CS 5633 Analysis of Algorithms – Spring 04

2/4/04

4. Homework

Due **2/11/04** before class

As usual, justify all your answers, or you may lose points.

1. **Randomized code snippets (6 points)**
   Analyze the expected runtimes of the following code snippets. Clearly define your random variable. *Hint: Define a separate random variable for each iteration of the loop. And remember, random variables are functions.*

   a) (3 points)
   `RandomBit()` takes $O(1)$ time and returns 0 or 1 each with probability 1/2.

```
for(i=1; i<=n; i++){
  if(RandomBit()==1){
    for(j=i; j<=n; j++){
      print(``hello'');
    }
  }
}
```

   b) (3 points)
   `RandomInteger(i)` takes $O(1)$ time and returns an integer between 1 and i, each with probability $1/i$.

```
for(i=1; i<=n; i++){
  if(RandomInteger(i)==i){
    for(j=1; j<=i; j++){
      print(``hello'');
    }
  }
}
```

2. **Sorting (8 points)**

   a) (2 point)
   What is the runtime of merge sort for an array of n elements sorted in ...
   - ... increasing order?
   - ... decreasing order?

   b) (2 point)
   What is the runtime of deterministic quicksort (pivot = 1st element) for an array of n elements sorted in ...
   - ... increasing order?
   - ... decreasing order?

FLIP OVER TO BACK PAGE ⟹

c) (1 point)
What is the expected runtime of randomized quicksort (pivot = random element) for an array of n elements sorted in ...

- ... increasing order?
- ... decreasing order?

d) (1 point)
What is the runtime of deterministic quicksort (pivot = 1st element) for an array that contains a random permutation of n numbers?

e) (2 point)
What is the runtime of heapsort (using a max-heap) for an array of n elements sorted in ...

- ... increasing order?
- ... decreasing order?

3. **Minimum element (3 points)**

How can you compute the minimum element in a max-heap of n elements? How much time does it take in the worst case?

4. **Heaps with links (9 points)**

Suppose that binary max-heaps are represented using explicit links, that means in a standard binary tree representation that uses nodes with pointers/references to left and right children. Consider the problem of merging the binary max-heap L with the binary max-heap R. Assume both heaps are complete trees containing $2^l - 1$ and $2^r - 1$ nodes, respectively, and let $n = \max\{2^l - 1, 2^r - 1\}$.

- **a) (3 point)**
  Give an $O(\log n)$ algorithm to merge the two heaps if $l = r$.
- **b) (3 point)**
  Give an $O(\log n)$ algorithm to merge the two heaps if $|l - r| = 1$.
- **c) (3 point)**
  Give an $O(\log^2 n)$ algorithm to merge the two heaps regardless of l and r.

5. **d-ary min-heaps (4 points)**

How would you represent a d-ary min-heap in an array? Please give formulas for the children and for the parent of a node. What is the height of such a heap?