

Trabalho Prático 2 Sistemas de arquivos

Dado um simulador que permite operar sobre um sistema de arquivos de um S.O. hipotético, assim como a API de tal S.O. para operações com discos, i-nodes e sistemas de arquivos (de forma genérica), implemente seu próprio sistema de arquivos. Todo o código-fonte encontra-se escrito em linguagem C e é composto pelos seguintes arquivos:

util.h – Arquivo de cabeçalho para declaração de funções úteis de conversão de valores
util.c – Arquivo que implementa funções de conversão entre inteiros (uint) e bytes (char)
disk.h – Arquivo de cabeçalho para a declaração da API de acesso e manutenção a discos
disk.c – Arquivo que implementa a API de acesso e manutenção a discos
inode.h – Arquivo de cabeçalho para a declaração da API de manutenção de i-nodes
inode.c – Arquivo com a implementação da API de manutenção de i-nodes
vfs.h – Arquivo de cabeçalho para a declaração da API genérica de sistemas de arquivos
vfs.c – Arquivo que implementa a API genérica de sistemas de arquivos
main.c – Arquivo do programa principal do simulador
~~llformat.c~~ – ~~Arquivo do programa para formatação de discos em baixo nível~~

Todo o seu desenvolvimento deve estar contido em novos arquivos de código-fonte, fazendo uso apenas das APIs disponibilizadas nos arquivos de cabeçalho, além é claro, da API padrão C.

Toda a descrição das APIs está contida nos respectivos arquivos de cabeçalho.

As informações e funções a serem implementadas possuem explicação na estrutura de informações sobre sistemas de arquivo (FSInfo), definida em **vfs.h**:

```
typedef struct fs_info {
    char fsid;      // Identificador do tipo de sistema de arquivos
    char *fsname;   // Nome do tipo de sistema de arquivos
    int (*isidleFn) (Disk *d);
    int (*formatFn) (Disk *d, unsigned int blockSize);
    int (*openFn) (Disk *d, const char *path);
    int (*readFn) (int fd, char *buf, unsigned int nbytes);
    int (*writeFn) (int fd, const char *buf, unsigned int nbytes);
    int (*closeFn) (int fd);
    int (*opendirFn) (Disk *d, const char *path);
    int (*readdirFn) (int fd, char *filename, unsigned int *inumber,
                     unsigned int enumber);
    int (*linkFn) (int fd, const char *filename, unsigned int inumber);
    int (*unlinkFn) (int fd, const char *filename);
    int (*closedirFn) (int fd);
} FSInfo;

//Funcao para instalar seu sistema de arquivos no S.O.
//(inclua a declaração em um de seus arquivos de cabeçalho *.h)
int installMyFS ( void );
```

Quaisquer funções auxiliares não podem ser externalizadas em novos arquivos de cabeçalho. Identificadores, tipos, assinaturas de função, enfim, linhas já escritas de código, não podem ser modificados.

Quaisquer funções necessárias e não implementadas nas APIs de disco, de i-node e de sistema de arquivos devem ser discutidas e solicitadas ao professor por e-mail. O mesmo vale para possíveis erros detectados.