorted initially. Besides the postcondition, we give three different invariants; write a loop (with initialization) for each one. Before you begin, write the precondition, postcondition, and invariant as pictures.

Precondition Q: $b[0..] = ?$ —i.e. nothing is known about the values in b .

Postcondition R: $b[0..k] \leq 6$ and $b[k+1..] > 6$

(a) invariant P1: $b[0..h] \leq 6$ and $b[k+1..] > 6$

(b) invariant P2: $b[0..k] \leq 6$ and $b[t..] > 6$

(c) invariant P3: $b[0..s-1] \leq 6$ and $b[k+1..] > 6$

5. Below is the precondition and postcondition for the partition algorithm. Below that are three different invariants. Develop the partition algorithm (which uses only swap operations) using each of the three invariants. Before you start, write all assertions as pictures. This algorithm manipulates an array segment $b[h..k]$.

Precondition: $b[h] = x$ for some x AND (this is just so we can talk about what is in $b[h]$ initially;
 $h \leq k < b.length$ x is not a program variable.)

Postcondition: $b[h..j-1] \leq x = b[j] \leq b[j+1..k]$

(a) invariant P1: $b[h..j-1] \leq x = b[j] \leq b[t..k]$

(b) invariant P2: $b[h..j-1] \leq x = b[j] \leq b[q+1..k]$

(c) invariant P3: $b[h..j-1] \leq x = b[j] \leq b[j+1..n-1]$

6. Write selection sort, to sort array b , where $b.length \geq 1$, in several ways, using the invariants provided below. Before you do each one, write the invariant as a picture. Then write the statement(s) you use to maintain the invariant in the repetend in English —state WHAT it is to do, not HOW it is to do it. We do NOT want to see a nested loop (a loop within the repetend of your first loop).

postcondition: $b[0..b.length-1]$ is sorted (in ascending order)

(a) invariant P1: $b[0..k-1]$ is sorted and $b[0..k-1] \leq b[k..]$

(b) invariant P2: $b[0..h]$ is sorted and $b[0..h] \leq b[h+1..]$

(c) invariant P3: $b[s+1..b.length-1]$ is sorted and $b[0..s] \leq b[s+1..]$