Homework 2

22C:44 Algorithms, Fall semester 2000

Four problems, ten points each. Due in class on Tuesday, Sept. 12.

1 Solve the following recurrences, that is, find a function $g(n)$ such that $T(n)$ is $\Theta(g(n))$. In each case we assume that $T(n)$ is positive for all small values of n.

(a) (2 points) $T(n) = 2T(n/5) + \sqrt{n}$.

(b) (2 points) $T(n) = T(n/2) + 1/n$.

(c) (3 points) $T(n) = 2T(n-1) + 3T(n-2)$. [Hint: Guess $T(n) = \Theta(3^n)$ and substitute to show that your guess is correct.]

(d) (3 points) $T(n) = T(\sqrt{n}) + 1$.

2 Analyze the asymptotic worst-case time complexities of the following algorithms. In other words, find a function $g(n)$ such that $T(n) = \Theta(g(n))$ where $T(n)$ is the worst-case time complexity over input lists containing n elements. In (b) and (c) write down a recurrence for $T(n)$.

(a) (2 points)

```
Funny(A[1...n])
for i ⟵ 1 to n
   for j ⟵ i to n do A[j] = A[i]+A[j]
```

(b) (4 points)

```
RecursiveMax(A[1...n])
if n = 1 then return A[1] else
a ⟵ RecursiveMax(A[2...n])
if a > A[1] then return a else return A[1]
```

(c) (4 points)

```
DivideAndConquerMax(A[1...n])
if n = 1 then return A[1] else
a ⟵ DivideAndConquerMax(A[1...⌊n/2⌋])
b ⟵ DivideAndConquerMax(A[⌊n/2⌋+1...n])
if a > b then return a else return b
```

3 Consider the following algorithm:

```
Mystery(A[1...n])
maxsum ⟵ A[1]
for i ⟵ 1 to n do
  for j ⟵ i to n do
    begin
      sum ⟵ 0
      for k ⟵ i to j do
        sum ⟵ sum + A[k]
      if sum > maxsum then maxsum ⟵ sum
    end
return maxsum
```

(a) (2 points) What does the algorithm do, I mean, what is the interpretation of *maxsum* it returns ?

(b) (3 points) Analyze the asymptotic worst-case running time $T(n)$ of `Mystery`, where n is the number of elements in the input array. In other words, find a function $f(n)$ such that $T(n)$ is $\Theta(f(n))$.

(c) (5 points) Write an equivalent algorithm (i.e. an algorithm that returns the same value as `Mystery` on all inputs) that runs in linear $\Theta(n)$ time.

4a (2 points) Alice an Bob flip a coin to determine who gets the last beer. If the coin comes up heads then Alice wins, if it comes up tails then Alice will want best out of three, that is, the coin is flipped twice more and Alice wins if it comes up heads both times. Calculate the overall odds of Alice winning.

4b (4 points) Alice flips a fair coin 100 times and Bob flips it 101 times. What is the probability that Bob gets more heads than Alice?

4c (4 points) Suppose we have two coins. One is fair and gives heads and tails with equal 0.5 probability. The other coin is unbalanced and produces heads with unknown probability $p \neq 0.5$, and tails with probability $1 - p$. Let us choose one of the two coins randomly, with equal probabilities, and flip it twice. If the first flip comes up heads and the second one tails, is it more likely that the coin is the fair one or the unfair one? Prove using Bayes's theorem.