CS314, Fall '25 - Operating Systems Homework 3

1 Assignment

1. Write a C program that starts at a given directory and descends the file tree from that point recording the sizes of all the files it finds. When it is all done, it should print a histogram of the file sizes using a bin width specified as a parameter (e.g., with 1024, file sizes of 0 to 1023 go in one bin, 1024 to 2047 go in the next bin, etc.).

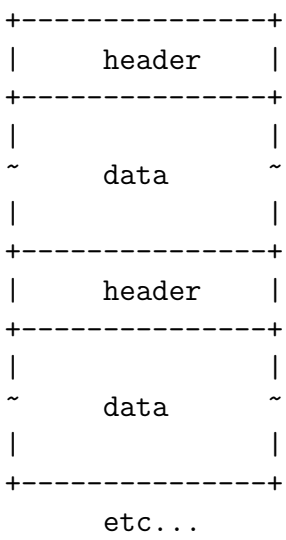
2. You've been using tar and zip to archive and turn in your assignments. Write an archiving utility in C that allows you to create an archive, and add, list, and extract files. Call it `simpletar`. It should have the following capabilities (`myarchive` and `somefile` are just example names):

   (a) `./simpletar -a myarchive somefile`
   Simpletar should add (hence the `-a`) the file `somefile` to the archive file `myarchive`. Subsequent calls to add to the same archive should result in the additional files being added to the archive.

   (b) `./simpletar -l myarchive`
   Simpletar should list only the names of the files stored in the specified archive.

   (c) `./simpletar -e myarchive somefile`
   Simpletar should extract the named file from the archive, creating a new file with the same name and containing the exact same data as the original and stored files. Careful not to overwrite something you can't live without. If a file is not specified, extract everything.

   The archive should follow the format given on the left, using at least the minimal header given on the right:




```
struct header{
  char name[256];
  int size;
};
```

3. (Practice question) A UNIX file system has 4096-byte blocks (this is typical) and 4-byte disk addresses. What is the maximum **file size** if i-nodes contain 12 direct block entries, and one single, double, and triple indirect entry each?

4. (Practice question) Consider having two files, A and B. Both are 2500 bytes long and disk blocks are 512 bytes each. The first block of A is in block 3 and the first block of B is in block 4. The remaining blocks of A are stored in subsequent odd-numbered blocks and the remaining blocks of B are stored in subsequent even-numbered blocks. Show the contents of a file allocation table for blocks 0-15.