UNIVERSITY OF COLOMBO, SRI LANKA

UNIVERSITY OF COLOMBO SCHOOL OF COMPUTING

DEGREE OF BACHELOR OF INFORMATION TECHNOLOGY (EXTERNAL)

Academic Year 2016 – 3rd Year Examination – Semester 5

IT5305: Computer Systems II

Structured Question Paper

13th May, 2016

(TWO HOURS)

To be completed by the candidate

BIT Examination Index No: ....

Important Instructions:

- The duration of the paper is **2 (two) hours**.
- The medium of instruction and questions is English.
- This paper has **7 questions** and **14 pages**.
- **Answer all questions.** All questions **do not** carry equal marks.
- **Write your answers** in English using the space provided **in this question paper**.
- Do not tear off any part of this answer book.
- Under no circumstances may this book, used or unused, be removed from the Examination Hall by a candidate.
- Note that questions appear on both sides of the paper.
  If a page is not printed, please inform the supervisor immediately.

Questions Answered

Indicate by a cross (×), (e.g. ☒) the numbers of the questions answered.

	Question numbers							
To be completed by the candidate by marking a cross (×).	1	2	3	4	5	6	7	
Marks allocated for each question	16	18	13	19	06	13	15	
To be completed by the examiners:								

1) (a) A processor has a 32 bit word length and a 32 bit floating point representation format with a 10 bit exponent (E) and a 21 bit mantissa where the actual exponent is interpreted as (E-1024).

(i) What is the minimum positive real number and the maximum positive real number that the processor can represent? Do not simplify and express in terms of powers of 2.

(4 marks)

ANSWER IN THIS BOX

Typical format: <S> : <M> : <E>

1 bit 21 bits 10 bits

Minimum Positive : <0> : <.1 ← zeros (21 bits)> : <zeros (10 bits)>

Gives ; 0.5×2^{-1024}

Maximum Positive : <0> : <.1 ← one’s (21 bits)> : <one’s (10 bits)>

Gives ; $(1 - 2^{-22}) \times 2^{+1023}$

(ii) If the processor floating point registers are of length 16 bits, how can a 32 bit floating point operation be handled by the registers?

(3 marks)

ANSWER IN THIS BOX

Combine two registers to hold the 32 bit operand.

(iii) How many contiguous (one after another) memory locations are required to store a 32 bit number on this processor?

(4 marks)

ANSWER IN THIS BOX

Each physical memory location always contain 8 bits; Therefore 4 contiguous memory locations are required.

(b) A certain computing cluster has 8 homogeneous (identical) nodes, each with 8 processors. Each processor has a 1GHz clock and a CPI (cycles per instruction) of 0.3. What is the maximum processing power of the cluster if used as a parallel computer, in terms of GFLOPS? Assume that FLOPS (floating point operations per second) is identical to IPS (instructions per second).

(5 marks)

ANSWER IN THIS BOX

Each Processor IPS $= \frac{Cycles/Sec}{Cycles/Instructions}$

$$\left(\frac{1 \times 10^9}{0.3}\right) flops$$

$\therefore$ Max throughput $= 8 \times \frac{10^9 \times 10}{3}\ flops$

$$\cong 27\ Gflops$$

2) Consider the following high level code fragment. Here, the initial values of **a** and **b** are stored at memory locations 500 and 700 respectively and the value of **p** is to be stored at memory location 900. Your code should start at memory location 2000.

```
if (a > b) then
      p = a + b;
else
      p = 0;
```

(a) Assuming the processor has only one internal register R, initially zero, write down the machine instruction sequence corresponding to a Register/Memory architecture. Your code should have not more than 8 instructions. The typical instruction set for a R/M architecture is as follows: **(9 marks)**

```
LOAD M : [R is loaded from memory location M]
STORE M: [R is stored at memory location M]
ADD M: [adds R to memory content at M and saves in R]
SUB M: [subtracts R from memory content at M and saves
result in R]
COMPARE M1, M2: [compares memory contents at locations M1
and M2 and raises a flag]
JUMP_flag M: [jumps to memory address M if flag is raised]
(flags: GT - greater than 0; EQ – equal to zero; GE –
greater than or equal to zero)
JUMP M: [jumps to memory address M]
```

ANSWER IN THIS BOX

```
2000 :  COMPARE    500,700      ; a > b ?
2001 :  JUMP.AT    2004         ; yes
2002 :  STORE      900          ; p = 0
2003 :  JUMP       2007         ; end

2004 :  LOAD       500          ; a
2005 :  ADD        700          ; a + b.
2006 :  STORE      900          ; p = a + b
2007 :  END
```

(b) Assuming the processor has a RISC architecture write down the machine instruction sequence. Content of register R_0 is zero. Your code should have not more than 8 instructions. The typical instructions for a RISC architecture are:

(9 marks)

LOAD R_x, offset[R_0]: [Rx is loaded from memory location (offset+R_0)]
ADD R_z,R_y,R_x : [Add contents of Rx to Ry and store it in Rz]
SUB R_z,R_y,R_x : [subtract contents of Rx from Ry and store it in Rz]
STORE offset[R_0],R_x : [Rx is stored at memory location (offset+R_0)]
JUMP_flag M: [jumps to memory address M if flag is raised]
(flags: GT - greater than 0; EQ – equal to zero; GE – greater than or equal to zero)
JUMP M: [jumps to memory address M]

ANSWER IN THIS BOX

2000: LOAD R_1 , 500[R_0] ; $R_1 = a$

2001: LOAD R_2 , 700[R_0] ; $R_2 = b$

2002: SUB R_3 , R_2 , R_1 ; $R_3 = a - b$

2003: JUMP_GT 2006 a>b ?

2004: STORE 900[R_0] , R_0 ; P=0

2005: JUMP 2008

2006: ADD R_4 , R_2 , R_1 ; $R_4 = a + b$

2007: STORE 900[R_0], R_4 ; P = a+b

2008: END

3) Consider a simple RISC instruction pipeline.

(i) Name the five stages of the pipeline.

(4 marks)

ANSWER IN THIS BOX

IF – Instruction Fetch

ID – Instruction decode/ Register fetch

EX – Execute arithmetic / logic

MEM – read / write memory

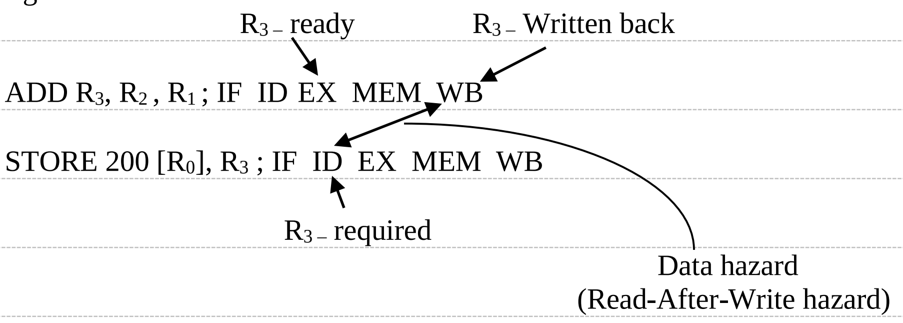
WB – Write results to register file

(ii) Explain using two example instructions, the scenario of a data hazard.

(4 marks)

ANSWER IN THIS BOX

Eg:



R_3 – ready

R_3 – Written back

ADD R_3, R_2 , R_1 ; IF ID EX MEM WB

STORE 200 [R_0], R_3 ; IF ID EX MEM WB

R_3 – required

Data hazard
(Read-After-Write hazard)

(iii) In terms of the CPI (cycles per instruction) what is the motivation for instruction pipelining?

(2 marks)

ANSWER IN THIS BOX

Instruction pipelining, if works perfectly, achieves a CPI=1. Without Instruction pipelining CPI>1. Objective is to reduce CPI below 1.

(iv) Explain one way the compiler can help by preprocessing the machine instructions, to increase the efficiency of the pipelined execution.

(3 marks)

ANSWER IN THIS BOX

Compiler can schedule independent instructions between hazard causing dependent instructions.

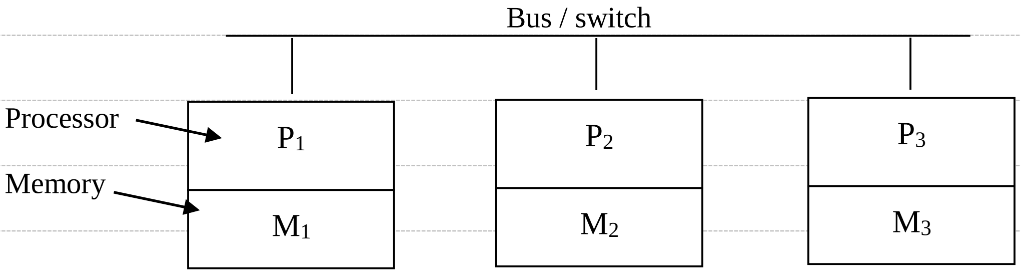
So that stalling is minimised. Loop-unrolling is another example.

4) (a) Consider a computing cluster with N nodes, each node having its own CPU and main memory.

(i) Draw a block diagram of the cluster.

(2 marks)

ANSWER IN THIS BOX



(ii) Is this a shared memory multiprocessor? Explain.

(2 marks)

ANSWER IN THIS BOX

No. This is a “distributed memory” architecture.

(iii) In the above cluster, how can a processor P1 access (to read or write) data that is produced by another processor P2?

(3 marks)

ANSWER IN THIS BOX

P_1 can't directly access P_2's memory. But P_1 can send a message (with serialized data for example). That will pick up by P_2 and then store it in P_2's memory (or, read and send back to P_1). This is the "Message passing" paradigm.

(iv) A certain computational task has fraction f (<1) inherent parallelism. The task takes T seconds to run on a single node. How long does the task take to run on the n-node cluster?

(3 marks)

ANSWER IN THIS BOX

Total Time taken = Time for serial fraction + Time for parallel fraction

$$f.T + \frac{(1-f).T}{n}$$

(b) Consider the multiplication of two symmetric matrices A and B, each of size m x m, where the product is matrix C.

(i) Express the element c_{11} as the sum of products of elements in a row of A (a_{11}, a_{12}, …, a_{1m}) and a column of B (b_{11}, b_{21}, b_{22}, b_{m1}).

(3 marks)

ANSWER IN THIS BOX

$$\underset{mxm}{[C]} = \underset{mxm}{[A]} + \underset{mxm}{[B]}$$

Rule is, multiply a row of A and a column of B , to get an element in C

Therefore;
$c_{11} = a_{11} .b_{11} + a_{12} .b_{21} + a_{13} .b_{31} + \ldots\ldots\ldots\ldots + a_{1m} .b_{m1}$

(ii) Write down a simple high level language loop instruction to show how c_{11} can be calculated.

(3 marks)

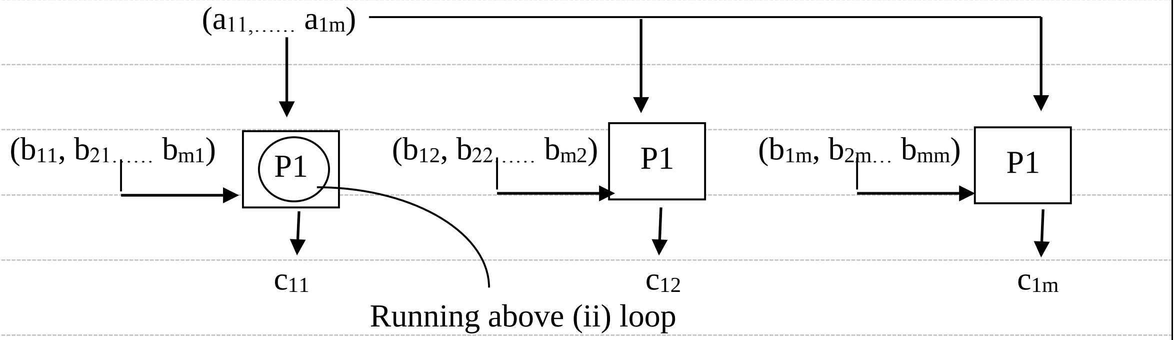
ANSWER IN THIS BOX

```
sum = 0;

 for(i=1; i≤ m; i+1 )
{

 sum = sum + a[1,i] x b[i,1] ;

}

 c[1,1] = sum ;
```

(iii) Show that, if there were m processors, all the elements in the vector (c_{11}, c_{12}, c_{13}, …, c_{1m}) could be calculated simultaneously.

(3 marks)

ANSWER IN THIS BOX



By observing that we have to multiplies the 1st row of A with 2nd colomn of B to get C_{12} etc, if we have m-processors, we can distribute each coloumn of B and first row of A to each P and get (c_{11}, c_{12} … c_{1m}) simultaneously

5) A data centre is a physically co-located group of computing servers, storage devices and other infrastructure with backups, typically presented as a uniform cloud by a service provider to its clients.

(i) To utilize the physical servers to the maximum, the service provider allocates computing resources (such as virtual machines) on demand to clients. What operating system concept can create virtual machines on demand on a given hardware platform?

(3 marks)

ANSWER IN THIS BOX

Concept is called ‘Virtualization’

(ii) Distinguish between Infrastructure as a Service (**IaaS**), Platform as a Service (**PaaS**) and Software as a Service (**SaaS**) types of cloud services giving an example in each.

(3 marks)

ANSWER IN THIS BOX

IaaS: provisioning of virtual machines, servers, storage etc.

PaaS: provisioning of execution environment, run time support, libraries etc.

SaaS: Provisioning of multi-tenant applications software for business, scientific computing, analytics etc

Continued…

6) (a) Under an operating system, "*when a user initialises several copies of the same program (task), a single process for all of them will be created*." Do you agree with that statement? Justify your answer.

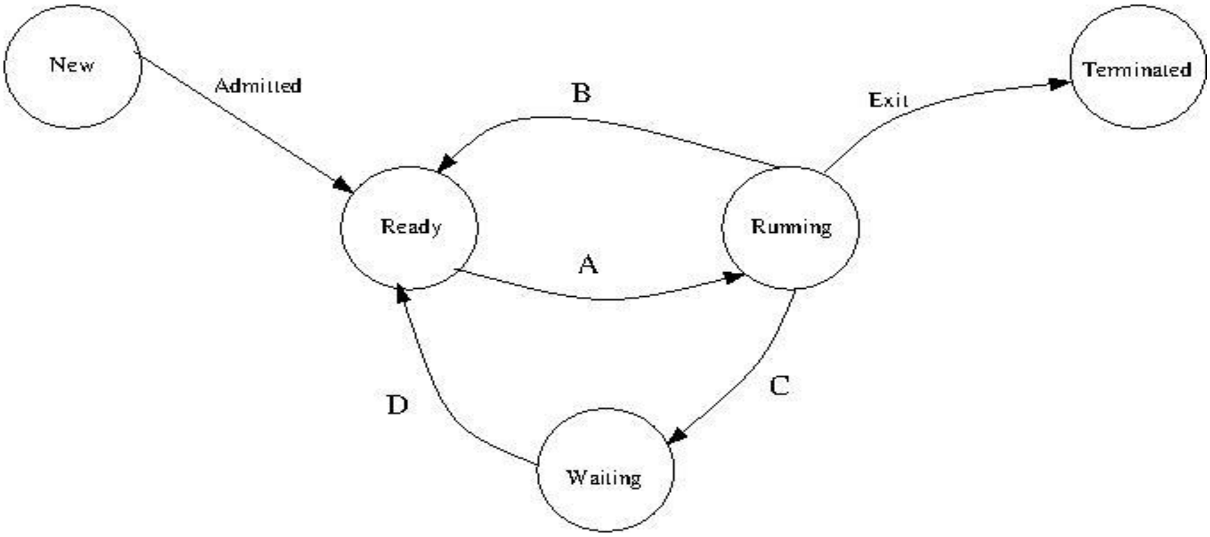
(2 marks)

ANSWER IN THIS BOX

No. When several copies of the same program are started, multiple processes will need to be created as they need to run separately.

(b) Following is the state transition diagram for the word processing process. A user starts a word processing application on a computer. After a while, he starts an Internet browser too. Explain events that cause transitions labeled as A, B, C and D.

(4 marks)



ANSWER IN THIS BOX

A - The operating system's scheduler selects the word processing process to run

B - The clock interrupt results in sending the word processing process to READY state so that another process (e.g. internet browser) could run

C - The word processing process has to wait for input/output or an event to occur

D - The I/O or event completion

(c) Most software applications that run on modern computes are multi-threaded. Explain how multi-threading will benefit a web browser application.

(4 marks)

ANSWER IN THIS BOX

Multiple threads can be used by a web browser to improve responsiveness. e.g., Different threads can be used to

- **display images**
- **display text**
- **retrieve data from a network etc.**

(d) Using an example code show how *pthreads* can be utilized for the above purpose in (c). (**Hint:**-Use the *pthread*-create function*)*

(3 marks)

ANSWER IN THIS BOX

One could create a thread to handle each concurrent task:

```
pthread_create(&tid[0], &attr, task1, argsForTask1Function);

pthread_create(&tid[0], &attr, task2, argsForTask2Function);

pthread_create(&tid[0], &attr, task3, argsForTask3Function);
```

7) (a) In a *uni-programming system*, the main memory is divided into two parts: Part X and Part Y where Part Y will be for the program currently being executed. In a multi-programming system, Part Y is sub-divided to accommodate multiple processes while Part X is reserved for the same purpose. For what type of content is Part X reserved?

(2 marks)

ANSWER IN THIS BOX

For the operating system

(b) *An executable program size will not be limited to the amount of physical memory that is available on a computer*. How is this possible? Explain.

(7 marks)

ANSWER IN THIS BOX

This is possible because the entire program need not be in memory at once. The main memory need to contain only the active pages of the program. The OS keeps track of the

Continued…

pages that are in memory and that are not and will initiate the transfer of non resident pages to memory as an when needed.

(c) *Contiguous, linked* and *indexed* are three major methods of allocating disk space on a computer. Describe each of them with their advantages and concerns (one each).

(6 marks)

ANSWER IN THIS BOX

(i) Contiguous: each file occupies a set of contiguous blocks on disk; *advantage*: speedy access; *concerns (any one of the following):* determining how much space is needed for a file, finding space for a new file, fragmentation, finding new space for growing files.

(ii) Linked: each file is a linked list of disk blocks. The directory contains pointers to the first and the last blocks of the file; *advantage:* solves all problems of contiguous allocation; *concerns (any one of the following):* can be used effectively for sequential access files. Accessing specific blocks is inefficient as they must be accessed in order, space requirement for pointers, reliability issues that arise when pointers are lost or damaged, performance may suffer as the data of the file may be spread all over a disk.

(iii) Indexed: All the pointers are stored in one location called the index block. Each file has its own index block which is an array of disk block addresses: the i'th entry in the index block points to the i'th block of the file. The directory contains the address of the index block. *Advantage*: resolves the direct access problems of the linked allocation*; concerns (any one of the following):* keeping the index block in memory may require considerable space, space requirement for pointers, performance may suffer as for linked allocation as the data blocks may be spread all over a disk.

****