MANIPAL UNIVERSITY JAIPUR
SCHOOL OF COMPUTING AND IT
IV Semester B. Tech. - Second Sessional Examination- 2017-18
Branch: CSE / IT /CCE
CS1401- Operating Systems
(OPEN BOOK)

Duration: 1 hour **Max. Marks: 15**

Instructions:

- All questions are compulsory.
- Missing data, if any, may be assumed suitably.
- One Book and one notebook (handwritten notes) is allowed.

1. Consider the following snapshot of a system with five processes P_0 through P_4 and three resource types A, B and C. Resource type A, B and C has initially 12, 7 and 5 instances respectively.

Process	Allocation			Max		
	A	B	C	A	B	C
P_0	2	1	1	3	3	4
P_1	1	2	0	1	3	2
P_2	3	1	1	6	2	1
P_3	2	2	2	2	5	4
P_4	1	0	0	1	3	1

Answer the following questions using Bankers Algorithm:

a) What are the contents of the matrix NEED? Is the system in a safe state?

b) If a request from process P_3 arrives for (0, 0, 1), can the requested resources be granted? If No, Why? [4]

2. On a system using paging memory management technique with 2^{38} pages of logical address space, page size of 512 bytes and 2^{32} number of page frames

a) What is the minimum space required for the page table?

b) What amount of physical memory will be wasted if the process size is 1048700 bytes, due to internal fragmentation?

c) What is the size of the physical address space? [3]

3. Write a program in 'C' to find the value of the following expression using two pthreads :

result= (sum of squares of integers from 1 to n) / n! , where *'n'* is an integer

The first thread finds the sum of squares of integers from 1 to n and the second thread computes factorial of n. The main thread then computes the value of result using the above formula and displays it. [4]

4. a) A shared variable *x*, initialized to zero, is operated on by four concurrent processes *W, X, Y, Z* as follows. Each of the processes *W* and *X* reads *x* from memory, increments by one, stores it to memory, and then terminates. Each of the processes *Y* and *Z* reads *x* from memory, decrements by two, stores it to memory, and then terminates. Each process before reading *x* invokes the P operation (i.e., wait) on a counting semaphore *S* and invokes the V operation (i.e., signal) on the semaphore S after storing *x* to memory. Semaphore *S* is initialized to two. What is the maximum possible value of *x* after all processes complete execution? Justify your answer with proper explanation. [2]

b) Whether there is a possibility of race condition(s) in the 'C' program given below? If yes, Use semaphores to avoid the race condition(s).

```
#include <pthread.h>
int x=4;
void main()      {
     pthread_t tid;
    pthread_create (&tid,NULL,f1,NULL);
     for (int i=0; I <10000000; i++)
          x=x+30;
     pthread_join ( tid, NULL);
     printf(" Final value of x is %d \n", x);      }
void * f1(void *param)      {
       for (int i=0; I <10000000; i++)
         x=x+20;
       pthread_exit(0);              }
```

[2]