Roll Number: 21234757061

Master of Computer Applications
MCAC403: Advanced Operating Systems
Unique Paper Code: 223401404
Semester IV
May-June-2023
Year of admission: 2021

Time: Three Hours **Max. Marks: 70**

Instructions:

- All questions are compulsory.
- Attempt all the parts of a question together.

1. a. Outline the primary functions of the clock interrupt handler. Also, write the algorithm for `clock handler` including input and output parameters. [2+4]

   b. Consider the execution of four processes, namely, *A, B, C, D* and E, and the following assumptions: [8]

   (i) The processes are created simultaneously with an initial priority of 70.

   (ii) The highest user-level priority is 60.

   (iii) The clock interrupts the system 100 times a second.

   (iv) The processes make no system calls, and no other process is ready to run.

   (v) The process *A* and *B* are in one group and processes *C*, *D*, and *E* are in another group.

   (vi) The first process scheduled by the kernel is *B*. If two or more processes have equal priority, then the kernel picks processes in alphabetical order.

   (vii) The `priority` of a process is computed as:

   $Priority = decay(CPU)/10 + groupCount/10 +$ *base level user priority,*
   $decay(CPU) = 0.1 * CPU$ and $groupCount = 0.1 * Group$ respectively.

   (viii) *Fair-share* scheduling principal is used to schedule the processes.

   Show the priority calculation and scheduling of these processes for **four** seconds and the steps involved in the scheduling.

2. a. Explain the Master and Slave Processors environment in Multiprocessor Systems to avoid race condition. Also, mention two challenges of the [2+2+2]

system with proposed solution. Write the systemcall handler algorithm in this environment.

b. Consider two processes A and B with their process ids 100 and 200 respectively. Write code snippet in C/C++ for processes A and B such that: [6]

(i) The process A is able to handle Interrupt signal by invoking a user-defined function named mySigHandler. The mySigHandler function prints the message "Received Interrupt Signal" with the type of signal number received followed by the termination of the process.

(ii) The process A ignores segmentation fault signal and the other signals are handled by the default signal handler.

(iii) The process B sends a segmentation fault signal to process A followed by an interrupt signal to process A.

c. Write a system call to create a socket that supports IPv4 Internet domain, and fixed-length, unreliable, and connectionless messages. [2]

3. a. What is the significance of reference and modify bits in page table? [2]

b. Illustrate the concept of aging with the help of state diagram. Describe the role of aging in page stealing. [2+2]

c. Consider a process that invokes brk system call to increase the size of the process when there is no space in the memory. Enumerate the steps performed by the swapper. Also, draw the layout of the virtual addresses of the region of the process and swap device. Clearly mention your assumption(s), if any. [4]

d. Consider the following map data structure. Show the changes in the map for the following operations occurred in the sequence: [4]

Address	Units
1	20000

(i) Kernel swap-out a process P_1 that requires 500 units

(ii) Kernel swap-out a process P_2 that requires 400 units

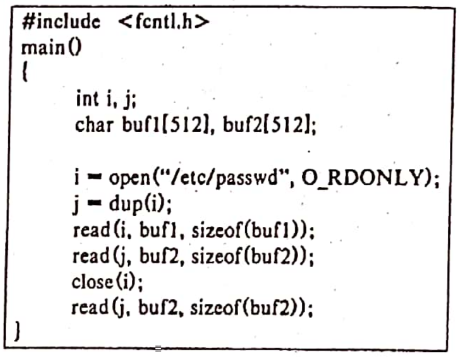
(iii) Kernel swap in a process P_1

(iv) Kernel swap-out a process P_4 that requires 200 units

4. a. Suppose a process wants to open the file "/etc/passwd". What are the steps involved in resolving the path to retrieve the *inode* of the file passwd? Assume that there the file passwd exists in the etc folder and the process is permitted to access the file and folder. Clearly mention your assumption(s), if any. [4]

b. Differentiate between management policy of *inodes* and *disk blocks* by the superblock. Also, show the configuration of super block to maintain *inodes* and *disk blocks* with the help of an example. [4]

c. Consider the following *C* code snippet. Show and explain the status of the kernel data structures after each system call. [6]



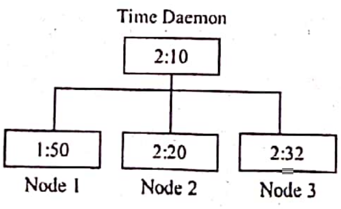
```
#include  <fcntl.h>
main()
{
        int i, j;
        char buf1[512], buf2[512];

        i = open("/etc/passwd", O_RDONLY);
        j = dup(i);
        read(i, buf1, sizeof(buf1));
        read(j, buf2, sizeof(buf2));
        close(i);
        read(j, buf2, sizeof(buf2));
}
```

5. a. Describe the issue of reliability in the client-server model in distributed systems. Write atleast two unique solutions to handle this. [4]

b. Consider a distributed system with 4 nodes (A, B, C and D) that uses Ricart and Agrawala' distributive Algorithm to resolve mutual exclusion. Assume that node B and C attempt to enter into critical section simultaneously with timestamps 14 and 11, respectively. Enumerate the various steps involved in handling mutual exclusion in this distributed system. Also, clearly state delay in entering (in terms of messages) the critical section. [4+1]

c. Write Berkeley's clock synchronization algorithm in the distributed systems. Considering the following scenario of a distributed environment, synchronize the clocks of all nodes using Berkeley's synchronization algorithm. Show all the intermediate steps. [5]


Time Daemon
2:10
1:50
2:20
2:32
Node 1
Node 2
Node 3