File Systems Exercise 2

Due Friday by 10pm   **Points** 0.5

Introduction

This exercise follows on from File System Exercise 1. Note that it is due on the same day as Exercise 8**If you wait until the deadline to do this exercise it is unlikely that you will be able to finish assignment 4.**

You may work in pairs for this exercise, with your A4 partner, since some code will potentially be common with the assignment. MarkUs will only create the appropriate directory in your repository when you log into MarkUs and either create your group, or declare that you will work alone. The groups will get a new shared repository, and the students working solo may also get a new repository. Please log into MarkUs well before the deadline to take these steps. (If you create the directory in your repo yourself, then MarkUs won't know about it and we won't be able to see your work.)

It is your responsibility to log into MarkUs *before* the exercise deadline to ensure that you know where to commit your work, and so that MarkUs can connect your work to the grading system.

Requirements

Using the same code you wrote in file system exercise 1, add the code to print out the inode and block bitmaps, and the inodes. This will take a little more code writing than file system exercise 1.

For `emptydisk.img` your output should *exactly* match the following. In other words, we should be able to use `diff` to compare your output to this and see that it is identical (the indentation below uses 4 spaces).

For `emptydisk.img` your output should *exactly* match the following. In other words, we should be able to use `diff` to compare your output to this and see that it is identical (the indentation below uses 4 spaces).

```
Inodes: 32
Blocks: 128
Block group:
    block bitmap: 3
    inode bitmap: 4
    inode table: 5
    free blocks: 105
    free inodes: 21
    used_dirs: 2
Block bitmap: 11111111 11111111 11111100 00000000 00000000 00000000 00000000 00000000 00000000 00000000
00000000 00000000 00000000 00000000 00000000 00000001
Inode bitmap: 11111111 11100000 00000000 00000000

Inodes:
[2] type: d size: 1024 links: 3 blocks: 2
[2] Blocks:  9
```

For `onefile.img` your output should match the following:

```
Inodes: 32
Blocks: 128
Block group:
    block bitmap: 3
    inode bitmap: 4
    inode table: 5
    free blocks: 104
    free inodes: 20
    used_dirs: 2
Block bitmap: 11111111 11111111 11111100 00000000 00100000 00000000 00000000 00000000 00000000 00000000
00000000 00000000 00000000 00000000 00000000 00000001
Inode bitmap: 11111111 11110000 00000000 00000000

Inodes:
[2] type: d size: 1024 links: 3 blocks: 2
[2] Blocks:  9
[12] type: f size: 128 links: 1 blocks: 2
[12] Blocks:  35
```

Submission

Ensure that you have a repository directory created by MarkUs for this exercise (fs2), then add, commit, and push `readimage.c` and `ext2.h` to this directory.