

104062203 陳涵宇

1. Problem Set

1. Contrast the performance of the three techniques for allocating disk blocks (contiguous, linked, and indexed) for both sequential and random file access.

contiguous sequential: Good, since the file is stored contiguously.

contiguous random: Good, it is easy to determine the adjacent disk block containing the position wish to seek to.

linked sequential: Not so bad, for it

linked random: Not so good since it may require many times to following the links before arriving the seeking point of file.

indexed sequential: Good, because sequential access involves sequentially accessing each index.

indexed random: Good, it is easy to find the position of seeking point by index.

2. Consider a file system that uses inodes to represent files. Disk blocks are 8 KB in size, and a pointer to a disk block requires 4 bytes. This file system has 12 direct disk blocks, as well as single, double, and triple indirect disk blocks. What is the maximum size of a file that can be stored in this file system?

$(12 * 8 \text{ KB})$ (direct disk blocks)
+ $(2048 * 8 \text{ KB})$ (single indirect disk blocks)
+ $(2048 * 2048 * 8 \text{ KB})$ (double indirect disk blocks)
+ $(2048 * 2048 * 2048 * 8 \text{ KB})$ (triple indirect disk blocks)
= 64 terabytes

Where 2048 comes from $8\text{KB} / 4\text{bytes} = 2048$

2. Programming Problems

2.1 What are the inode values of file1.txt and file2.txt? Are they the same or different?

Do the two files have the same—or different— contents?

file1.txt: 393901 file2.txt: 393901

They have the same content.

2.2 Next, edit file2.txt and change its contents. After you have done so, examine the contents of file1.txt. Are the contents of file1.txt and file2.txt the same or different?

Contents of file1.txt and file2.txt are same.

2.3 Next, enter the following command which removes file1.txt:

rm file1.txt

Does file2.txt still exist as well?

Yes, file2.txt still exists.

2.4 Now examine the man pages for both the `rm` and `unlink` commands. Afterwards, remove file2.txt by entering the command

```
strace rm file2.txt
```

The `strace` command traces the execution of system calls as the command `rm file2.txt` is run. What system call is used for removing file2.txt?

```
unlinkat(AT_FDCWD, "file2.txt", 0)
```

2.5 A soft link (or symbolic link) creates a new file that “points” to the name of the file it is linking to. In the source code available with this text, create a soft link to file3.txt by entering the following command:

```
ln -s file3.txt file4.txt
```

After you have done so, obtain the inode numbers of file3.txt and file4.txt using the command

```
ls -li file*.txt
```

Are the inodes the same, or is each unique? Next, edit the contents of file4.txt. Have the contents of file3.txt been altered as well? Last, delete file3.txt. After you have done so, explain what happens when you attempt to edit file4.txt.

```
file3.txt: 394015   file4.txt: 393486
```

The inode numbers of them are different.

The contents of file3.txt are altered after editing the contents of file4.txt.

After deleting file3.txt, file4.txt turns to be a read-only file and thus can't be edited.