

Homework 5

Nick Alvarez

Q1/

We want to choose partitions that cause worst-case performance. To do this, we must partition so that the search checks every element

$[3, 2, 9, 0, 7, 5, 4, 8, 6, 1]$
 $[2, 9, 0, 7, 5, 4, 8, 6, 1]$
 $[2, 9, 0, 7, 5, 4, 8, 6, 1]$
 $[2, 9, 0, 7, 5, 4, 8, 6, 1]$
 $[2, 9, 0, 7, 5, 4, 8, 6, 1]$
 \vdots
 $[2, 9, 0, 7, 5, 4, 8, 6, 1]$
 $i=k \ \& \ p=r$

Q2/

See problem2.cpp

Q3/

See problem3.cpp

Q4/

This theoretical algorithm is simply an implementation of counting sort. All elements are placed into the "C" array which would be elements between a and b.