Homework 5: Reading the xv6 filesystem in Linux - Documentation

The hw5 routine can perform one of the following operations:

- List all root directory entries in the given img file (“ls”).
- Copy a file located in the root directory in the given img file to a linux file (“cp”).

Routine:

The hw5 routine consists of a few stages:

1. Get user arguments.
2. Parse superblock and root inode.
3. Scan root directory – used for both “ls” and “cp” commands.
   a. Go over all blocks in root directory. The routine supports both direct and indirect blocks.
      i. For each block:
         1. Go over all dirents.
         2. For each dirent:
            a. Get all interesting fields: name, type, inum, size.
            b. In case of “ls” command – print the details according to the given format.
            c. In case of “cp” command - check if we found the expected file (if the name matches the user request, and the type is a file type). If the file is found – can stop searching.
4. Copy file – used only at “cp” command.
   a. Go over all blocks in the file. The routine supports both direct and indirect blocks.
      i. For each block – copy the data from the source to the destination file.

Filesys module:

Structs (taken from *xv6-public/fs.h*):

1. ***superblock***:

```
typedef struct superblock {
  uint size;         // Size of file system image (blocks)
  uint nblocks;      // Number of data blocks
  uint ninodes;      // Number of inodes.
  uint nlog;         // Number of log blocks
  uint logstart;     // Block number of first log block
  uint inodestart;   // Block number of first inode block
  uint bmapstart;    // Block number of first free map block
} superblock;
```

2. ***dinode:***

```
typedef struct dinode {
  short type;           // File type
  short major;          // Major device number (T_DEV only)
  short minor;          // Minor device number (T_DEV only)
  short nlink;          // Number of links to inode in file system
  uint size;            // Size of file (bytes)
  uint addrs[NDIRECT+1];   // Data block addresses
} dinode;
```

3. ***dirent:***

```
typedef struct dirent {
  ushort inum;
  char name[DIRSIZ];
} dirent;
```

Functions:

- `int get_block_idx_in_fs(int block_num_in_dir, dinode *dir_inode, FILE *fp);`
  This function finds the block index in the filesystem, based on its number in the directory. Using this function, we support both direct and indirect blocks.
- `void get_inode_size_type(ushort inum, int inodestart, FILE *fp, int *inode_size, short *inode_type);`
  This function returns the size and type of an inode (used for each root entry).
- `void goto_inode(ushort inum, int inodestart, FILE *fp);`
  This function navigates the file pointer to the beginning of a given inode number.
- `void copy_file(ushort inum, int inodestart, FILE *fp_src, FILE *fp_dst);`

  This function copies a file from the xv-6 filesystem image to a linux destination.

- `bool scan_root_dir(dinode* root_inode, FILE* fs_img_fp, superblock* sb, UserArgs* user_args, int* inum_of_file_to_copy);`
  this function performs the scanning of xv-6 root directory, used for both “ls” and “cp” commands.