THE OPEN UNIVERSITY OF SRI LANKA
BACHELOR OF SOFTWARE ENGINEERING - LEVEL 5
ECX5263 COMPUTER ORGANIZATION AND OPERATING SYSTEMS
FINAL EXAMINATION 2012
DURATION : THREE HOURS

DATE : 18th August 2013 **TIME : 0930 - 1230 HOURS**

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

1.

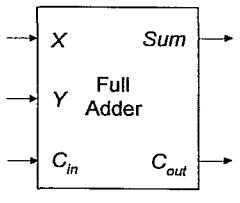
(i) Add the following numbers as fixed-point integers. Your calculations must be shown by using binary numbers in two's complement. Choose appropriate word length.
(a) -10.5 + (-0.75)

(b) -14.125 + 1.25

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

S (1 bit)	Exponent (8-bit)	Mantissa (23-bit)

(iii) Draw a truth table for a typical *Full Adder* given in the following figure, which adds two binary digits (X and Y), and a carry (C_{in}) to give a sum (*Sum*) and a carry out (C_{out}). Draw a logic circuit diagram for the *Full Adder* using standard logic gates.



2.

(i) Draw typical virtual machine schematics for Memory-Memory Architecture and briefly describe the functionality of it.

(ii) Compare and contrast Memory-Memory architecture with Load-Store architecture considering performance, size of the processor and size of the code of a program.

(iii) Write a program using the ISA (given in the Appendix) of the Accumulator architecture of the Students' Experimental Processor (SEP) to find the total (X) of an array, i.e.

$$X = \sum_{i=1}^{n} A_i$$

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

3. It is observed that in a computer system, memory operations take 40% of the execution time. Usually L1 cache speeds up 75% of the memory operations by 5 times and L2 cache can speed up only 60% of the rest of the memory operations by 3 times. Assume that sections of execution sped up by L1 and L2 caches do not overlap.

   (i) Draw a diagram to illustrate the above scenario indicating the percentages/factors of each section of execution.

   (ii) Calculate the speedup when the system has only L1 cache.

   (iii) Calculate the achievable speedup when the system has both L1 and L2 caches.

   (iv) Describe the working principle of cache memory.

4.

   (i) Distinguish between *Fixed Partition Multiprogramming* and *Variable Partition Multiprogramming*.

   (ii) Describe what the pre-emptive and non pre-emptive algorithms are. What are the advantages and disadvantages of them?

   (iii) Given the following information:

Job #	**Arrival Time**	**CPU Time**
1	0	9
2	1	2
3	2	4
4	3	1
5	4	6

   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 2**, ignore context switching and natural wait)

5.

   (i) What is meant by a page fault in virtual memory organization?

   (ii) When a page fault occurs describe what the computer system would do.

   (iii) For the virtual memory organization a hard disk is purchased for employing disk subsystem. According to the vendor the following specifications are given: 8 GB capacity, 1024 cylinders, 255 heads, 63 sectors, 7200 rpm and average seek time 9ms. The page size of the virtual memory is 8 Kbytes. Performance of the computer system is given as 100 MIPS

   (a) Calculate the number of bytes in a sector of the hard disk. (Note: bytes in a sector should be the nearest to the power of two value)
   (b) Calculate the data transfer rate of the disk subsystem.
   (c) Compute the wasted time in the event of a page fault while executing a program.

6.

(i) Briefly describe what an instruction pipeline is.

(ii) An instruction execution path will take T ns to execute an instruction in one-stage pipeline. However this logic path can be divided into any number of stages and the logic delay (T ns) may be able to subdivide equally as well. Moreover the sum of setup time of a latch and clock skew will be τ ns. Let k as the number of stages.

(a) What is the execution time of n number of instructions in one-stage pipeline?

(b) What is the clock period T_{clock} of the pipeline in terms of T, k and τ?

(c) What is the execution time of n number of instructions in the multistage pipeline (with k stages)?

(d) Derive an equation to find the minimum number of instructions (the smallest n) to have less execution time in a multistage pipeline than a one-stage pipeline.

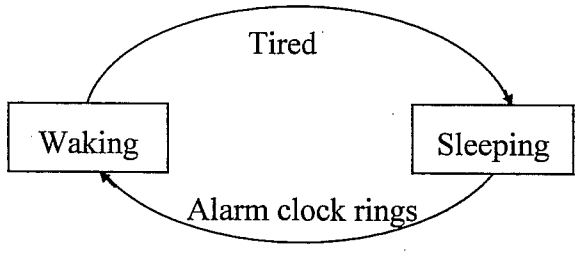
(e) Using the above equation show that execution of only one instruction in a pipeline will decrease the performance.

7.

(i) Draw the complete *process state transition* diagram and describe it for a typical multi-user operating system.

(ii) Describe how state transition diagram is useful in any process description.

(iii) The following diagram is a simplified process model of you, in which there are only two states: sleeping and waking.



You make the transition from waking to sleeping when you are tired, and from sleeping to waking when the alarm clock goes off.

(a) Add three more states to the diagram.

(b) State all of the possible transitions among the five states.

8.

(i) Describe the term *deadlock* in computer systems.

(ii) Describe four necessary conditions to exit a deadlock in computer systems.

(iii) Distinguish *page* from *segment* in virtual memory management.

(iv) What is the information of a process usually kept by an operating system? Explain how it is done by an operating system.

(v) What are the major functions of an operating system?

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