Indian Institute of Technology, Kharagpur
Department of Computer Science and Engineering
Mid-Spring Semester 2019-20
COMPUTER ARCHITECTURE AND OPERATING SYSTEMS (CS31702)
Date: 20-2-2020 Session: FN (9-11am) Duration: 2 hrs Marks: 100

Note: *(i) The question paper contains **two** parts. Answer **all** questions in both Part – A and Part – B.*
(ii) Do not mix up answers of Part – A and Part – B.
(iii) Use of simple calculators is permitted.
(iv) All answers should be brief and to the point. Unnecessary lengthy answers will be penalized.
(v) No clarifications will be entertained during the examination. You may mention the assumptions while answering the questions, if required.

PART – A (50 Marks)
Operating Systems (CS31702)

(1) Answer the following: **(2+2+1+1+1+1+1+1=10)**

(i) What is a *bootstrap program*? Where is it stored in a computer? Mention its functionality.

(ii) Provide the expansion of DMA. How DMA is carried out?

(iii) Name two important data structures used in kernel software implementation.

(iv) What is the main difference between traditional and mobile computing environments?

(v) Name two popular operating systems used in mobile environments.

(vi) Mention the active and passive entities in client-server environment.

(vii) What is Peer-to-Peer computing? Name an example application related to this.

(viii) Provide a diagram to illustrate the concept of virtualization when using multiple guest operating systems.

(2) Answer the following: **(2+2+1+5=10)**

(i) Mention different types of user interfaces provided by various operating systems.

(ii) What is an API? Name two APIs and specify for which OS they are compatible?

(iii) Mention different modes of parameter passing to system calls.

(iv) Match the unix system calls with their respective tasks mentioned in the table.

System Call	Task
chown()	Process Control
pipe()	File manipulation
getpid()	Information Maintenance
close()	Communication
wait()	Protection
exit()	
read()	
sleep()	
mmap()	
chmod()	

(3) Answer the following: **(2+2+2+2+2=10)**

(i) What are the *system programs*? Give two example system programs.

(ii) What is layered architecture in the context of design of OS? Justify why layered architecture is not appropriate for designing the OS?

(iii) Discuss the salient features of Microkernel operating system structure. Mention its positive and negative aspects.

(iv) Discuss the concept of lodable kernel modules in the context of designing OS. Mention the merits of this design.

(v) When are core-dump and crash-dump files generated? What do they contain? What is the use of those files?

(4) Answer the following: **(2+2+2+2+2=10)**

(i) With appropriate diagram show the structure (various sections) of a process in main memory.

(ii) Draw the process state transition diagram. Labelall states and transitions.

(iii) Mention the various components of information maintained by process control block.

(iv) Discuss salient features of (i) short-term, (ii) long-term and (iii) medium-term process schedulers.

(v) What is context switch? Name the sequence of events (actions) that take place during context switch. How to optimize the delay involved in context switching?

(5) Answer the following: **(1+1+1+1+2+2+1+1=10)**

(i) How is a child process created?

(ii) Up-on creating a child process, what differences would you observe between the parent and the child processes?

(iii) How can the users control the execution of parent and child processes?

(iv) How can a child process transforms itself to a completely different process?

(v) What is a Zombie process and how are these handled?

(vi) What are the orphan processes and how are these handled?

(vii) How is the child process termination notified to the parent?

(viii) What do you mean by cascaded process termination?

---------- **END OF PART – A** ----------

PART – B (50 Marks)
Computer Architecture (CS31702)

(1) Identify one factor that makes RISC processors faster than CISC processors and one factor that makes RISC processors slower than CISC processors. Restrict your answer to two short bullet points only. **[4]**

(2) Give one advantage and one disadvantage of doubling the number of registers in the ISA of a processor. **[4]**

(3) In a certain program, 20% of the instructions are load/stores. A certain RISC processor took 20 seconds to execute the program. If the processor data cache is increased so that the average time required for a load/store operation is reduced by 30%, what will be the new execution time of the program? **[5]**

(4) The frequencies of two processors A and B are 2.5GHz and 2.0GHz respectively. The CPIs of these two processor are 2.5 and 2 respectively. Which processor would have a better MIPs rating? **[5]**

(5) Assume that the following MIPS assembly instructions are executed. Copy the following table to your answer book. In your table, write down the registers whose contents change as execution of the program proceeds and also their changed values. The first column has been filled in for you. **[5]**

```
        addi   $t0, $R0, 2
        addi   $t1, $R0, 5
        slt    $t2, $t1, $t0
        beq    $t2, $zero, skip
        addi   $t1, $t2, 3
skip:
        addi   $v0, 42
```

Register	$t0				
Value	2				

(6) The MIPS assembly code given below is a translation of a C function named **flathead.** Reconstruct the C function **flathead** from the given MIPS code. Your code should be as concise as possible and should not use any goto statements or explicit pointers. Then, using one sentence describe what the function does. **[10+3]**

```
flathead:
            addi    $t0, $a2, 1
loop:
            bge     $t0, $a1, exit
            mul     $t1, $t0, 4
            add     $t1, $t1, $a0
            lw      $t2, 0($t1)
            sub     $t1, $t1, 4
            sw      $t2, 0($t1)
            addi    $t0, $t0, 1
            j       loop
exit:
            jr      $ra
```

(7) Implement the following C function using MIPS assembly code. **[14]**

```
int minimum(int V[], int n){
        int min, i;
        min = V[0];
        for (i=1;i<n; i++)
                if (V[i] < min)  min = V[i];
        return  min;
}
```

---------- END OF PART – B ----------