Coursework 2: scandisk

Part 1: Print out a list of clusters that are not referenced from any file.

- The '**follow_dir**' function, that is from the '**dos_ls'** file has been edited slightly for this coursework. This function is used with the parameter '**find_inconsistencies**' so that the function '**mark_referenced_cluster**' is called, which marks all clusters referenced by any directory entry and 'tags' the cluster number with a value of '-1' (the tag numbers are explained below) and '-3' if a cluster is supposed to be an end of file cluster but isn't (for part 4)
- Unreferenced clusters are found by calling the function '**find_unreferenced_clusters**', which iterates through the cluster_tag array and changes the tag value to '-2' if the cluster is not empty but it is referenced (i.e. it has a cluster tag value not equal to 0)

Part 2: From the unreferenced blocks your code found in part 1, print out the files that make up these blocks.

- The function '**find_lost_files**' iterates through the cluster_tag array and when a cluster tag value of '-2' is found, it stores the start cluster of the lost file and calls the '**recover_lost_file**' function

---

Part 3: Create a directory entry in the root directory for any unreferenced files. These should be named "found1.dat", "found2.dat", etc.

- '**recover_lost_file**' generates the file name (found1.dat, found2.dat etc) and creates a directory entry for the file with the functions '**create_dirent**', '**write_dirent**' which are from '**dos_cp**'

Part 4: Print out a list of files whose length in the directory entry is inconsistent with their length in the FAT. Print out the filename, its length in the dirent and its length in the FAT

- '**follow_dir**' is called again but with the parameter '**find_inconsistencies**' set to 1 so that '**find_length_inconsistencies**'. This function iterates through the cluster tag array and finds any cluster that has a tag of '-3'. It then calculates the number of extra clusters by calling the '**calculate_number_of_extra_clusters**', which counts the number of clusters after the one with the '-3' tag until an end of file cluster is found

---

Part 5: Free any clusters that are beyond the end of a file.

- '**free_clusters**' is called after the number of extra clusters is called. It sets the fat entry of all the clusters after the '-3' tag as empty clusters and stops when the end of file cluster is found.

Cluster tagging

- A cluster_tag array stores a 'tag' for each cluster number with the following values:
  - '0' - empty cluster
  - '-1' - referenced cluster
  - '-2' - unreferenced cluster (part of a lost file)
  - '-3' - cluster is supposed to be an end of file cluster but isn't (part 4)