Tyler Paquet
CIS 452
Due: 04/19/2017

Homework Chapter 9, 10 & 11

9.11 Suppose that a disk drive has 5000 cylinders, numbered 0 to 4999. The drive is currently serving a request at cylinder 2150; the previous request was at cylinder 1805. The queue of pending cylinder requests, in order of arrival, is:

2069, 1212, 2296, 2800, 544, 1618, 356, 1523, 4965, 3681

Starting from the current head position (2150), give the order of service required to satisfy all pending requests, for each of the following disk-scheduling algorithms (the first algorithm has been given, as an example).

a. FCFS 2150 – 2069 – 1212 – 2296 – 2800 – 544 – 1618 – 356 – 1523 – 4965 – 3691
b. SSTF 2150 – 2069 – 2296 – 2800 – 3681 – 4965 – 1618 – 1523 – 1212 – 544 – 356
c. SCAN 2150 – 2069 – 1618 – 1523 – 1212 – 544 – 356 – 2296 – 2800 – 3681 – 4965
d. LOOK 2150 – 2296 – 2800 – 3681 – 4965 – 2069 – 1618 – 1523 – 1212 – 544 – 356
e. C-SCAN 2150 – 2069 – 1618 – 1523 – 1212 – 544 – 356 – 4965 – 3681 – 2800 – 2296
f. C-LOOK 2150 – 2296 – 2800 – 3681 – 4965 – 356 – 544 – 1212 – 1523 – 1618 – 2069

10.9 Consider a flawed file system where a file can be deleted and its former disk space reclaimed while other links (aliases) to that file still exist.
a. What problem may occur if a new file is created utilizing the same storage area the deleted file used?
b. How can this problem be avoided?

Let file1 be the old file and file2 be the new file. A user wishing to access file 1 through an existing link will actually access file 2. The access protection from file1 is also used rather than the one associated with file2. This problem can be avoided by ensuring that all links to a file are deleted when the file gets deleted. You could do this several ways..

1.) retain the links, removing them when an attempt is made to access a deleted file
2.) maintain a list of all of the links to a file, removing each of them when the file is deleted
3.) maintain a file reference list, deleting the file only after all links to that file have been deleted

11.15 Consider a file system on a disk that has both logical and physical block sizes of 512 bytes. Assume that files are always less than 512 blocks in size, and that directory information about each file is already in memory. For each of the three allocation strategies (contiguous, linked, and indexed), answer these questions:
b. If we are currently at logical block 10 (i.e. the current block accessed is block 10) and we now want to access logical block 4, how many physical blocks must be read from the disk (including the read of logical block 4)?

Contiguous: 1 - can calculate the exact block by dividing the logical address by 512. The integer quotient will be the block number.
Linked: 4 – can calculate block by dividing the logical address by 512. The integer quotient will be the block number. Then you can go through the links to find the correct block.

Indexed: 2 – can calculate the block by dividing the logical address by 512. The integer quotient will be the block number. Get the index block into memory, then the physical block address is contained in the index.

Practical: Locate the definition of the inode structure in the Linux source code. Hint: recall that definitions typically appear in <include> files.

- What are the first five fields of the inode structure?

The first five fields of the inode structure are….

```
umode_t         i_mode;
unsigned short  i_opflags;
kuid_t          i_uid;
kgid_t          i_gid;
unsigned int    i_flags;
```