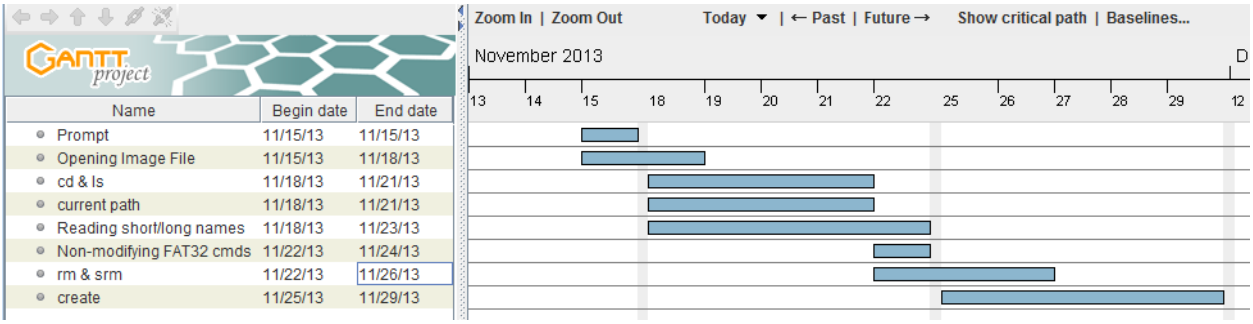
Corey Owens
Eric Daniels

Project 3:

File System Project – FAT32 Editor

Table of Contents

Project Timeline



Name	Begin date	End date
Prompt	11/15/13	11/15/13
Opening Image File	11/15/13	11/18/13
cd & ls	11/18/13	11/21/13
current path	11/18/13	11/21/13
Reading short/long names	11/18/13	11/23/13
Non-modifying FAT32 cmds	11/22/13	11/24/13
rm & srm	11/22/13	11/26/13
create	11/25/13	11/29/13

Nov 07, 2013

A Github repository was set up for the project. A test makefile was committed. A readme was made.

Contributors: Eric Daniels

Nov 18, 2013

Updated makefile. Added selection of commands and printing prompt. Implemented opening of image file. Created a FAT32 class for abstraction and usability. Made stubs for all commands. Made sure no public facing command can work if the FAT32 hasn't been initialized.

Contributors: Eric Daniels, Corey Owens

Nov 18, 2013

Simplified things by opening the file in main.

Contributors: Eric Daniels

Nov 21, 2013

Implemented a basic change directory and ls. Made structs packed and memcpy instead. Worked on a few small bugs. Working on reading short and long names. Change directory completed. Showing current path completed as well.

Contributors: Eric Daniels, Corey Owens

Nov 23, 2013

Small changes to makefile to simplify things. All non-modifying FAT32 commands implemented. Updated documentation. Fixed several bugs.

Contributors: Eric Daniels

Nov 24, 2013

Switched back to using pre-C++ 11. Wasn't properly calculating free clusters, fixed. Implemented rm and srm. Added a todo for size.

Contributors: Eric Daniels

Nov 25, 2013

Cleaned up code. Worked quite a bit on create. Documented code.

Contributors: Eric Daniels, Corey Owens

Nov 28, 2013

Create implemented. All functions completed and implemented. Debugging in progress.

Contributors: Eric Daniels, Corey Owens

Major Problems

A majority of the problems we faced were due to ambiguities in the FAT specifications. They were, at times, very vague and would refer to topics with incompletion such as the 8.3 naming conventions. Also, the lack of examples in the specification made it exceedingly difficult to implement without any reference material. Other than that, balancing the abstraction of the program between readability and performance was quite difficult.

Knowledge Gained

Through working on the program, we formed a better understanding of hex editing tools, how file systems employ different methods of storage, (de)serialization of data, retrieval of files, and reading files consisting of hex. I believe this project was much more intuitive than the previous ones, but also took a good chunk of time. The TA's slides were also very useful in understanding file systems. Having the FAT specification to read through and follow saved time normally spent desperately searching through the web for something related.