

Quiz 7

Course: cs241 - System Programming, CS Department

Date: March 31, 2006

Netid:

Name:

UIN:

Note: Completion of quiz is an individual effort. The quiz takes 10 minutes. The student gets additional 5 points for taking the quiz. *Each question has ONLY ONE ANSWER!!!*

1. (1 Point) Consider the following code segment

```
int myfd;
myfd = open("/home/ann/my.dat", O_RDONLY);
```

How can the call to open fail?

- file does not exist
 - file does not have the READ access right permission
 - file does not have the WRITE access right permission
 - (a) and (b)
 - (a) and (c)
2. (1 Point) Consider the following code segment

```
int myfd;
while ((myfd = open("/home/ann/my.dat", O_RDONLY)) == -1
      && errno == EINTR);
if (myfd == -1)
    perror("Failed to open the file");
else
```

How would you modify the code in Question 1 to open /home/ann/my.dat for non-blocking write?

- The modification cannot be done
 - You would OR the O_RDONLY with the O_NONBLOCK flags
 - You would AND the O_RDONLY with the O_NONBLOCK flags
 - You would OR the O_WRONLY with the O_NONBLOCK flags
 - You would AND the O_WRONLY with the O_NONBLOCK flags
3. (1 Point) What does it mean to open a file in non-blocking mode?
- File cannot be open in non-blocking mode
 - The open operation on the file descriptor will cause the calling process to wait
 - The calling process will not wait for open operation to return the file descriptor.

4. (1 Point) What can go wrong with the following code segment?

```
#define BLKSIZE 1024
char buf[BLKSIZE];

read(STDIN_FILENO, buf, BLKSIZE);
write(STDOUT_FILENO, buf, BLKSIZE);
```

- a. this is a correct code, nothing can go wrong with this code;
 - b. `write` may output garbage, although the `read` operation filled `buf` with `BLKSIZE` bytes correctly
 - c. `read` may fail to read full `BLKSIZE` bytes and then `write` will output garbage
5. (1 Point) How large a file can you access using only the single indirect, double indirect, and triple indirect pointers in the file inode if the block size is 8K (2^{13}) and pointers are 64 (2^6) bits?
- a. 8,388,608 bytes (8MB = 2^{23})
 - b. 8GB (2^{33})
 - c. 8TB (2^{43})
 - d. 16MB (2^{24})
 - e. 32GB (2^{35})
 - f. 64TB (2^{46})
 - g. (a) + (b)
 - h. (a) + (b) + (c)
 - i. (d) + (e) + (f)
 - j. (d) + (e)
6. (1 Point) File implementation based on inodes has the following limitation:
- a. File size cannot grow dynamically
 - b. File must fit entirely in a single disk partition
 - c. Files can be linked only through hard links
7. (1 Point) In a UNIX file-system the block-size has been set to 4K. Given that the inode blocks are already allocated on disk, how many free blocks need to be found to store a file of size 64K?
- a. 16
 - b. 17
 - c. 64
 - d. 65
 - e. none of the above

8. (1 Point) Which of the following methods would you choose if the file requires frequent direct access and also external fragmentation is to be avoided (to keep disk utilization high)?
- Linked allocation
 - Contiguous allocation
 - Indexed allocation
9. (1 Point) Which of the following is not a problem associated with contiguous allocation of disk space for a file?
- External fragmentation of disk space
 - Frequent copying
 - Random access
10. (1 Point) Given a file of 100 blocks, what is the minimum number of disk I/O operations needed to insert a block in the middle of the file if linked list allocation is used (assume the block to be inserted is already in the memory)?
- 2
 - 52
 - 101
 - 151