Department of Computer Science and Engineering
National Sun Yat-sen University
First Semester of 2022 PhD Qualifying Exam

Subject: **Operating Systems**

INSTRUCTIONS: *If any question is unclear or you believe some assumptions need to be made, state your assumptions clearly at the beginning of your answer.*

---

1. (10%) What are the necessary conditions for a deadlock to occur in a system?

2. (20%; 10% each) Suppose that a scheduler has k ready processes at time 0 and that no new processes are created after time 0. Process i ($0 < i \leq k$) requires i units of computing time. Answer each of the following questions.

   (a) For a preemptive, round-robin scheduler with a scheduling quantum of one time unit, what is the mean turnaround time for these processes, assuming that process k is at the front of the ready queue and that other processes appear in decreasing order of required computing time?

   (b) For a non-preemptive, shortest-job-first scheduler, what is the mean turnaround time for these processes?

3. (10%; 5% each) Consider the two-dimensional array `a`:

   ```
   double a[][] = new double[250][250];
   ```

   where each double occupies 8 bytes and `a[0][0]` is at location 200, in a paged system with pages of size 200 bytes. A small process is in page 0 (locations 0 to 199) for manipulating the matrix; thus, every instruction fetch will be from page 0. For three page frames, how many page faults are generated by the following array initialization loops, using LRU replacement and assuming (1) page frame 0 has the process in it, (2) the other two are initially empty, and (3) the array is stored in memory column-major. *Justify your answer for full credit.*

   (a)
   ```
   for (int i = 0; i < 250; i++)
       for (int j = 0; j < 250; j++)
           a[i][j] = 0;
   ```

   (b)
   ```
   for (int j = 0; j < 250; j++)
       for (int i = 0; i < 250; i++)
           a[i][j] = 0;
   ```

4. (20%; 4% each) Consider a file currently consisting of 150 blocks. Assume that the file control block is already in memory. Calculate how many disk I/O operations are required for contiguous and linked allocation strategies, if, for one block, the following conditions hold. In the contiguous allocation case, assume that there is no room to grow in the beginning, but there is room to grow in the end. Assume that the block information to be added is stored in memory.

   (a) The block is removed from the beginning.

   (b) The block is removed from the end.

   (c) The block is added at the beginning.

   (d) The block is added in the middle.

   (e) The block is added at the end.

5. (10%) What is the purpose of the `volatile` keyword in C?

6. (20%; 5% each) Given an i-node with ten direct blocks and three levels of indirect blocks and assuming that the sizes of a pointer and a block are, respectively, 4 bytes and 4 Kbytes, answer the following questions. *Justify your answer for full credit.*

   (a) What would be the size of the smallest file allowed in bytes?

   (b) What would be the size of the largest file allowed in bytes?

   (c) How many blocks are needed for a file of size 2138112 bytes?

   (d) How many blocks are needed for a file of size 4239360 bytes?

7. (10%) What would be the output of the following C program? (*Note that the line numbers are for reference only.*) *Justify your answer for full credit.*

```
1  #include <stdio.h>
2  #include <unistd.h>
3  #include <sys/types.h>
4  #include <sys/wait.h>
5  int main()
6  {
7      int status, fd[2];
8      pipe(fd);
9      pid_t pid = fork();
10     if (pid > 0) {
11         close(fd[1]);
12         close(0);
13         dup(fd[0]);
14         close(fd[0]);
15         waitpid(-1, &status, 0);
16         char buf[128];
17         int n = scanf("%s", buf);
18         printf("%d:%s\n", n, buf);
19     }
20     else if (pid == 0) {
21         close(fd[0]);
22         close(1);
23         dup(fd[1]);
24         close(fd[1]);
25         execl("/bin/echo", "echo", "how", "are", "you?", (void*) 0);
26     }
27     else {
28         return 1;
29     }
30     return 0;
31 }
```