Task 2. STAR: Simple TAR-like file archiving program

Overview

This task asks you to construct `star`, a simplified version of a file-archiving program `tar`. `star` provides two functionalities. First, it generates a file that aggregates all files in a specified directory (i.e., archive). Second, for a given archive file generated by itself, star restores all files with their original file names (i.e., extract).

Background

As first step, you need to review the basics of file and file system interfaces. Read each of the given C library reference pages and solve the associated problems.

(1) Read and write a file as binary (not as text)

- `fread()` : http://www.cplusplus.com/reference/cstdio/fread/
- `fwrite()`: http://www.cplusplus.com/reference/cstdio/fwrite/
- What's the other file interfaces that read and write a file as text?
- How accessing a file as binary is different from accessing it as text?
- Write a program that shows each byte of a given file as a hexadecimal number (like xxd does)

(2) Accessing a directory.

- Opening a directory
  https://www.gnu.org/software/libc/manual/html_node/Opening-a-Directory.html
- Reading a directory stream
  https://www.gnu.org/software/libc/manual/html_node/Reading_002fClosing-Directory.html
- Simple directory lister
  https://www.gnu.org/software/libc/manual/html_node/Simple-Directory-Lister.html
- What is the *path* of a file? How it is different from the *name* of a file?
- Write a program that enumerates the paths of all files under a given directory and its subdirectories recursively.
- Write a program that copies the entire files and the directory structure of a given path to another specified path.

(3) Retrieving file attributes

- Stat system call https://linuxhint.com/stat-system-call-linux/
- What is a regular file? How one can check if a file is a regular file or not?
- How one can find the size of a file?

Command-line Interface

(1) Archive mode: produce a new file <archive-file-name> that aggregates all regular files in <target directory path>

```
star archive <archive-file-name> <target directory path>
```

(2) List mode: show the paths of all files aggregated in the

```
star list <archive-file-name>
```

(3) Extract mode

```
star extract <archive-file-name>
```

Format of archive file

An archive file is a sequence of pairs of a file header and file data. A header starts with an unsigned integer n with 4 bytes which represents the length of the file path. After that, a string of n bytes that represents the file path follows. After that, another unsigned integer m with 4 bytes is given which represents the size of the file data. This is the end of the file header. From the next bytes, m bytes of the file data follow. An archive has one pair of a header and file data for each original file. We can see that archiving is to convert each file to a pair of a header and file data and write the pairs in sequence. Listing is to retrieve all headers. Extracting is to construct the corresponding file for each pair in an archive file.