CS 6212 – Mid Term

Name:
Score: **/ 60**

(60 points) (120 minutes)

Q1-6 5 t pts each, write answer only

Q1 (5 pts): Solve the following recurrence: $T(n) = 6\,T(n/2) + n^2$.

Q2, 3, 4 (5 points each): What is the time complexity of these algorithms, in terms of n? ***[Sum += y is a short form notation for Sum = Sum + y.]*** ***[Only write answers, NO explanations]***

Answer

`int j = 2`
`while (j < n) {`
`  int k = j`
`  while (k < n) {`
`  Sum += a[j]*b[k]`
`  k = k * k`
`  }`
`  j = sqrt(2) * j`
`}`
	`for (int i = 1 to n) {`
	`  for (int j = i to n) {`
	`  for (int k = j to n) {`
	`  Sum += a[i]*b[j]*c[k]`
	`  }`
	`  If (j == 2 * i) {`
	`  j = n`
	`  }`
	`  }`
	`}`
		`for (int j = 1 to n) {`
		`  int k = j`
		`  while (k < n) {`
		`  Sum += a[k]*b[k]`
		`  k += sqrt n`
		`  }`
		`}`

Q5 (5 pts): Compare recurrence relations R and S. $R(n) = R(n/2) + n$. $S(n) = S(2n/3) + \log n$.

Q6 (5 pts): Estimate this recurrence relation. $T(n) = T(n/2) + T(n/3) + \sqrt{n}$

Q7 (10 points): **Finding n-th power in log n time**

Describe an algorithm for function power(integer a, integer n) that computes a^n, in *O(log n)* time.

Q8 (10 points): **Rotten teeth**

Dentist Smilefacher is very popular in Hollywood. She routinely receives timeslots from celebrities who want to come in on a certain day. An example request could be: Tom: [8 AM-2 PM], Meg: [9 AM-10 AM], Denzel: [11 AM-4PM], Britney: [5PM-520PM]. Being rich, all celebrities pay her a flat $5000 fee, irrespective of how long they stay. Considering they are celebrities, you cannot have more than one in the office at the same time (they bite). Also, considering they are celebrities, they only come in as per their own schedule. Give a polynomial time algorithm to maximize the revenue for Dr. Smilefacher.

Q9 (10 points): **Longest common subsequence**

Given two strings (sequences of characters), the longest common subsequence (LCS) problem is to find the longest subsequence (not necessarily contiguous) that exists in both of the input strings. For example, given strings "mangoes and oranges" and "mementos nor pores", the subsequence "mnos n ores" is common in both and is the longest common subsequence. Given two strings of sizes n_1 and n_2 respectively, find a dynamic programming algorithm to find the longest common subsequence in $O(n_1 n_2)$ time.