

Operating System Lab - CSE 341

Faculty: *Prof. Mansukh Savaliya*

Project Progress Report: *Week 5*

Group Number: 3

Group Members:

Name	Enrolment Number
Yashil Depani	AU1841005
Vidish Joshi	AU1841019
Manav Patel	AU1841037

OSProject - toaruOS

An operating systems project on understanding and solving issues for the open source OS - [toaruOS](#)

Index

1. [Printing Directories](#)
2. [Formatting/\(proper naming convention\) Directory Name](#)
3. [Change the directory](#)
4. [Reading File](#)
5. [Cluster Info](#)
6. [Cluster Size](#)
7. [Directory Info](#)
8. [Plan for coming weeks](#)
9. [Contribution](#)

Printing Directories

This function works as "ls" command, just traversing through all the directories and printing them. For this, we need offset and then we are pointing the file pointer to that offset location in the image file and along with this, directory attributes are also required for the validation purpose (It can be obtained from the defined property of directory). Finally, after all validations and other logic print the file/directory name.

Formatting Directory Name

This function formats the proper naming convention for the file directories in order to store it. This should be in capital letters (max 8 characters) and followed by 3 characters of extension.

This is based on fact that name of a file/directory actually consists of 2 names: The old (short) 8.3 convention and the new (long) 255-character name. This function updates both these names whenever you give a name to the file. The short name is created automatically, based on the first characters of the long name (e.g. SyncToy_4264be.dat becomes SYNCTOY~1.DAT). If the name fits into the old naming convention, the long name is discarded. Hence through this function, it converts all characters to uppercase.

Change the directory

This function does the work of " cd .. " command. For every directories in img file, there are two pointers like ".." and ".". "." is associated with parent directory of that current directory. So whenever user writes "cd .." , the function after validating the input, gets the offset of parent directory. Changes the pointer to that directory through *fseek* and reads that pointer so now we are in that parent directory.

Reading File

This function does the work of reading the file. For this we pass the filename as the argument. We call the *getCluster* function to know the cluster of that file. With the help of this cluster we get the offset. We move the pointer to that file through that offset with the help of that *fseek* function. Now we read the data of the file and store it in a character array and finally print this array.

Cluster Info

This function is used to get the cluster name given the directory name as an argument.

To get the cluster number for a directory name, we first get the formatted name of the directory. Then we traverse through the list of all the directories and get the formatted name of the current directory under consideration. If this current formatted name matches with the formatted name of the directory name passed as argument, then we have got a match. We get the cluster number from the structure of that directory name and end the function.

The default return value is -1 representing that no matching directory found to get the cluster information.

Cluster Size

This function is used to get the size of cluster given the cluster number as argument.

We traverse through the list of all the directory names and if the cluster number of directory under consideration matches with the cluster number passed as argument, then we have gotten a match. We return the size of the cluster from the structure of the cluster.

The default return value is -1 if no such cluster exists.

Directory Info

This function shows the information like size, attributes, starting cluster, cluster high. For this, the cluster is obtained from the given directory name and then getting the size of cluster which is actual size of directory and then obtain information from the given statistics like cluster low, attribute, cluster high and everything that is stored as the property of the directory.

Plan for coming weeks

Write the functions that allow us to change the contents of the .img files. In other words, functions to create, alter and delete directories inside the root directory.

Test the code with several .img files.

Integrate the code with base operating system.

Contribution

Writing, debugging and testing the code/functions - *All 3 members*