

INF01112 - Arquitetura e Organização de Computadores II (4)
Trabalho T2 e Recuperação do Trabalho T1 -- Turmas A e B
Light-FS

Formação dos grupos

- É recomendado que o trabalho seja executado por grupos de 3 ou 4 alunos. Grupos de 5 ou mais alunos **não serão tolerados**. Não é recomendada a execução do trabalho por grupos de 2 ou menos alunos, pois o projeto demanda tempo e organização!
- Grupos poderão ser formados com alunos de ambas as turmas. Neste caso, favor indicar o grupo no cabeçalho de **cada** arquivo do trabalho.
- Para os alunos que não entregaram o Trabalho T1, ou que queiram melhorar sua nota nessa avaliação, há uma atividade extra no final deste enunciado. Nesse caso, a nota do trabalho T2 substituirá a nota do trabalho T1, seja ela qual for. Não haverá recuperação para o trabalho T2.
- O trabalho deverá ser apresentado por ao menos um dos integrantes do grupo na data designada, **a combinar**. Não apresentar o trabalho acarreta em nota zero.

Enunciado

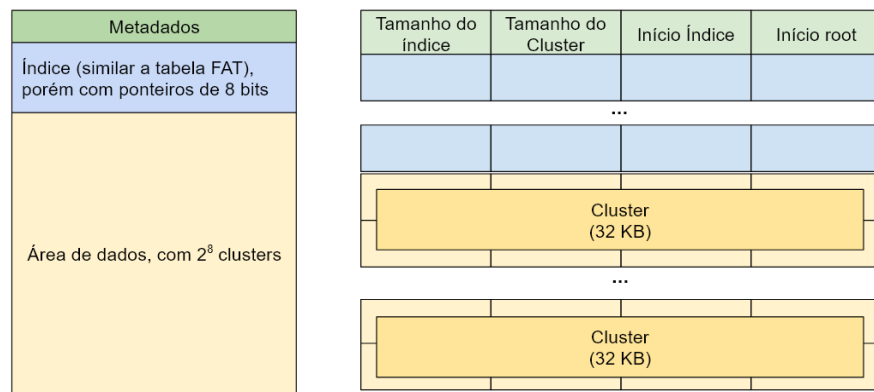
- Você e seu grupo são responsáveis pela implementação de um sistema de arquivos chamado Light-FS (*light filesystem*), criado unicamente para este trabalho e descrito ao decorrer deste documento.
- O sistema de arquivos deverá ser implementado na linguagem C ou C++. Outras linguagens **não** serão aceitas.
- O sistema de arquivos consiste de dois programas: (i) um programa que fará a gerência de um arquivo binário, onde serão armazenados os cluster de setores do sistema de arquivos e o índice do sistema de arquivos; e (ii) uma interface em console, que permitirá ao usuário manipular o sistema de arquivos. Recomenda-se essa separação em razão do tamanho do projeto.
- O programa de interface em console deverá implementar, **necessariamente**, todos os comandos mostrados na seção *comandos*, respeitando a estrutura de arquivo mostrada na seção *arquivo de dados*.

Comandos

Comando	Descrição
CD (<i>change directory</i>)	Permite ao usuário mudar de diretório. Ao iniciar o programa, o usuário deverá estar posicionado no diretório raiz, que deverá se chamar "root". Para navegar entre diretórios, o usuário precisará informar o caminho do diretório de destino, ex. CD /root/pasta1/pasta2. Para facilitar a implementação, não serão admitidos espaços nos nomes de diretórios e arquivos
DIR (<i>directory</i>)	Mostra uma lista com os arquivos e diretórios que estão dentro do diretório atual, um abaixo do outro. Para um diretório vazio, deverá mostrar "<vazio>".
RM (<i>remove</i>)	Deleta (remove do sistema de arquivos) um arquivo ou diretório. O diretório precisa estar vazio para ser removido. Ex.: RM /root/pasta1/pasta2/arquivo.txt RM /root/pasta3
MKDIR (<i>make directory</i>)	Cria um novo diretório dentro do diretório atual. Ex: MKDIR pasta1
MKFILE (<i>make file</i>)	Cria um novo arquivo dentro do diretório atual. Neste trabalho, apenas arquivos de texto poderão ser criados (embora sejam armazenados na forma binária).Ex: MKFILE arquivo.txt
EDIT	Modifica um arquivo, substituindo o texto atual pelo texto informado. Ex.: EDIT /root/arquivo.txt "novo conteúdo do arquivo"
MOVE	Move um arquivo ou diretório para dentro de um diretório dentro do sistema de arquivos. O nome do arquivo/diretório e diretório de destino deverão ser informados. Ex.: MOVE /root/arquivo.txt /root/pasta1 MOVE /root/pasta2 /root/pasta3
RENAME	Renomeia um arquivo ou diretório. Ex.: RENAME /root/arquivo.txt laranjas.txt RENAME /root/pasta1 pasta2

Arquivo de Dados

- O arquivo de dados deverá possuir a seguinte estrutura:



- Os primeiros 32 bits do arquivo deverão conter os metadados do sistema de arquivos
 - Tamanho do índice (quantas entradas o índice possui, utilizar o valor 2⁸)
 - Tamanho do cluster (utilizar o valor 32KB)
 - Byte onde o índice inicia (metadados iniciam no byte zero e vão até byte 3)
 - Byte onde inicia o primeiro cluster
- O índice deve ser posicionado logo após (abaixo) os metadados. Cada entrada do índice possui 8 bits. Os valores armazenados em cada entrada correspondem aos valores do sistema FAT32 (ponteiro, end-of-file, vazio, corrompido).
- Os clusters são posicionados logo abaixo do índice e possuem 32KB de tamanho cada. A estrutura interna dos clusters é de livre escolha do grupo.

Avaliação

- O trabalho vale 10 e cada operação (descritas acima) vale até 1 ponto, dependendo da forma como foi implementada (capricho, funcionalidades, atenção aos itens do enunciado), num total de 8 pontos.
- 1 ponto será creditado aos trabalhos que apresentarem boas práticas de programação: (i) domínio das funções de manipulação de arquivos, fseek, fread, (ii) organização do código, indentação, comentários, utilização correta das estruturas de dados, e (iii) encapsulamento da interface e do programa manipulação do arquivo.

- 1 ponto será creditado pela apresentação (clareza, domínio do tema, organização).

Atividade Extra (substitui a nota do T1)

Implemente os seguintes comandos em adição aos comandos do trabalho T2:

RF (<i>remove -force</i>)	Deleta (remove do sistema de arquivos) um arquivo ou diretório. Se o diretório possui outros diretórios ou arquivos, deverá ser executada a remoção em cascata. Ex.: RF /root/pasta1/pasta2/arquivo.txt RF /root/pasta3
DISK	Mostra quanto de espaço está ocupado no arquivo (em KB) pelo diretório atual e todos seus arquivos e subdiretórios. Pode ser implementado juntamente com o comando DIR, porém deve mostrar também a soma dos itens.
DEFRAG	Deve reorganizar o índice de forma que todos os cluster ocupados estejam no início da área de dados do arquivo. Para isso, deve mover os arquivos e pastas existentes. Ao final da operação, mostrar quantos arquivos/diretórios foram movidos.