JJ

PhD Comprehensive Examination

Department of Computer Science and Engineering

Date: 17-Sept-2012 **Time:** 4 Hours **Maximum Marks:** 100

[*Answer any five questions from Group A, and any five questions from Groups B and C combined.*]

Group A

A.1 (a) For solving a problem, there are two algorithms, namely A1 and A2. The time-complexity of A1 is always $T(n) = T(n/2) + n$, and that of A2 is always $T(n) = 2T(n/2) + n/2$. Explain which one is better for a sufficiently large input.

(7)

(b) Given a set S containing only positive integers in the interval $[1, n]$, suggest a data structure that can store the elements of S in such a manner that insertion and deletion of an element from S can be done in constant time. hash table

(5 + 5 = 10)

A.2 Describe an algorithm to construct a min-heap from a binary search tree. You should try to reduce the time complexity of your algorithm. Explain the worst-case time complexity of your algorithm.

(8 + 2 = 10)

A.3 Given an integer k and an unordered set A with n integers, suggest an algorithm to prepare a subset B of A such that the sum of elements of B is at least k and the number of elements of B is minimum. You should try to reduce the time complexity of your algorithm. Explain the worst-case time complexity of your algorithm.

(8 + 2 = 10)

minimum spanning tree

A.4 Given a weighted, undirected, connected graph $G(V, E)$ with positive weight function w, suggest an algorithm to construct a subgraph $G'(V', E')$, where $V' \subset V, E' \subset E$, such that:

i) $|V'| = \lceil V/2 \rceil$

ii) G' is connected and acyclic

iii) the sum of weights of edges in E' is minimum among all such subgraphs

Explain its worst-case time complexity.

(8 + 2 = 10)

A.5 Suppose you have a linked list, with the data field of each node of the linked list containing an integer.

(10)

(a) Declare a C structure called **LListNode** to describe a single node of the linked list.

(b) Write a C function **void RemoveDuplicate (LListNode *head)** to remove nodes having duplicate numbers in the linked list, where **head** is a pointer to the first node of the linked list. For example, if the linked list contains $10 \rightarrow 20 \rightarrow 25 \rightarrow 20 \rightarrow 15 \rightarrow 10$, after processing by **RemoveDuplicate()**, it should contain $10 \rightarrow 20 \rightarrow 25 \rightarrow 15$.

(3 + 7 = 10)

A.6 Consider the following recursive definition of the *Fibonacci Sequence*: $f(0) = 0$, $f(1) = 1$ and $f(n) = f(n-1) + f(n-2)$ for $n \geq 2$. The sequence of (recursive) function calls occurring in a program to calculate the n-th Fibonacci number can be represented by a "recursion tree", which is a binary tree where each node represents a function call $f(i)$, and the two children of the node represent the function calls $f(i-1)$ and $f(i-2)$.

In this question, given an integer n, you will have to construct and traverse a recursion tree for computing $f(n)$ that will store the integer argument i at the node corresponding to $f(i)$. An example recursion tree to calculate $f(4)$ is shown below in Fig. 1.

prev ptr prev ptr

10 → 10 → 15 → 20 → 20 → 25

prev=ptr;

ptr prev prev ptr

10 → ~~10~~ → 15 → 20 → 20 → 25

~~10 → 20 → 20~~

(a) Declare a C structure called **TreeNode** to describe a node of the recursion tree.

(b) Write a C function **TreeNode* ConstructTree (int n)** to construct the recursion tree representing the evaluation of $f(n)$. The function should return a pointer to the root node of the recursion tree.

(6)

(c) Write a C function **long int NumNodes (TreeNode *root)** to estimate the total number of function calls that needs to be done to calculate $f(n)$, **root** being a pointer to the root node of the corresponding recursion tree.

(2 + 5 + 3 = 10)

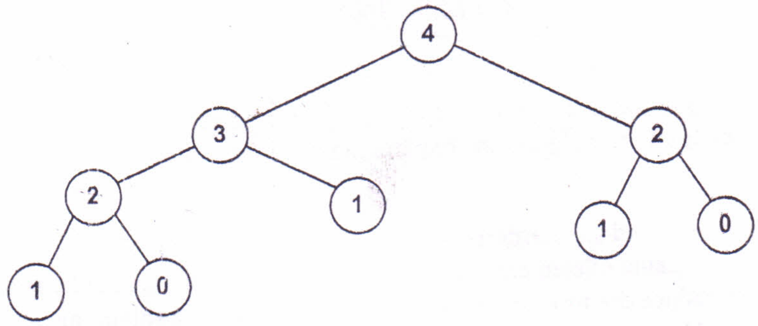

Figure 1: Binary tree to be created for representing recursion tree of $f(4)$

A.7 Consider the problem of multiplying a $n \times n$ matrix of integers with a length-n column vector of integers. Each column of the matrix is stored in a linked list with n nodes, with each node storing one integer. The column vector to be multiplied is also similarly stored in a linked list. The product column vector will also be stored in a linked list. Let the C structure **Number** describe a single node of such linked lists. Write a C function **void MatVecMul (Number *Cols[], Number *Vec, Number *Result)** to perform the matrix-vector multiplication. Here, **Cols** is an array of pointers, where each pointer points to one column of the matrix; **Vec** is a pointer to the column vector to be multiplied, and **Result** is a pointer to the product column vector. Assume sufficient space has been pre-allocated in the calling function for the result vector. **(10)**

Group B

B.1 (a) Prove that given a nonnegative integer n, there is a unique nonnegative integer m such that $m^2 \leq n < (m+1)^2$

(b) The Fibonacci sequence is as follows:

$$\begin{aligned} F(0) &= 0 \\ F(1) &= 1 \\ F(n) &= F(n-1) + F(n-2) \text{ for } n > 1 \end{aligned}$$

② /

Prove by induction that for all $n \geq 1$, it holds that $F(3n)$ is an even number.

(5 +5 = 10)

B.2 (a) For the following function $f : \mathcal{R} \to \mathcal{R}$, determine whether f is injective (one-to-one) and/or surjective (onto). Briefly justify your answers.

$$f(x) = (x-1)(x-2)(x-3)(x-4)$$

(b) Let $A = \{1,2,3,4\} \times \{1,2,3,4,6\}$. A relation on A is defined by $(x,y)R(s,t)$ if $\frac{x}{t} = \frac{y}{s}$. Show that f is an equivalence relation.

(5+5 = 10)

B.3 (a) Suppose there is a $n \times n$ square grid where $(0,0)$ is the lower left corner, and (n,n) is the upper right corner. Let C_n be the number of routes from $(0,0)$ to (n,n) where we are restricted to travelling only to the right or upward, and we are allowed to touch but not go above a diagonal line from the lower-left corner to the upper-right corner.

(a) Derive a recurrence relation for C_n

(b) Find a formula for C_n

(b) Suppose we have a biased coin that comes up heads with probability p and tails with probability $1-p$. Suppose we toss the coin n times.

(a) Find the probability of at least one tail.

(b) Find the probability of at most one tail.

(c) Find the probability of at least two tails.

(5 + 5 = 10)

B.4 (a) Let L_4 denote the set of all strings w over the alphabet $\{a,b,c\}$ such that w does <u>not</u> contain the substring ac. Design a finite automaton (deterministic or nondeterministic) to accept L_4.

(b) Convert the automaton of Part (a) to a regular expression.

(5 + 5 = 10)

5 B.5 (a) Prove that regular languages are closed under concatenation. (The concatenation of two regular languages L_1 and L_2 is defined as $\{w_1w_2 \mid w_1 \in L_1, w_2 \in L_2\}$.)

(b) It is given that the language $\{a^nb^n \mid n \geq 0\}$ is not regular. Using Part (a) prove that the language $L_5 = \{a^nb^{n+1} \mid n \geq 0\}$ is not regular. (Do not use the pumping lemma. You may assume that regular languages are closed under union, intersection, and complement.)

(5 + 5 = 10)

$F(4n) = F($

F

$F(3n) = F(3n-1) + F(3n-2)$

$= F(2) + F(1)$

$= 1 - 0 \times 1 = 1$

B.6 (a) Let L_6 denote the set of all strings over the alphabet $\{a, b\}$, that are <u>not</u> of the form ww^R for some $w \in \{a, b\}^*$ (where w^R denotes the reverse of w). Design a context-free grammar to accept L_6.

(b) Design a pushdown automaton to accept L_6.

(5 + 5 = 10)

Group C

C.1 (a) Suppose a process wants to read a file from the disk. List clearly the steps that occur from the time the process initiates the read to the time the process gets the data back and uses it. (You do not have to write any steps that is related to file management, like accessing the file descriptior table, finding the file on the disk etc. Assume that the file is found magically by the file and disk subsystem given the filename).

(b) Name six values that will be stored in the process control block of a process.

(7 + 3 = 10)

C.2 (a) What is the use of a TLB?

(b) Suppose that a process accesses a virtual memory address that is not in physical memory. List clearly step-by-step what happens from the time the process issues the address (starting with how it is detected that the content is not in memory) till the time the process uses the content of the address.

(2 + 8 = 10)

C.3 A barber shop has a barber, a barber chair, and a waiting room with 5 chairs. When a barber finishes cutting a customer's hair, the barber fetches another customer from the waiting room if there is a customer, or stands by the barber chair and daydreams if the waiting room is empty. A customer who needs a haircut enters the waiting room. If the waiting room is full, the customer goes away and comes back later. If the barber is busy but there is a waiting room chair available, the customer takes a seat. If the waiting room is empty and the barber is daydreaming, the customer sits in the barber chair and wakes the barber up.

Think of the barber and customers as processes which should be synchronized. You are required to write the code for the barber process and the customer process using semaphores. Your answer should first define what semaphores you have to use (including their initial value), and then write the pseudo code of the two processes. You can assume that wait and signal are available as primitive calls on a semaphore with their usual meanings. (10)

C.4 (a) Consider the IEEE floating point representation consisting of the exponent e and the significand s. Define gap as the difference between two successive numbers in a representation. For a 32-bit floating point number, evaluate the gap and compare it with that of a fixed point representation where there are 16 bits for the integer part, and 16 bits for the decimal part.

(b) Prove the following statement: In the 2's complement number system an unsigned binary integer is a power of 2 if and only if the bitwise logical AND of x and $x - 1$ is 0.

(5 + 5 = 10)

C.5 (a) Consider two compilers producing machine code for a given program to be run on the same machine. The machine's instructions are divided into class A (CPI=1) and class B (CPI=2). Machine language programs produced by the compilers lead to the execution of the following number of instructions from each class:

Class	Instructions for compiler 1	Instructions for compiler 2	Comments
A	600M	400M	CPI=1
B	400M	400M	CPI=2

(a) What are the execution time of the two programs, assuming a 1 GHz clock?

(b) Which compiler's machine language output runs at a higher MIPS rate?

(c) Make a one line comment on *MIPS as a performance metric* based on the above problem.

(b) A computer system has L1 and L2 caches. The local hit rates for L1 and L2 are 95% and 80%, respectively. The miss penalties are 8 and 60 cycles, respectively. Assuming a CPI of 1.2 without any cache miss and an average of 1.1 memory accesses per instruction, what is the effective CPI after cache misses are considered? Assume when data is not found in cache, it must be first brought into the cache and then accessed from the cache.

(5 + 5 = 10)

C.6 The sum of two 32-bit integers may not be representable in 32 bits. In this case we say that an overflow has occurred. Write a sequence of micro-MIPS instructions (see table below), that adds two numbers stored in the registers $\$s1$ and $\$s2$, stores the sum (modulo 2^{32}) in register $\$s3$, and sets register $\$t0$ to 1 if overflow occurs and to 0 otherwise. **(10)**

Class	Instruction	Usage	Meaning	op	fn
Copy	Load upper immediate	lui rt, imm	$rt \leftarrow (imm, 0x0000)$	15	
Arithmetic	Add	add rd,rs,rt	$rd \leftarrow (rs) + (rt)$	0	32
	Subtract	sub rd,rs,rt	$rd \leftarrow (rs) - (rt)$	0	34
	Set less than	slt rd,rs,rt	$rd \leftarrow$ if $(rs) < (rd)$ then 1 else 0	0	42
	Add Immediate	addi rt,rs,imm	$rt \leftarrow (rs) + imm$	8	
	Set Less than immediate	slti rt,rs,imm	$rt \leftarrow if(rs) < imm$then 1 else 0	10	
Logic	AND	and rd,rs,rt	$rd \leftarrow (rs) \wedge (rt)$	0	36
	OR	or rd,rs,rt	$rd \leftarrow (rs) \vee (rt)$	0	37
	XOR	xor rd,rs,rt	$rd \leftarrow (rs) \oplus (rt)$	0	38
	NOR	nor rd,rs,rt	$rd \leftarrow ((rs) \vee (rt))'$	0	39
	AND immediate	andi rt,rs,imm	$rt \leftarrow (rs) \wedge imm$	12	
	OR immediate	ori rt,rs,imm	$rt \leftarrow (rs) \vee imm$	13	
	XOR immediate	xori rt,rs,imm	$rt \leftarrow (rs) \oplus imm$	14	
Memory Word	Load Word	lw rt,imm(rs)	$rt \leftarrow mem[(rs) + imm]$	35	
	Store Word	sw rt,imm,(rs)	$mem[(rs) + imm] \leftarrow (rt)$	43	
Control transfer	Jump	j L	goto L	2	
	Jump register	jr rs	goto (rs)	0	8
	Branch on less than 0	bltz rs,L	if$(rs) < 0$ then goto L	1	
	Branch on equal	beq rs,rt,L	if $(rs) = (rt)$ then goto L	4	
	Branch on not equal	bne rs,rt,L	if$(rs) \neq (rt)$ then goto L	5	
	Jump and link	jal L	goto L; 31 $\leftarrow$ (PC)+4	3	
	System call	syscall	Associated with an OS system routine	0	12