

# KING SAUD UNIVERSITY

COLLEGE OF COMPUTER & INFORMATION SCIENCES  
DEPT OF COMPUTER SCIENCE

CSC311 Computer Algorithms

Second Semester 1428/1429 AH

Second Mid-term Examination: Sun 13.05.1429 A.H./18.05.2008 C.E. (duration = 2 hours)

Instructor:

Dr. Aqil Azmi

---

---

**1. [Marks 30=10+10+10]**

Solve the following recurrence relations, assume that  $T(2) = C$ .

- $T(n) = 4T(n/2) + n^3$ , using repeated substitution.
- Do, using master table.
- $T(n) = T(\sqrt{n}) + 1$ .

**2. [Marks 20]**

Draw the binary decision tree for the standard binary search for the case  $n = 18$ . Compute the exact average number of comparisons needed for the *successful* and the *unsuccessful* searches.

**3. [Marks 20=7+9+4]**

Suppose we wrote a modified binary search which splits the search area into two partitions  $\alpha : 1 - \alpha$  where  $0 < \alpha < 1$ . For example, if  $\alpha = \frac{1}{3}$  then we split into  $\frac{1}{3} : \frac{2}{3}$ . **(a)** Write the formula for computing the split point  $m$ . **(b)** What is the height (*i.e.* # of levels) of the binary decision tree for  $n$  elements. For simplicity you may assume that  $\alpha \geq \frac{1}{2}$ . **(c)** Compute the height of the binary decision tree for  $n = 100$  when  $\alpha = \frac{3}{4}$ . **Hint:** for part (b) write the recurrence relation for the worst case, *i.e.* the case where we always end up picking the larger partition.

**4. [Marks 30=15+10+5]**

Given a list  $A[ ]$  of *unordered* elements and two numbers  $x, y$ . Devise a divide & conquer based algorithm  $\text{Count}()$  which returns the tuple  $(c_1, c_2)$  where  $c_1$  is the number of elements in list  $A$  which are *less* than  $x$ , and  $c_2$  which is the number of elements in list that are *greater* than  $y$ . For example, if input list is  $[5, 3, 2, 10, 9, 16, 4, 5, 11]$  and  $x = 10, y = 7$  then the procedure returns  $(6, 4)$ . **(a)** Write the pseudocode for  $\text{Count}()$ ; **(b)** Write the recurrence relation for the code you wrote in (a); and **(c)** What is the complexity of this code?