

Prof. Bruno Silvestre

1. DESCRIÇÃO

O trabalho consiste em desenvolver um sistema de arquivos em modo usuário que permite:

1. Caminhar pelos diretórios de uma imagem FAT16.
2. Ver os atributos dos arquivos e diretórios.
3. Copiar um arquivo de qualquer diretório de dentro da imagem para fora da imagem.

Deve-se utilizar a biblioteca FUSE para criar o programa “*mount_fat16*” que proverá ao sistema Linux acesso à imagem. Por exemplo, para montar a imagem no diretório:

```
./mount_fat16 <diretorio>
```

Para desmontar, basta digitar o comando:

```
fusermount -u <diretorio>
```

As informações sobre a especificação da FAT16 podem ser encontradas no Moodle.

Será disponibilizado um *template* de código para facilitar o processo de compilação.

2. CRIANDO UMA IMAGEM FAT16

Usaremos o ambiente Linux para criar imagens que possam ser utilizadas em testes do programa, mais especificamente, usaremos as ferramentas:

- **dd**: para converter ou copiar arquivos.
- **mkfs.msdos**: para formatar o arquivo como FAT16.

Para criar a imagem, podemos seguir os passos:

1. Gerar um arquivo vazio com 30Mb de tamanho:
dd if=/dev/zero of=fat16.img bs=1M count=30
2. Formatar o arquivo como FAT16:
mkfs.msdos -F 16 fat16.img

2.1. MANIPULANDO A IMAGEM

Para a realização do trabalho é necessário criar diretórios ou copiar arquivos para imagem. Para isso nós podemos montar essa imagem em um diretório, criar diretórios ou copiar arquivos, e finalmente desmontar a imagem.

O Linux se encarrega de tratar a imagem como se fosse um disco normal. Os passos são:

- Criar um diretório para ser o ponto de montagem no sistema de arquivos
mkdir temp

- Montar a imagem no diretório criado (só *root* pode montar, usaremos o *sudo*)
sudo mount -t msdos -o loop fat16.img temp
- Criar subdiretórios
sudo mkdir temp/dir
sudo mkdir temp/dir/dir
- Copiar os arquivos
sudo cp arquivo.txt temp
sudo cp outro.txt temp/dir/dir
- Desmontar a imagem:
sudo umount temp

Caso deseje modificar o conteúdo da imagem, basta repetir o processo acima: montar, copiar/remover/criar diretório, desmontar, ou seja, não é necessário criar outra imagem do zero.

Note que é necessário usar o comando *sudo* para montar e desmontar a imagem, bem como para manipular o conteúdo. Como é montada com acesso somente para o *root*, a manipulação no diretório só é permitida pelo *root* (i.e., deve-se usar o comando *sudo* para copiar, apagar ou renomear os arquivos).

ATENÇÃO:

- Não utilize o seu programa em uma imagem que está montada porque o Linux faz cache das informações em memória. Sempre desmonte a imagem

3. REGRAS

- **Data de Entrega: 28 de maio de 2017, 23:59.**
- Trabalho deve ser feito em grupo de 2 alunos.
- Trabalho só será considerado válido após o grupo realizar a apresentação ao professor.
 - O integrante do grupo que não comparecer à apresentação terá nota zero.
- Qualquer acesso de leitura ou gravação deve ser feito por meio das funções *sector_read* e *sector_write*, caso contrário todo o trabalho será considerado inválido. O grupo deve implementar essas funções.
- O grupo deve utilizar o Github para desenvolvimento e entrega do trabalho.
 - O professor criará e compartilhará com cada grupo um repositório.
 - *Commits* fora do prazo serão ignorados.
- O grupo deve produzir um relatório descrevendo como os principais pontos do trabalho foram realizados, ou seja, abordagens utilizadas para atingir os objetivos do trabalho.
 - O documento deve ser em formato PDF e o arquivo deve estar no repositório, no diretório “doc”.
- Boas práticas de programação também serão consideradas na avaliação do trabalho:
 - Ex: comentários, qualidade e organização do código, etc.
- Plágio (“cola”) total ou parcial implica em nota zero para todos os envolvidos, sendo passível de processo administrativo na UFG.