00062

THE OPEN UNIVERSITY OF SRI LANKA
BACHELOR OF SOFTWARE ENGINEERING - LEVEL 5
ECX5263 COMPUTER ORGANIZATION AND OPERATING SYSTEMS
FINAL EXAMINATION 2015
DURATION: THREE HOURS

DATE : 01st December 2016 **TIME : 0930 - 1230 HOURS**

Answer any ***five*** questions. All questions carry equal marks.

1.

(i) Add the following numbers as fixed-point integers. Your calculations must be shown by using binary numbers in two's complement. Choose appropriate word length.

(a) 6.625 + (-11.25)

(b) -5.125 + 0.25

(ii) Represent all the above numbers as floating point numbers in the following format using biased exponent.

S (1 bit)	Exponent (7-bit)	Mantissa (24-bit)

(iii) In a source code a programmer declared an array of integers. All elements of the array are signed 16-bit integers. Later the programmer declared another array of floating point numbers at the same starting pointer of the integer array. Assume all the floating pointer number elements are in the above format as in question 1.(ii) and values are saved in the integer array.

(a) Draw a diagram to show how these two arrays are stored in the memory. Assume that the memory is byte addressable.

(b) Explain when you get positive numbers in the floating point array.

(c) In order to get values greater than 1 in all elements of the floating point array which numbers do you have to save in the integer array.

(d) What will be decimal values of the elements of the floating point array if you save the following values in the integer array in the given order: 1408, 00, -96 and -32768 in decimal.

2. The Fig. 2.1 shows a linear pipeline with k stages (S_i) separated by latches (L). Common clock (C) is applied to all latches simultaneously. Each stage S_i has a time delay τ_i, where $i=1,2,...,k$.

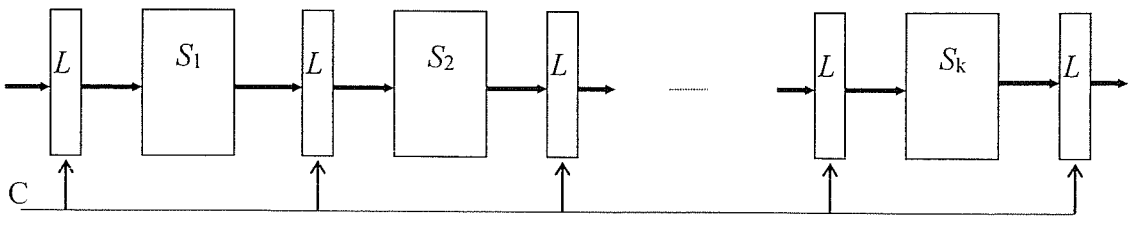

Fig. 2.1 Linear pipeline with k stages.

(i) If the time delay of the each latch is τ_l what is the clock period (τ) of the pipeline?
(ii) Get an equation to calculate the speedup (s) of the k stage linear pipeline over an equivalent nonpipeline processor.
(iii) Show that maximum achievable speedup of a linear pipeline is equal to its number of stages.
(iv) Calculate the maximum speedup of a linear pipeline with 4 stages, where $\tau_1 = \tau_3 = 40$ ns, $\tau_2 = 80$ ns, $\tau_4 = 50$ ns and $\tau_l = 10$ ns.
(v) Explain why the speedup you get in (2. iv) is less than 4. How can you improve the speedup in this case?

3.

(i) Compare and contrast a dynamic partition system and a relocatable dynamic partition system.
(ii) Describe how the function of the Page Map Table differs in paged vs. segmented/demand paging memory allocation.
(iii) Given that main memory is composed of three page frames for public use and that a seven-page program (with pages a, b, c, d, e, f, g) requests pages in the following order:

a, b, a, c, d, a, e, f, g, c, b, g

(a) Using the following page replacement algorithms, show how pages are allocated in page frames indicating page faults. Then compute the number of page faults.
a) First-In First-Out (FIFO)
b) Least Recently Used (LRU)
(b) Increase the size of memory so it contains four page frames for public use. Using the same page requests as above and page replacement algorithms, compute the number of page faults for each algorithm.

4.

(i) Distinguish MIMD multiprocessors from multi-computers or computer networks.
(ii) What are the salient features of RISC architecture and CISC architecture? Provide advantages and disadvantages of each architecture.
(iii) Explain the Amdahl's law and derive an equation for that.
(iv) A processor executes 10 million floating point and one million overhead instructions in 50 ms. What is the MFLOPS rating of this processor? What would be the MIPS rating of the same processor?

5.

(i) Draw typical virtual machine schematics for different styles of Instruction Set Architecture (ISA) i.e. Accumulator, Memory-Memory, Stack, Load-Store.
(ii) Describe advantages and disadvantages of each virtual machine in question 5. (I) comparing each other.
(iii) Write a program using the ISA (given in the Appendix) of the Accumulator architecture of the Students' Experimental Processor (SEP) to find the elements of the array X according to the following formula selecting positive values of the arrays A and B:

$X_i = A_i + B_i$; where $i = 1$ to n.

Assume that all elements of the arrays and the value n are stored in the memory.

6.

(i) Explain how you prevent the occurrence of a deadlock.

(ii) What are the objectives of scheduling of processes running in a processor?

(iii) Consider preparation for the Final Examination by a student. Final Examination for each course is held on different days. Sometimes, two courses may be held on the same day, but at different time periods. Assume that you have registered for more than four courses where the Final Examinations for 2 courses are fallen on the same day.

(a) What is/are the scheduling method/s you use for the preparation for the Final Examination? Briefly explain your answer.

(b) Draw a state transition diagram for the preparation of the Final Examination.

(c) What information you have to save in the Process Control Block in the process of preparation for the Final Examination.

7.

(i) Briefly describe the disk allocation schemes, Contiguous Allocation and Linked Allocation.

(ii) Briefly describe the RAID levels. What is/are the suitable RAID level/s for the following systems/situations?

(a) Data-critical real-time systems

(b) Transferring large quantities of non-critical data

(c) Data critical systems with high data transfer rates

(iii) A disk drive has eight surfaces, with 512 tracks per surface and a constant 64 sectors per track. Sector size is 1 Kbytes. The average seek time is 8ms, the track-to-track access time is 1.5ms, and the drive runs at 3600 rpm. Successive tracks in a cylinder can be read without head movement.

(a) What is the drive capacity?

(b) What is the average access time for the drive?

8.

(i) Describe how the state transition diagram is useful in any process description

(ii) Describe why operating systems need Process Control Box (PCB).

(iii) Given the following information:

Job #	Arrival Time	CPU Time
1	0	7
2	1	2
3	2	2
4	2	1
5	4	3

Draw a time line for each of the following scheduling algorithms and compute the waiting time and turnaround time for every job.

(a) First Come First Served

(b) Round Robin (using a time quantum of 1, ignore context switching and natural wait)

Appendix

ISA of Accumulator Architecture

Arithmetic instructions		
ADD	Addition	Acc ← Acc + op
SUB	Subtraction	Acc ← Acc - op
MUL	Signed multiplication	Acc (16 bit) ← Acc(8 LSBs) * op(8 LSBs)
DIV	Unsigned division	Acc ← Acc / op2
INC	Increment by 1	Acc ← Acc + 1
Logical instructions		
AND	Bit-wise And	Acc ← Acc AND op
OR	Bit-wise OR	Acc ← Acc OR op
XOR	Bit-wise XOR	Acc ← Acc XOR op
SHL	Shift left by 1-bit	CF ← Acc (MSB), op← Acc (14 down to 0) & 0
SHR	Shift right by 1-bit	CF ← Acc (LSB), Acc ← 0 & Acc (15 down to 1)
ROL	rotate left by 1-bit	CF ← Acc (MSB), op← Acc (14 down to 0) & CF
ROR	rotate right by 1-bit	CF ← Acc (LSB), Acc ← CF & Acc (15 down to 1)
NOT	One's compliment negation	op1 ← NOT Acc
Control Transfer instructions		
Conditional Branches		
JC	Jump if carry	If CF =1 then IP ← IP + Operand
JOF	Jump if over-flow	If OF =1 then IP ← IP + Operand
JS	Jump if Sign	If SF =1 then IP ← IP + Operand
JP	Jump if parity	If PF =1 then IP ← IP + Operand
JZ	Jump if result is zero	If ZF =1 then IP ← IP + Operand
Unconditional branch		
JUMP	Jump	IP ← IP + Operand
Loops		
LOOZ	Loop until zero	Count ← Count – 1 IF Count =0; Loop termination ELSE; IP ← IP + operand
Calls and Returns		
CALL	Procedure call	implied return address ← IP IP ← Immediate address
RETURN	Return from procedure	IP ← Contents of implied return address
Miscellaneous instructions		
NOP	No operation	
Data Movement instructions		
LOADacc	Copy the operand to the accumulator	Immediate: Acc ← op Direct: Acc ← memory (op)
STOREacc	Copies the accumulator to the memory address	Direct: Memory (op) ← Acc Indirect : Memory {memory (op)} ← Acc