COMP 3711 Design and Analysis of Algorithms
Spring 2015 Midterm Exam Solutions

Problem 1 (35 pts)

1.1 (10 pts)

(d)(e)(c)(a)(b)

1.2 (3 pts)

No. The $\Omega(n \log n)$ sorting lower bound holds for the worse case. Insertion sorting runs in $O(n)$ time only in the best case.

1.3 (6 pts)

(a) Counting sort; $O(n)$;
(b) Radix sort; $O(n)$;
(c) Quicksort, merge sort, or heap sort; $O(n \log n)$;

1.4 (16 pts)

(a) $\Theta(n^2)$;
(b) $\Theta(n^{\log_3 4})$;
(c) $\Theta(n)$;
(d) $\Theta(\log n)$;

Problem 2 (20 pts)

(a)

Algorithm 1 Find-k(A, p, q)

```
m ← ⌊(p+q)/2⌋
if A[m+1] < A[m] then
  return m
end if
if A[m] ≥ A[1] then
  return Find-k(A, m, n)
else
  return Find-k(A, 1, m−1)
end if
```

The depth of recursion is $O(\log n)$ and it takes constant time for each recursion, so total cost is $O(\log n)$.

(b) Find-k and BinarySearch both take $O(\log n)$ time, so the total cost is $O(\log n)$.

Algorithm 2 Find-x(A, x)

```
k ← Find-k(A, 1, n)
if x ≥ A[1] then
  return BinarySearch(A, 1, k, x)
else
  return BinarySearch(A, k + 1, n, x)
end if
```

Problem 3 (10 pts)

10,9,7,4,8,5,2,3,1,6

Problem 4 (10 pts)

The worst-case running time is $O(n^2)$. This happens with there are $n/2$ strings with length 1 and 1 string with length $n/2$. Then radix sort takes $n/2$ iterations, where each iteration takes $\Theta(n/2)$ time, so the total time is $\Theta((n/2)^2) = \Theta(n^2)$.

Problem 5 (10 pts)

Algorithm 3 RotateLeftLeft(A,B,P)

```
P.left ← B;
A.left ← B.right;
B.right ← A;
A.size ← A.left.size + A.right.size + 1;
B.size ← B.left.size + B.right.size + 1;
```

Problem 6 (15 pts)

(a) The i-th element is not thrown away iff all previous $i-1$ elements are hashed to a location other than $A[h(x_i)]$, which happens with probability $(\frac{n-1}{n})^{i-1}$. Thus, the probability that it is thrown away is $1 - (\frac{n-1}{n})^{i-1}$.

(b) Define the indicator random variable

$$X_i = \begin{cases} 1, & \text{the } i\text{-th element is thrown away;} \\ 0, & \text{the } i\text{-th element is not thrown away.} \end{cases}$$

Then,

$$\begin{aligned} E[\text{number of elements thrown away}] &= E\left[\sum_i X_i\right] \\ &= \sum_i E[X_i] \qquad \text{(linearity of expectation)} \\ &= \sum_{i=1}^{n} \left(1 - \left(\frac{n-1}{n}\right)^{i-1}\right) \\ &= n - \frac{1 - (\frac{n-1}{n})^n}{1 - \frac{n-1}{n}} = n\left(\frac{n-1}{n}\right)^n \end{aligned}$$