Final Exam
CSE 3320.001
Spring 2017

Name: _____

UTA ID:_____

"I certify that the following work is my work alone and I will follow the highest standards of integrity and uphold the spirit of the Honor Code"

Signature:_____

Directions: This is a closed book, closed notes exam. You may use a hand written 3x5 index card with notes. Please answer the questions briefly. Complete sentences are not necessary. Write your answers legibly. Unreadable answers will be counted wrong. You may write on back if needed. There is a powers of two tables on the last page. Treat all abbreviated values as powers of 2, e.g. 4K = 4096

1. Explain the concept of shadow page tables, as used in virtualization.

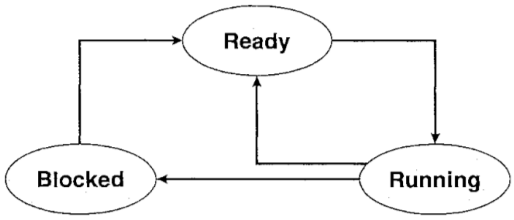
IGNORE THIS QUESTION FOR SPRING 2018

2. What is the difference between full virtualization and paravirtualization?

3. Given a page request reference string of:

A C D B C E A F A C G E D B A C F

and a page table size of 4, calculate how many page faults will occur with the Optimal page replacement algorithm. Assume no pages are initially loaded into the page table automatically. If all pages are equally replaceable pick the first available.



4. A simplified view of thread states is Ready, Running, and Blocked, where a thread is either ready and waiting to be scheduled, is running on the processor, or is blocked (for example, waiting for I/O). This is illustrated in the following figure. Assuming a thread is in the Running state, answer the following questions, and explain your answer:

a. Will the thread change state if it incurs a page fault? If so, to what new state?
b. Will the thread change state if it generates a TLB miss that is resolved in the page table? If so, to what new state?
c. Will the thread change state if an address reference is resolved in the page table? If so, to what new state?

Process ID	Arrival Time	Runtime (seconds)	Priority
1	0	5	5
2	1	6	4
3	3	1	1
4	6	2	4
5	10	2	1
6	13	1	2
7	14	2	1

5. Show the GANTT chart for a Shortest Job Next (SJN) with preemption scheduler. If there is a tie, pick the lowest PID. The jobs with the lowest priority value have the highest priority if applicable. What is the average wait time, average turnaround time and average response time?

6. What are the differences between a microkernel and a monolithic kernel? Give an advantage of a monolithic and a disadvantage of a microkernel kernel

7. Briefly explain priority inversion. How can you ensure it never occurs?

8. Describe, not just list, the four conditions for a deadlock

9. Given the following request queue -- 10,20,33,43,55,78,80 with the disk head initially at the track 50 initially moving in the positive direction (towards 200). The beginning of the disk at 0 and the end of the disk is at 200. Calculate the travel time for the SSTF algorithm. Assume all reads are made in the negative direction if applicable.

10. How much cylinder skew is needed for a 7200-RPM disk with a track-to-track seek time of 4 msec? The disk holds 1024 bytes per sector and has a data read rate of a single track is 147,456,000 bytes/ second.

11. For the following code, describe what happens, in what order, and what will get printed (in what order). For return values from fork use any integer that makes sense from a system standpoint. Assume all system calls succeed and all headers are included.

```
int main ()
{
  pid_t pid;
  int status;
  pid = fork();
  if (pid == 0)
  {
    wait( &status );
    execl( "/bin/gcc", "gcc", "main.c", NULL );
    printf("Compilation finished\n");
  }
  else
  {
    wait( &status );
    printf("Your executable is compiling.\n");
  }
  wait( &status );
  pid = fork();
  if (pid != 0){
    pid_t cp_pid = fork();
    if( cp_pid )
    {
      wait( &status );
      printf("Copying the file\n");
      execl( "/bin/cp", "cp", "a.out", "main", NULL );
    }
    else
    {
      wait( &status );
      printf("Removing the file\n");
      execl( "/bin/rm", "rm", "main.c", NULL );
    }
    execl( "/bin/rm", "rm", "backup.c", NULL );

  }
  else
  {
    int status;
    wait( &status );
    pid = fork();
    if( !pid )
    {
       wait( &status );
       printf("Done waiting for the child\n");
    }
    else
    {
      wait( &status );
      printf("Done with deliverable\n");
    }
    printf("Build process done!\n");
  }
  printf("Program complete\n");
}
```

12. Describe page-based virtual memory. You should consider addresses spaces, pages, frames, page tables, and Memory Management Units in your answer.

13. Explain the disk allocation schemes we discussed. Make sure to explain if they suffer from external fragmentation and if they support random access. Give a real-world example of each.

14. Given a file system that uses inodes to represent files. Disk blocks are 2 KB in size, and a pointer to a disk block are 16 bit. This file system's index nodes have 5 direct disk blocks, as well as 5 indirect disk blocks and 4 triple indirect blocks. What is the largest file that can be held using this inode layout? Calculate all values using powers of 2, e.g. 4K = 4096

15. Given five free fixed memory partitions of 50 KB, 200 KB, 500 KB, 350 KB, 2ß00 KB (in order), how would the first-fit, next-fit, best-fit, and worst-fit algorithms place processes of 125 KB, 75 KB, 60 KB, 255 KB, and 40 KB (in order)? Assume leftover space in the partitions may be reused but not combined.

16. What is an address space? How does the physical space differ from the virtual space?

17. What is the difference between kernel mode and user mode? Include software and hardware differences in your answer.

18. What is a kernel?

19. Given a 64 bit processor, with 1GB of physical RAM divided into 2K unique frames. How many page tables entries are in a single level page table if the maximum addressable virtual memory is 2^{64}?

20. You are given a choice of two TLB. Type A executes lookups parallel and has a memory access time of 125ns and a TLB lookup time of 13ns with a 73% hit rate. Type B executes lookups in serially and has a memory access time of 125ns and a TLB lookup time of 13ns and a 75% hit rate. Calculate the EAT for both. From a purely performance perspective, which TLB should you choose?

n	2^n	n	2^n	n	2^n
0	1	11	2,048	22	4,194,304
1	2	12	4,096	23	8,388,608
2	4	13	8,192	24	16,777,216
3	8	14	16,384	25	33,554,432
4	16	15	32,768	26	67,108,864
5	32	16	65,536	27	134,217,728
6	64	17	131,072	28	268,435,456
7	128	18	262,144	29	536,870,912
8	256	19	524,288	30	1,073,741,824
9	512	20	1,048,576	31	2,147,483,648
10	1,024	21	2,097,152	32	4,254,967,296