Task: Long File names

Carmi Merimovich

An `xv6` directory is a file composed of sequence of records of the following form.

```
#define DIRSIZ 14

struct dirent {
  ushort inum;
  char name[DIRSIZ];
};
```

We aim at allowing long filenames, by which we mean that a filename can have up to 255 characters. Of course having a fixed length record, as above, with such a huge place for filenames, is not a good idea. After all, most file names will be quite short. So, a variable length record is to be used. Here is a suggestion for a record structure.

```
#define DIRMAXSIZ 255

struct dirent {
  ushort inum;
  uchar  namelen;
  char name[DIRMAXSIZ];
};
```

It is very important to note: In memory, the `name` field can be of the maximal length. *However*, on disk, only `namelen` bytes should be stored!

Points to consider.

1. Keep the convention that `inum == 0` means the record is free.
2. The field `namelen` is valid also for free records. Without it we wouldn't have know where the next record begins.
3. Unlike the fixed record case, we can have a free entry which is not good for us. I.e., if `namelen` is less than the length of the filename we need to store.

4. When searching for a free record, you *must* use the first viable free record, i.e., a free record with a large enough `namelen`.

5. Upon finding an empty record one of the following cases can happen.

   (a) `namelen` is the length of the filename to be store. In this case, keeping the record structure is enough.

   (b) `namelen` is considerably larger than the length of the filename to be stored. In this case the record should be broken into two records. The first one will have the new information. The second one will be free. The meaning of considerable is: 4 bytes larger. The reason for this, is that below 4 bytes of extra storage, we can't really have a new empty record.

   (c) If neither of the above hold, then the record structure should be kept, and null bytes should be stored into the extra bytes of the `name` field.

6. Do observe: You have to fix the `mkfs` program! Of course, there are other utilities which you need to change in order check your changes. E.g., `ls`.