School of Information Technology
IIT Kharagpur

Course Id: IT60101 Foundations of Computing Systems   Mid-Semester Examination

Date: September 15, 2006   Total Time: 2 Hours (Part A + Part B)

Answer Part A and Part B in separate answer scripts

PART A

Answer all the questions

1. Answer the following questions. **(6x3=18)**

a. What are the maximum and minimum numbers that can be represented by 12-bit 2's complement representation? What decimal values do the 12-bit 2's complement numbers 111111111111 and 0111111100 represent?

b. What are user and supervisory modes in a processor? How does the mode typically change from user to supervisory, and vice versa?

c. Suggest a method of providing memory protection in contiguous memory allocation scheme.

d. For a processor, state two principal advantages of microprogrammed control unit over hardwired control unit. Which of the two is used in most present-day processors, and why?

e. What are *dirty bit* and *valid bit* in the context of cache memory?

f. In Verilog, if three register type variables X, Y and Z have values 10, 15 and 30 respectively, what will be their new values after the following block of code is executed?

```
always  @ (posedge clock)
begin
   Y <= X + 10;
   X <= X + 2;
   Z <= X – Y;
end;
```

2. **(8+8=16)**

a. Draw the schematic diagram of a processor organization using three internal buses, showing the important control signals. The processor should contain standard registers like PC, IR, MDR, MAR, and sixteen 32-bit registers R0..R15 organized as a register file.

b. For the above processor, write the complete micro-operations for the instructions:

```
LOAD        R1, R5 (XYZ)        // R1 <= M [R5 + XYZ]
ADD         R2, R3              //  R2 <= R2 + R3
```

Make relevant assumptions.

3. **(8+8=16)**

a. Consider a 3-level memory hierarchy with access times t_{A1}, t_{A2} and t_{A3} respectively, and hit ratios in the first two levels as H_1 and H_2 respectively. Let t_{B1} and t_{B2} respectively denote the block transfer times between levels 1-2 and 2-3. Derive an expression for the average access time of the memory hierarchy. Empirically extend the result to an n-level hierarchy (no derivation required).

b. Consider a 2-level memory hierarchy consisting of cache memory and main memory, which uses set-associative mapping. The size of the main memory is 1 Gbytes, size of the cache memory is 512 Kbytes, block size is 256 bytes, and the number of cache blocks per set is 8. Clearly state how a logical address generated by the processor gets mapped to a cache block. What hardware resources are needed to implement this mapping?

PART B – Data Structures and Algorithms Max. Marks: 50

Instructions: Answer all questions. You may answer the questions in any order. However, all parts of the same question must be answered together. Clearly state any reasonable assumption you make.

1. Master theorem for solving recurrence relations may be stated as follows:

   Let $a \geq 1$ and $b > 1$ be constants, let $f(n)$ be a function, and let $T(n)$ be defined on the nonnegative integers by the recurrence $T(n) = aT(n/b) + f(n)$. Then, $T(n)$ can be bounded asymptotically as follows:

   (i) If $f(n) = O(n^{\log_b a-\varepsilon})$ for some constant $\varepsilon > 0$, then $T(n) = \theta(n^{\log_b a})$

   (ii) If $f(n) = \theta(n^{\log_b a} \lg^k n)$, then $T(n) = \theta(n^{\log_b a} \lg^{k+1} n)$

   (iii) If $f(n) = \Omega(n^{\log_b a+\varepsilon})$ for some constant $\varepsilon > 0$, and if $af(n/b) \leq cf(n)$ for some constant $c < 1$ and all sufficiently large n, then $T(n) = \theta(f(n))$

Using the above-stated Master Theorem, solve the following recurrence relations.

(a) $T(n) = 4T(n/2) + n$
(b) $T(n) = 9T(n/3) + n^{2.5}$
(c) $T(n) = 2T(n^{1/2}) + lgn$

Hint: For problem (c), you may want to use a new variable for substituting lgn (not be confused with substitution method)

[4×3=12]

2. Prove that

(a) $f(n) = \theta(g(n))$ if and only if $g(n) = \theta(f(n))$
(b) If $f(n) = O(g(n))$ and $g(n) = O(h(n))$ then $f(n) = O(h(n))$

[4+4=8]

3.

(a) Write an algorithm for the PARTITION (A, p, r) procedure of Quicksort, where A is the input array, p and r are the leftmost and rightmost positions of A within which the partition has to be performed.

(b) Diagrammatically show the effect of running PARTITION (A, 1, 9) on the array A = < 4, 19, 6, 5, 2, 9, 10, 1, 7 > for each iteration of the main loop. You do not have to show complete sorting of the whole array.

Assume that the index of array A runs from 1 to 9.

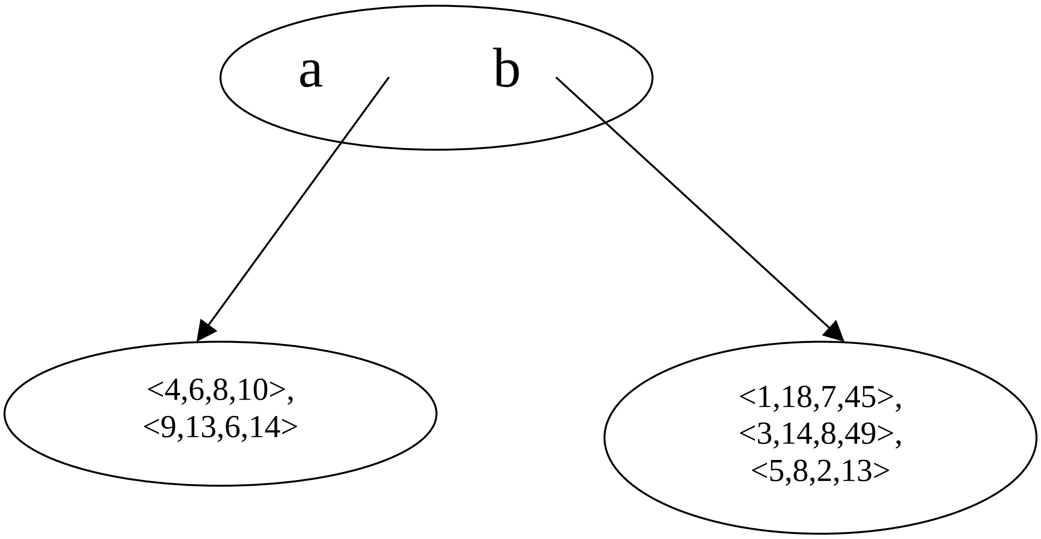
[3+5=8]

4.

A series of nodes with the following key values are inserted in an empty Red-Black Tree: 4, 7, 12, 15, 3. Show the Red-Black Tree after the end of each insert.

[16]

5. Consider that an R-Tree has been constructed as shown below. The left leaf node contains two entries and the right leaf node contains three entries. What are the values of the keys *a* and *b* in the root node? Do not worry about the child/tuple pointers. Clearly state any reasonable assumption that you make. **[6]**


a
b
<4,6,8,10>,
<9,13,6,14>
<1,18,7,45>,
<3,14,8,49>,
<5,8,2,13>