Files, Links and Inodes

Sources:

http://web.eecs.utk.edu/~jplank/plank/classes/cs360/360/notes/Links/lecture.html

http://140.120.7.21/LinuxKernel/LinuxKernel/node17.html

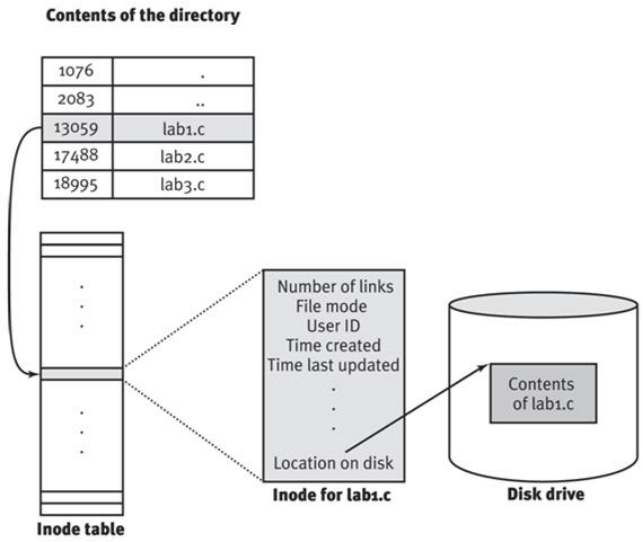
http://faculty.salina.k-state.edu/tim/unix_sg/advanced/links.html

Files, Links and Inodes

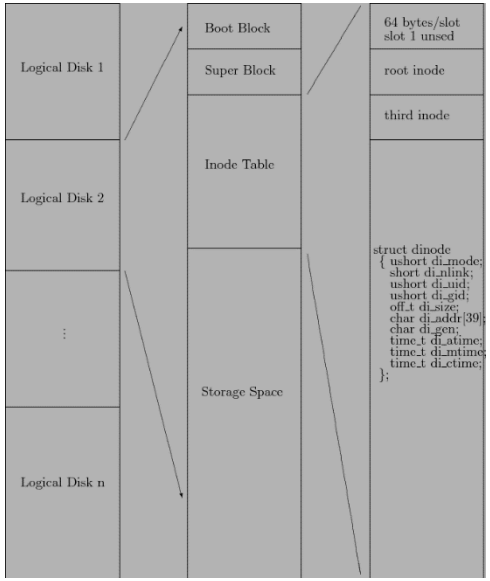
- Each physical file on the disk is stored in the operating system with a data structure called inode.
- Each time we create a file in a directory, the system simply allocates a free I-number from the file system and uses an empty slot in the correspondent directory to record this I-number and the name of the file we created.

Files, Links and Inodes

- Each inode, an entry in the inode table, is a data structure which the system uses to store the following information about a file:
  - Type of file (ordinary, directory or special file).
  - Access permissions for the file owner, the owner's group members and others (i.e. the general public).
  - Number of links.
  - File owner's user and group IDs
  - File size in bytes.
  - The disk addresses of the data blocks where the contents of the file are actually stored.
  - Time of last access (read or executed), time of last modification (i.e. written) and time which the inode itself was last changed.



Source: http://faculty.salina.k-state.edu/tim/unix_sg/advanced/links.html



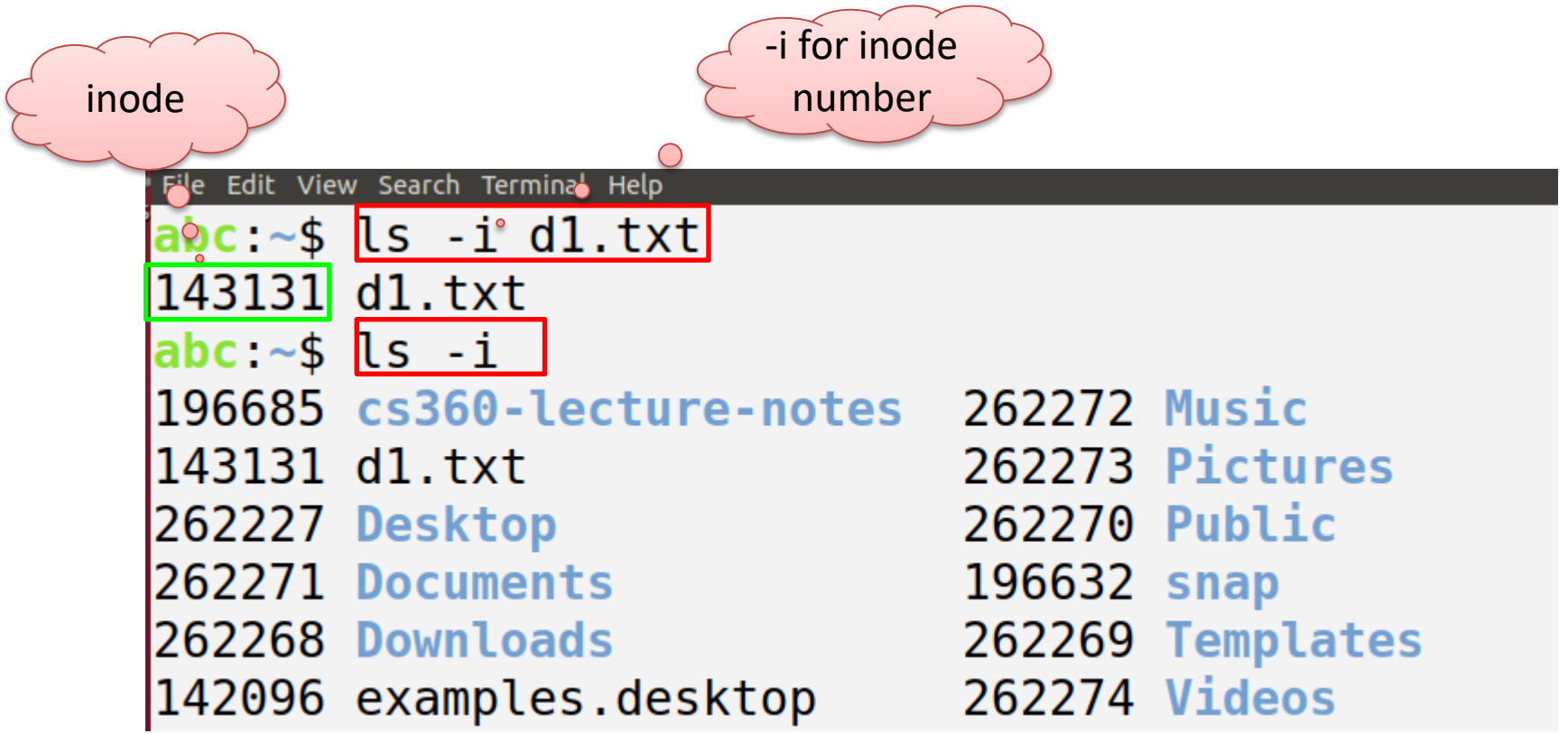
Source: http://140.120.7.21/LinuxKernel/LinuxKernel/node17.html

Files, Links and Inodes

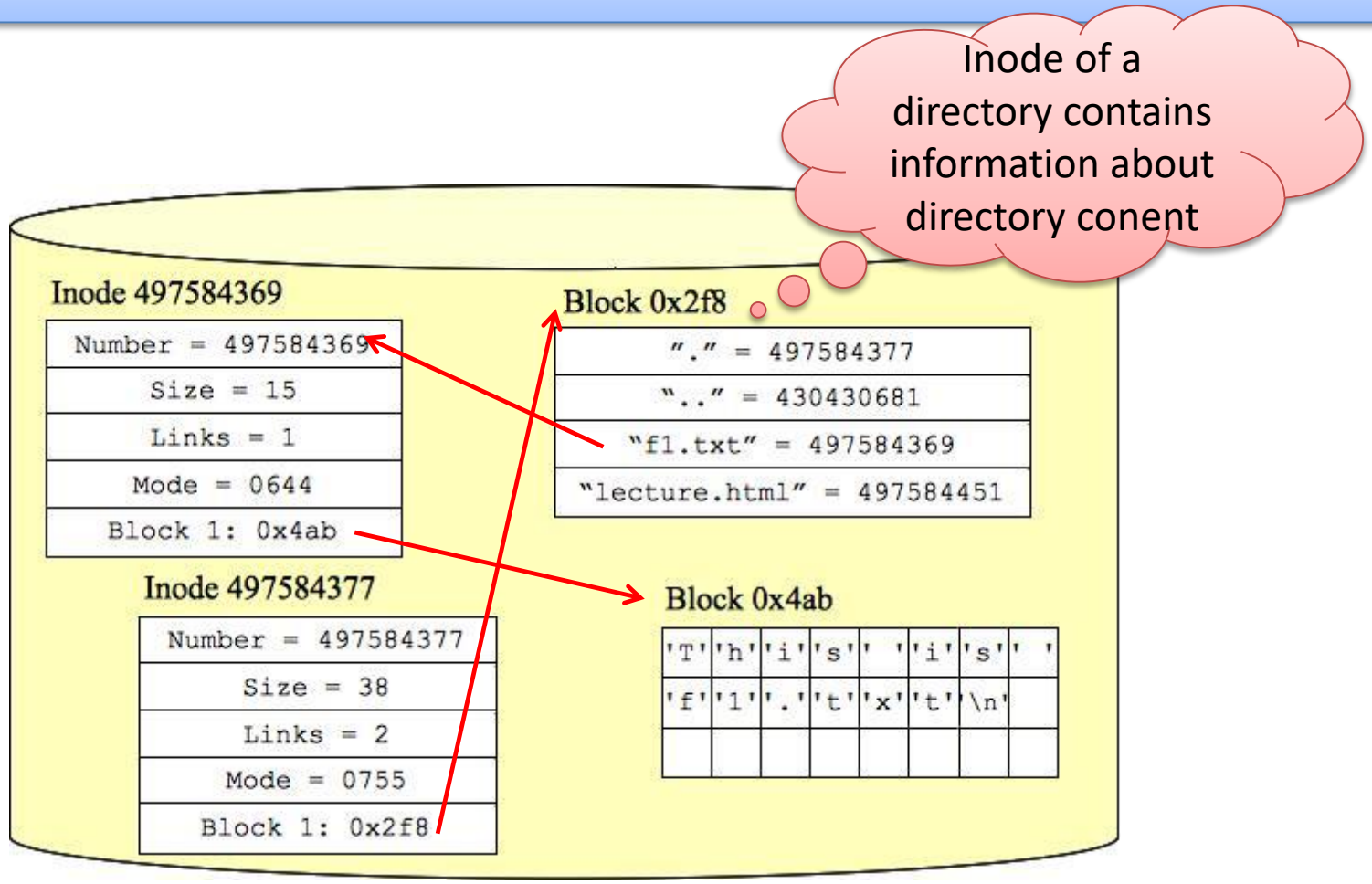
- In this lecture, we are going to focus on three components of a file in Unix:
  - The bytes of the file itself.
  - The metadata of the file.
  - The links to the file that are relative to a directory.
- Link
  - Hard links
  - Soft (Symbolik) links

inode

- We can learn the inode number of a file using **ls** together with **–i** flag from Shell screen



inode



- You can see the inode for f1.txt (497584369), and how it points to block 0x4ab, which contains the bytes of the file (you don't have direct access to this information. But you can find the address of the first character indirectly)

Source: http://web.eecs.utk.edu/~jplank/plank/classes/cs360/360/notes/Links/lecture.html

Creating a link

- There may be more than one link pointing to a file

```
File  Edit  View  Search  Terminal  Help
abc:test$ echo "This is the content of f1.txt" >f1.txt
abc:test$ ls -li f1.txt
263287 -rw-r--r-- 1 abc abc 30 Apr  5 19:56 f1.txt
```

ln creates a link between files.

```
abc:test$ ln f1.txt f2.txt
abc:test$ ls -li f1.txt f2.txt
263287 -rw-r--r-- 2 abc abc 30 Apr  5 19:56 f1.txt
263287 -rw-r--r-- 2 abc abc 30 Apr  5 19:56 f2.txt
```

f1.txt and f2.txt points to the same file. They have the same inode.

Number of links pointing to the same file.

Creating a link

- There may be more than one link pointing to a file

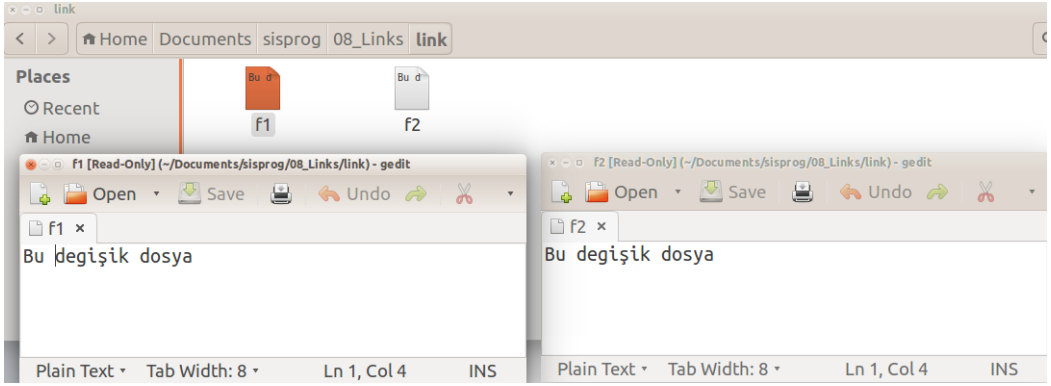
Contents of the file are the same

```
abc:test$ ls -li f1.txt f2.txt
263287 -rw-r--r-- 2 abc abc 30 Ap
263287 -rw-r--r-- 2 abc abc 30 Ap
abc:test$ cat f1.txt f2.txt
This is the content of f1.txt
This is the content of f1.txt
abc:test$
```

File content

f1 ve f2 aynı dosyaya işaret ettiği için değişiklik ikisinde de oluşur

```
File Edit View Search Terminal Help
link: cat f1
Bu degişik dosya
link: cat f2
Bu degişik dosya
link:
```

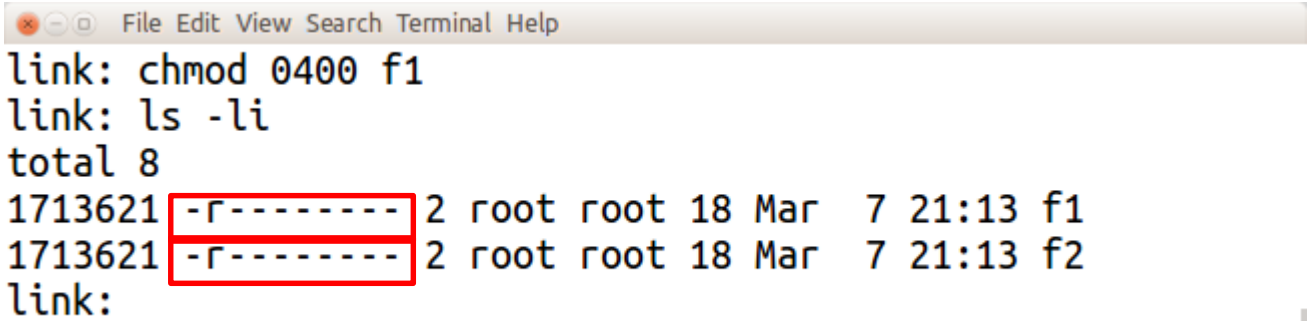



```
File Edit View Search Terminal Help
link: ls -li f1 f2
1713621 -rw-r--r-- 2 root root 18 Mar  7 21:13 f1
1713621 -rw-r--r-- 2 root root 18 Mar  7 21:13 f2
link:
```

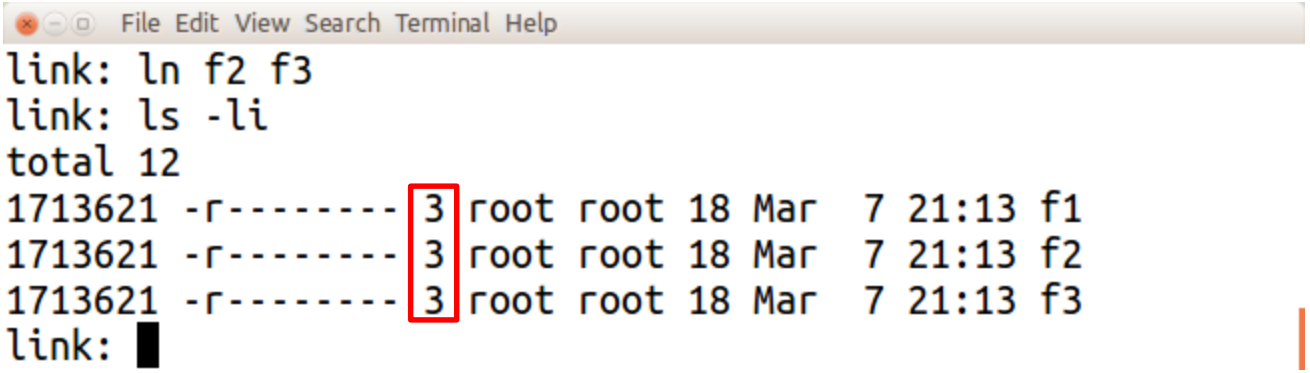
- Düzenleme zamanı dosyanın bir parçası olarak saklandığı için ikisinin de düzenleme zamanları aynıdır.

Changes on links

- If we change the protection mode of one file, it changes for the other.

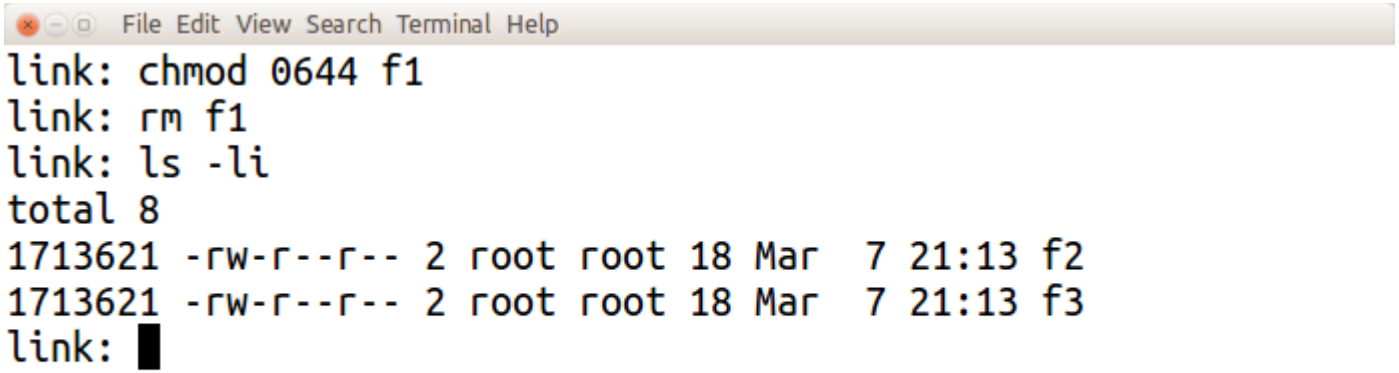


- The number of links to the file can be increased.



Removing link

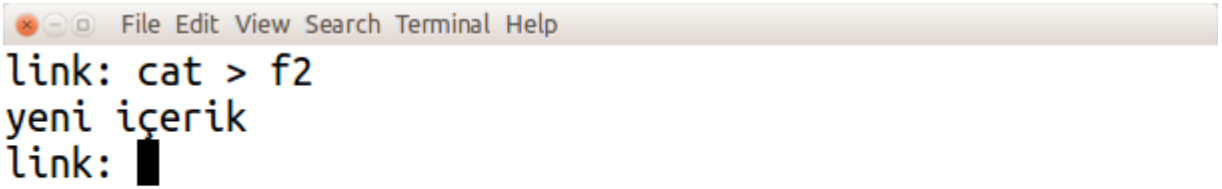
- Link can be removed using **rm**



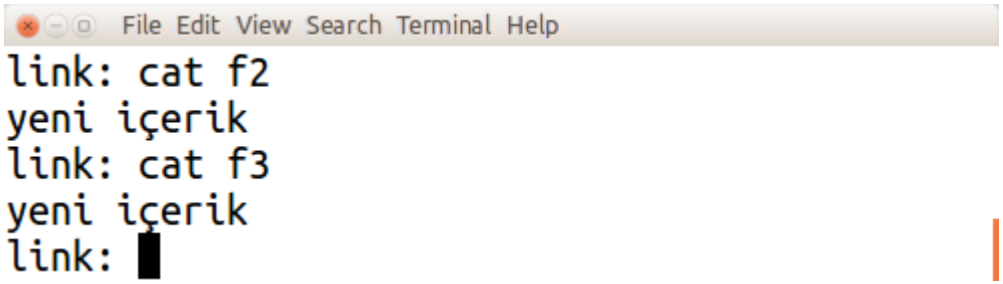
- The file is not deleted until the last link is deleted.

Üzerine yazma (overwrite)

- Cat ile f2 dosyası oluşturulup yeni bir içerik eklenirse mevcut dosya silinmez.

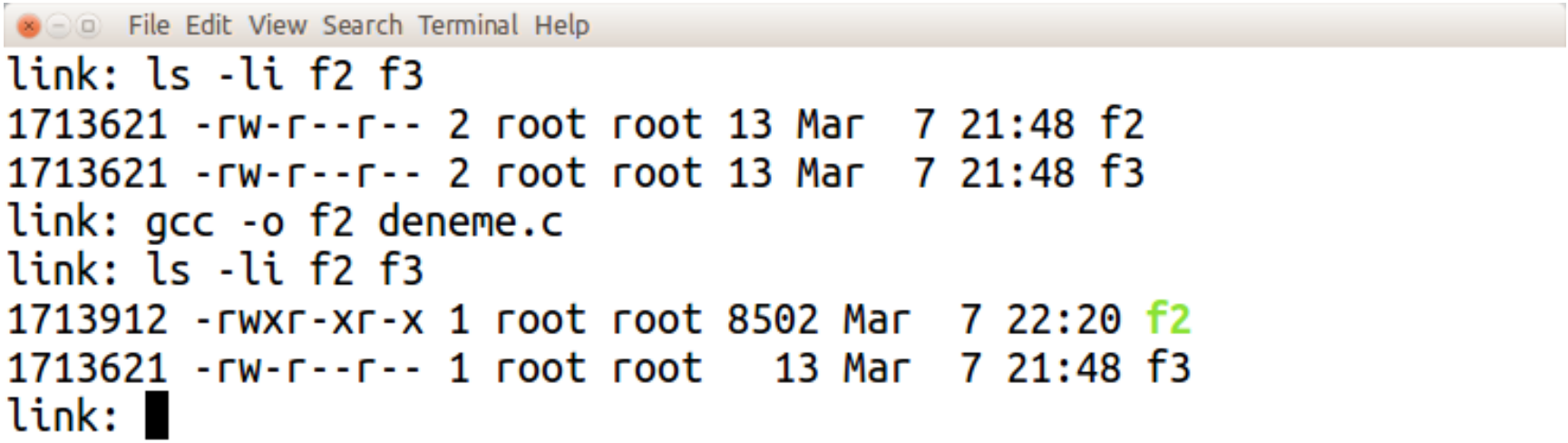


- f2 ve f3 bağlantıları aynı dosyayı göstermeye devam eder.



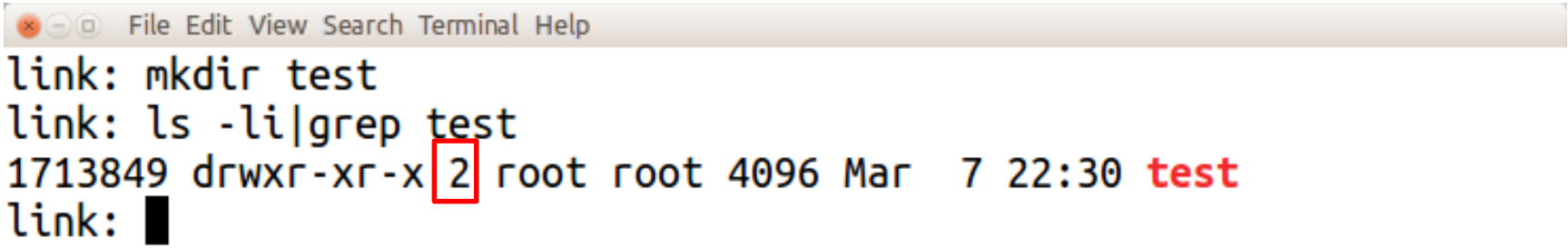
Overwriting

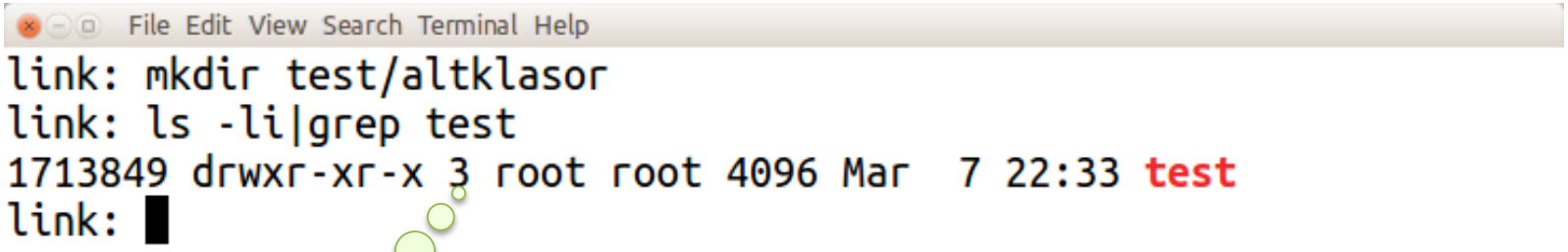
- C compiler ile f2 isimli bir dosya derlediğimiz zaman önce rm f2 gerçekleşir ve f2 isimli yeni bir yürütülebilir dosya oluşturur.



Dosyaların link sayısı

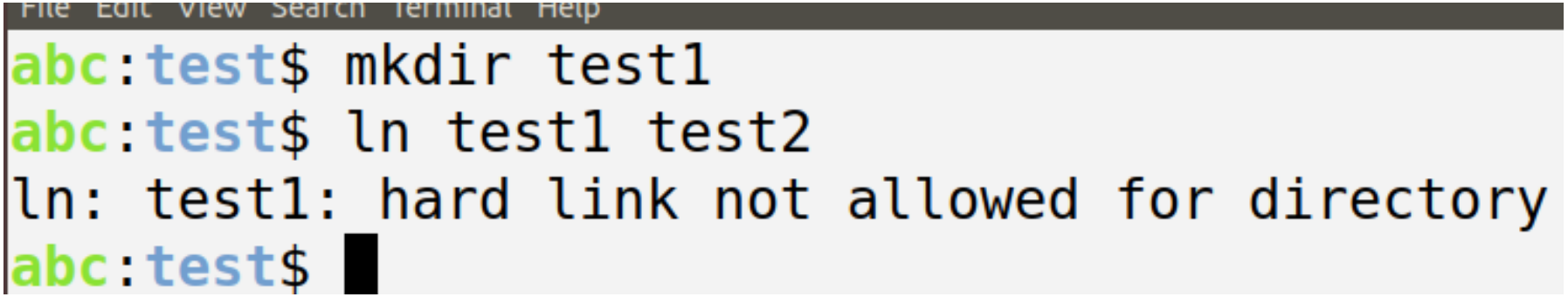
- Each newly created directory contains “.” and “..” subdirectories.
- The first one points to the same directory and the other one points tyo the parent directory.





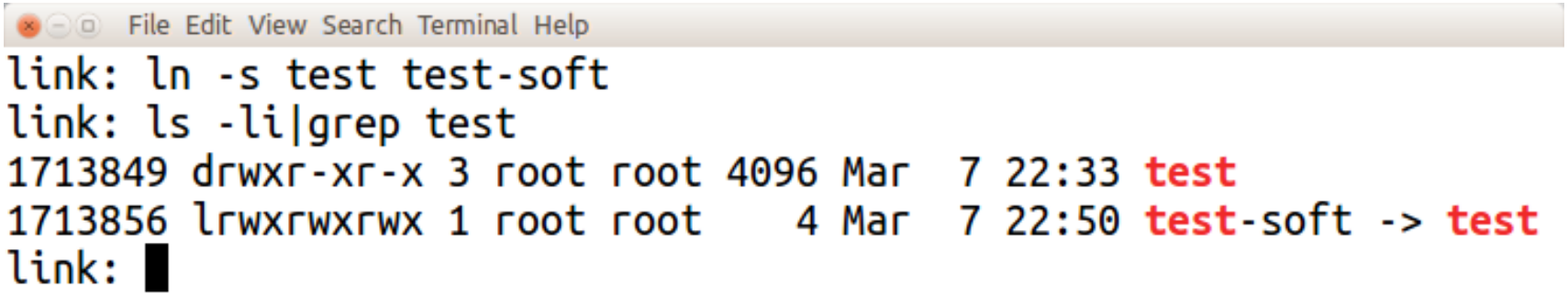
Yeni bir alt klasör oluşturduk ve link sayısı 3’e çıktı

Hard link for a directoriy is not alllowed



Soft link (Symbolic link)

- hard links : Refer to the specific location of physical data.
- symbolic links: Refer to a symbolic path indicating the abstract location of another file
- Soft link can be created with ln -s.
- Soft links point to the file without modifying the inode.



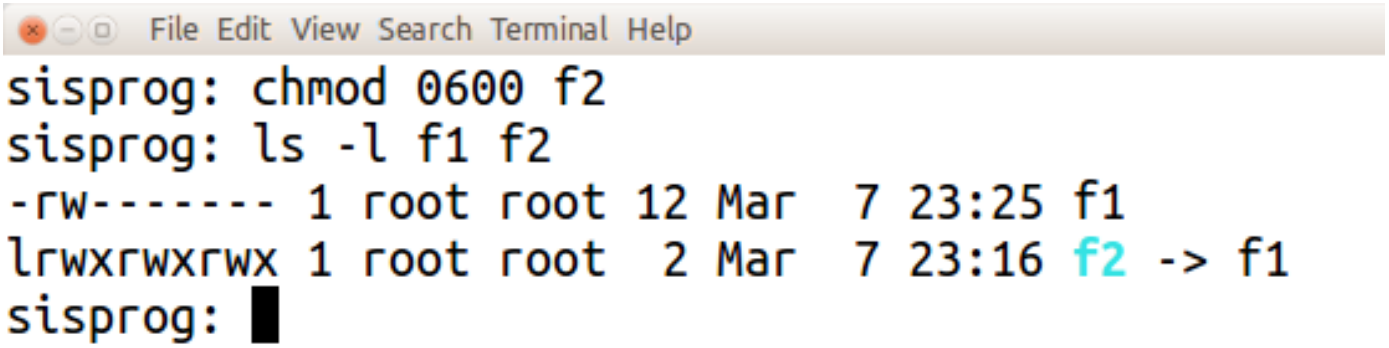
Soft link (Symbolik link)

```
File Edit View Search Terminal Help
sisprog: cat > f1
f1 dosyası
sisprog: ln -s f1 f2
sisprog: cat f2
f1 dosyası
sisprog:
```

Symbolic link

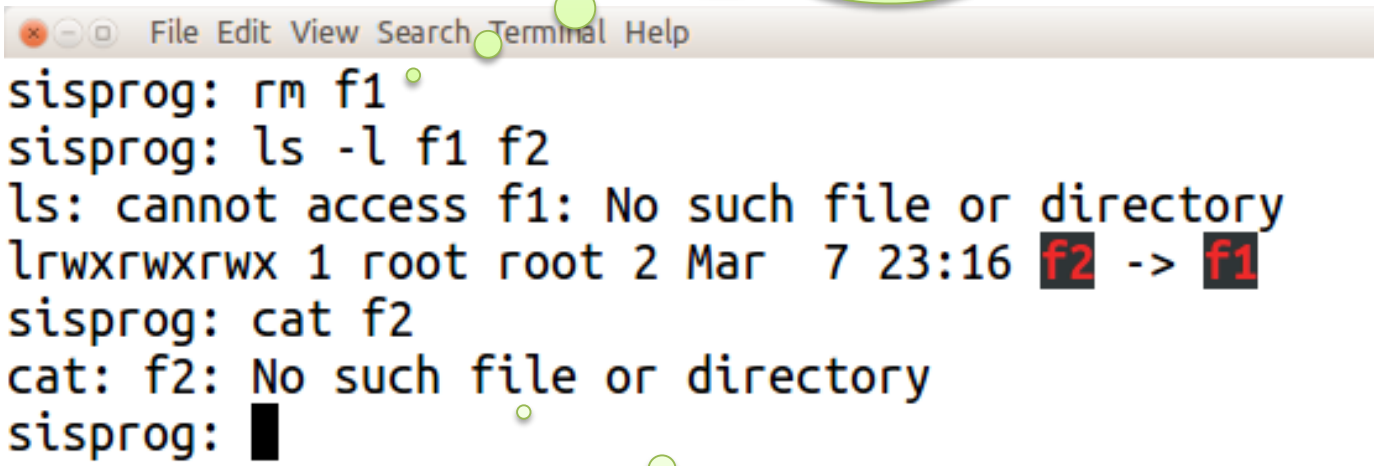
```
File Edit View Search Terminal Help
sisprog: cat > f2
f2 dosyası
sisprog: cat f1
f2 dosyası
sisprog: ls -l f1 f2
-rw------- 1 root root 12 Mar  7 23:25 f1
lrwxrwxrwx 1 root root  2 Mar  7 23:16 f2 -> f1
sisprog:
```

Soft link (Symbolik link)



Soft link (Symbolik link)

Hardlink is removed



An error occurs when we try to access it via the symbolic link.

inode

- Hard links have the same inode number as the source file.
- Soft links have a different inode number than the source file.
- When the last hardlink is deleted, the file is deleted even if the soft links are not deleted.
- A hardlink cannot be created from one file system to another.

inode

- Making shortcut:

- **Documents: ln -s /home/bilg/Documents/deneme.txt /home/bilg/Desktop/kısayoldeneme**

- Deleting the file using its the inode number
- **find -inum** 1969436 **-delete**

find -inum

```
abc:test$ ln f3.txt f4.txt
abc:test$ ls -li
total 16
263399 -rw-r--r-- 4 abc abc 10 Apr  7
263399 -rw-r--r-- 4 abc abc 10 Apr  7
263399 -rw-r--r-- 4 abc abc 10 Apr  7
263399 -rw-r--r-- 4 abc abc 10 Apr  7
abc:test$ cat f4.txt
zxcvbnm
abc:test$ find -inum 263399
./f2.txt
./f4.txt
./f3.txt
./f1.txt
```

Finds all files with the same inode

Finds and deletes all files with the same inode

```
abc:test$ find -inum 263399 -delete
abc:test$ ls
abc:test$
```