CS341 Midterm Marking Scheme

March 2, 2011

1 Question 1

a)

$a = 9, b = 3, \log_b a = 2$

$f(n) = \Theta(n^{\log_b a})$ -case 2 of the Master Theorem

Answer: $\Theta(n^2 \log n)$

b)

$a = 10, b = 3, \log_b a = \log_3 10 > 2$

$f(n) = O(n^{\log_b a - \epsilon})$ for $\epsilon = 0.001$ - case 1

Answer: $\Theta(n^{\log_3 10})$

c)

$a = 8, b = 3, \log_b = \log_3 8 < 2$

$f(n) = \Omega(n^{\log_b a + \epsilon})$ for $\epsilon = 0.001$.

We also have that $af(n/b) < cf(n)$ for $c = 0.9$ - case 3

Answer: $\Theta(n^2)$

Marking scheme: 3 marks for each question (1 mark for the correct case of the Master Theorem +2 marks for the correct answer) + 1 mark for checking the extra condition in c). 1 mark was deducted for confusing $O(.)$ with $\Theta(.)$.

2 Question 2

a) no
b) yes
c) yes
d) no
e) yes
f) yes
g) yes
h) no

i) yes
j) no

Marking scheme: 1 mark for each question

3 Question 3

a)

Algorithm:

1. $P = (a + b) \cdot (c + d)$ (1st multiplication)
2. $Q = a \cdot c$ (2nd multiplication)
3. $R = b \cdot d$ (3rd multiplication)
4. Output $Qx^2 + (P - Q - R)x + R$

b)

We divide each of the two polynomials into two parts and multiply parts similarly as in part a).

Pseudocode:

$multiply((a_0, ..., a_n), (b_0, ..., b_n))$:
if $n < 2$
multiply polynomials using the basic algorithm
endif
$(p_0, ..., p_n)$=multiply($(a_0+a_{n/2+1}, a_1+a_{n/2+2}, ..., a_{n/2}+a_n), (b_0+b_{n/2+1}, b_1+b_{n/2+2}, ..., b_{n/2}+b_n)$)
$(q_0, ..., q_n)$=multiply($(a_0, ..., a_{n/2}), (b_0, ..., b_{n/2})$)
$(r_0, ..., r_n)$=multiply($(a_{n/2+1}, ..., a_n), (b_{n/2+1}, ..., b_n)$)
for $i = 0..n/2$
$s_i = q_i$
for $i = n/2 + 1..n$
$s_i = q_i + (p_{i-n/2} - q_{i-n/2} - r_{i-n/2})$
for $i = n + 1..2n$
$s_i = r_{i-n} + (p_{i-n/2} - q_{i-n/2} - r_{i-n/2})$
return $(s_0, ..., s_{2n})$

Marking scheme: 4 marks for a), 6 marks for b). Up to 2 marks deducted for problems with the "conquer" phase in b). No marks given for solutions with more than 3 multiplications.

4 Question 4

Algorithm:

1. Sort the boxes according to their base area
2. H(0)=0
3. Fill the dynamic programming table according to the recursion:

$$H(j) = max_{k<j:w(k)\leq w(j)\wedge d(k)\leq d(j)} H(k) + h(j)$$

Marking scheme: 10 marks total. 3 marks - any DP formulation 4 marks - some good idea 6 marks - only consider area instead of comparing both width and height 9 marks - only count the boxes instead of summing height 10 marks - correct formulation

5 Question 5

Algorithm:

1. Sort numbers in B in ascending order
2. Let b be the first number in B
3. Find an interval (a_{begin}, a_{end}) in A s.t. $a_{begin} < b$ and a_{end} is maximized
4. Remove from B all numbers smaller or equal to a_{end}
5. Repeat steps 2-5 until B is empty

Proof:

Let $(g_1, ...g_k)$ be the intervals in the solution found by the greedy algorithm (sorted according to their right endpoints). Suppose that there exists an optimal solution $(o_1, ..., o_m)$ with $m < k$. The first number in B must be contained in g_1. Also, g_1 must contain all the numbers contained by o_1 since the choice of g_1 is greedy. Therefore $(g_1, o_2, ..., o_m)$ is also an optimal solution. Similarly, among the elements of B not contained in g_1, g_2 contains all the elements contained by o_2, so $(g_1, g_2, o_3, ..., o_m)$ is also an optimal solution. By repeating this argument, we can see that $(g_1, ..., g_m)$ is also an optimal solution. But then the algorithm terminates after m steps, which contradicts the assumption that $m < k$.

Marking scheme:

4 for the algorithm,6 for the proof. For the algorithm, 3 marks were deducted for wrong choices of the next interval (e.g. choosing the interval based on its total length). 1 mark was deducted for minor errors.

For the proof, 4 marks were deducted for trying to prove that $(g_1, ..., g_k) = (o_1, ..., o_m)$. 1 or 2 marks were deducted for lack of clarity or minor errors. Completely wrong proofs were given 0 marks.