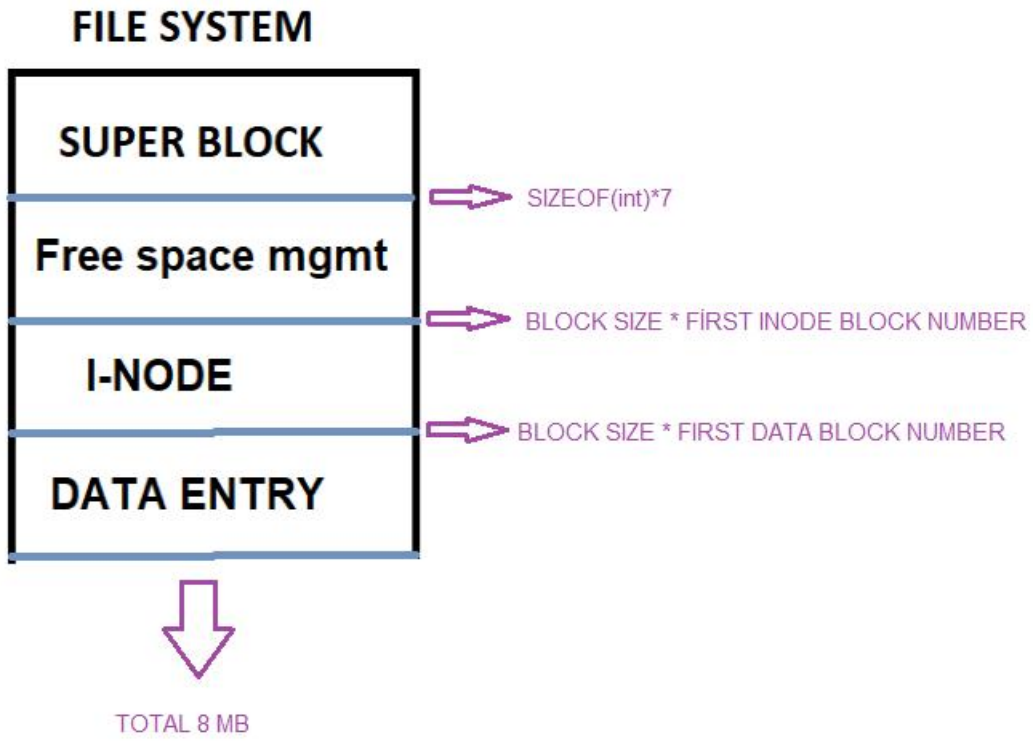
CSE 312 HOMEWORK 3

AHMET OKUR 1801042655

1.OVERWİEV



- The simple view of the file system is as follows.
- First, I write the super block to file. The super block contains information about the system.
- secondly, I am writing the space management data to file. Space management stores the availability of data
- Third, I write the inodes. Here I set the number of inodes as 200.
- Finally, I write the data entries. The data entry blog is as in the book.

2. STRUCTURES

```
class super_block
{

    public:

        super_block(int block_size,int inode_number,int total_blocks_number);
        super_block();
        void print();
    public:
        int block_size;
        int inode_number;
        int total_blocks_number;
        int blocks_inode_number;
        int superblock_and_spacem_block;
        int first_inodes_block_num;
        int first_data_block_num;

};
```

- The system is determined as 8 Mb, so the block size will be determined by the user according to 8 Mb
- The number of inodes is set to 200
- Other information is saved in the system for direct access while searching the disc.

```
class space_management
{
    public:
        int  free_block_num;
        char free_inode_blocks[200];
        char *free_data_blocks;
    public:
        void space_management_items(int total_blocks_number,int first_data_block_num);
};
```

- Domain management tells us which blog is empty and which node is not used, so we can directly access the node and blog we want.
- Denoted by 'f' means empty
- Specified by 'u' means being used

```
class i_node
{
    public:
        i_node();
    public:
        bool is_dir;
        int size;
        int blogs[13];
        char modified_time[20];
};
```

- is_dir determines whether the node is a folder or file.
- The blogs array holds 13 data blog addresses, so we can access the data directly.

```
class data_entry
{
    public:
        data_entry(short node_id , char file_name[14]);
    public:
    short  node_id;
    char file_name[14];
};
```

- The data entry holds a 14-byte name and a 2- byte inode address as in the book.

```
class creation_file_system{
    private:
        int block_size;
        int inode_number;
        int total_blocks_number ;
        FILE * fp;
        super_block superBlock;
        space_management spaceManagement;
        i_node inode;
        i_node root;
        void write_superblock();
        void write_space_management();
        void write_inodes();
        void write_inode_dir();
        void write_data();
        void read_information();


    public:
        creation_file_system(int block_size,int inode_number,int total_blocks_number,FILE *fp);

};
```

- This is my main class I make the file system in this class

I didn't do part 3 . I just pulled data from the disk for one read