```
#include <stdio.h>
#include <stdlib.h>
#include <strings.h>
#include <string.h>
#include "dos2sd.h"

/*
 * Name:Yuxian Wang
 * Student No:215170418
 * Date: 22/9/2017
 * This is my solution to Lab3 of eecs2031 in oder to read the disk, then list out the
sequence of sectors
 * that make up the file as well as the total number of bytes in the file.
 */

static void listFiles(FILE *fd, struct ATRSSDISK *disk, char fileName[])
{
  int sector, entry, i, count, start, baseFileNumber, temp, size;
  char name[9], ext[4];
  char name_ext[13];

  baseFileNumber = 0;
  for(sector=361;sector<=368;sector++) {
    for(entry=0;entry<ATR_SECTOR_SIZE;entry+=16) {
      if(disk->sector[sector-1][entry] == 0x042) {
        for(i=0;i<8 && disk->sector[sector-1][entry+5+i] != ' ';i++)
          name[i] = disk->sector[sector-1][entry+5+i];
        name[8] = '\0';
        for(i=0;i<3;i++)
          ext[i] = disk->sector[sector-1][entry+13+i];
        ext[3] = '\0';
        count = disk->sector[sector-1][entry+1]|disk->sector[sector-1][entry+2]<<8;
        start = disk->sector[sector-1][entry+3]|disk->sector[sector-1][entry+4]<<8;

        sprintf(name_ext,"%s.%s", name, ext);
        if(!strcmp(name_ext, fileName)){
                fprintf(fd, "%s.%s sector List ", name, ext);
                size = 0;
                temp = start;
                for(i=start;i<count+start;i++){
                        fprintf(fd, "%d ", temp++);
                        size += disk->sector[i-1][ATR_SECTOR_SIZE - 1];
                }
                printf("Total file size %d\n", size);
        }
      }
      baseFileNumber++;
    }
  }
}


int main(int argc, char *argv[])
{
  struct ATRSSDISK *disk;
  char *fileName;

  if(argc != 3) {
    fprintf(stderr,"usage: %s disk\n", argv[0]);
    exit(1);
  }
  if((disk = readDisk(argv[1])) == (struct ATRSSDISK *)NULL) {
    fprintf(stderr,"Unable to read disk %s\n", argv[1]);
```

```
    exit(1);
  }
  fileName = argv[2];
  listFiles(stdout, disk, fileName); /* put it in atari offset notation 1..720 */
  freeDisk(disk);
  return 0;

}
```