Charandeep Parisineti,
Ravali Kandur

Kernel.cc

static int link(char pathname1,char* pathname2)*

This method is used to create a link at path “pathname2” for the file at “pathname1”

Terminal Usage: ./ln /pathname1 /pathname2

Result: New link “pathname2” is created

static int unlink(char pathname)*

This method is used to delete the link at path “pathname”.

Terminal usage: ./rm /pathname

Result: link “pathname” is deleted.

Notes: 1) The original file can also be deleted using ./rm /filename

2) Never unlink the original file before its link.

static int deleteFile(char path)*

This is a supporting method for unlink. It frees block memory for the file when nlinks is 0.

static int removeDirectoryEntry(char name, char* dirname)*

This is a supporting method for unlink. It is used to remove the directory entry “name” from directory “dirname”

IndexNode.cc

void setIndirectBlock(int address)

Sets the address of indirect block of the index Node to “address”

int getIndirectBlock()

Returns the address of the indirect block.

static void setDataBlockOffset(int dbOffset)

Used to set the data block offset in the filesystem in the index Node. This is used as writing indirect blocks to filesystem is done in IndexNode.cc. Please take a look at IndexNode.cc for details or email me at paris102@umn.edu.

fsck.cc

countNlinks(parentdir,currIndexNode)

Counts the nlinks in the directory “parentdir” and also reads the nlinks from indexNodes. They are stored in nlinks and inode_nlinks arrays respectively.

checkBlocks(parentdir,currIndexNode)

Checks the allocated blocks for all the Index Nodes in the directory “parentdir”. It stores the allocation data to blocks array. The blocks array is then compared to the filesystem allocation of all the blocks and the status is printed.

Usage: ./fsck