



FACULDADE DE INFORMÁTICA DE PRES. PRUDENTE
SISTEMA DE INFORMAÇÃO

BRUNO ALISON DA SILVA MELO
GABRIEL BARBOSA SALVADOR
JOÃO PEDRO NARCIZO FRANÇA
JOÃO VITOR DIAS DA SILVA

IMPLEMENTAÇÃO DE ARQUIVOS
I-NODE

Presidente Prudente - SP
2023

Sumário

1. Introdução.....	3
2. Implementação.....	3
3. Estruturas Utilizadas.....	5
4. Instruções do Sistema.....	12
5. Relatórios.....	26
6. Conclusão.....	28

1. Introdução

Neste presente trabalho, iremos explorar a implementação de arquivos Inode em sistemas Linux/Unix. Os Inodes são componentes fundamentais para o funcionamento desses sistemas, governando a organização e o gerenciamento de arquivos. Além disso, abordaremos a implementação de diretórios, arquivos, links e a criação de um disco simulado (vetor com tamanho escolhido pelo usuário). Os blocos serão monitorados em uma lista de blocos livres. Este trabalho oferece uma visão profunda sobre os sistemas de arquivos Unix/Linux e sua estrutura interna.

2. Implementação

A implementação será conduzida em linguagem de programação C++ para representar o sistema de arquivos I-node. Inicialmente, será empregada uma estrutura de dados estática, ou seja, um vetor, para simular a memória secundária, que equivale ao disco. Cada posição nesse vetor terá a capacidade de armazenar informações, como arquivos, diretórios, I-nodes, I-nodes indiretos, links físicos e links simbólicos. Além disso, cada posição poderá representar um setor defeituoso no disco (bad sector), tornando-a inutilizável para o armazenamento de dados. Caso uma posição ocupada no disco se torne defeituosa, os dados nela contidos serão irremediavelmente perdidos.

Os blocos do disco, correspondentes a cada posição do vetor, terão dois estados possíveis: livres ou ocupados. Para gerenciar esse controle, será utilizada uma estrutura denominada lista ligada de blocos livres. Nesta lista, todos os blocos de disco disponíveis serão registrados como livres, e quando um bloco for requisitado, a lista de blocos livres fornecerá a posição que poderá ser utilizada. Se uma posição previamente ocupada for liberada, ela será reintegrada à lista de blocos livres, tornando-se disponível para futuras alocações. Caso o bloco retirado da lista de blocos livres se encontre no estado “bad”, será imediatamente descartado.

Os arquivos regulares serão simulados ocupando posições de blocos no disco, variando de acordo com o tamanho dos arquivos. Esse tamanho poderá ser definido pelo usuário ou, em alguns casos, será determinado pelo sistema. O tamanho do arquivo será representado ocupando posições no vetor de disco. Por sua vez, os arquivos de diretório ocuparão inicialmente um único bloco no disco, correspondendo a uma única posição no vetor. Um diretório possui entradas de diretório que é representado pelo nome do arquivo e o endereço do Inode relacionado àquele arquivo, a princípio essa entrada de diretório possui dez posições fixas, caso a entrada de diretório seja toda preenchida, será ocupado um outro espaço no disco.

As informações sobre a localização das posições de armazenamento dos arquivos no vetor de disco serão registradas na estrutura do I-node, especialmente nas seções dedicadas aos blocos diretos e indiretos, quando necessário. Dessa forma, o sistema

será capaz de rastrear eficientemente a alocação de espaço para arquivos regulares e diretórios, garantindo que a estrutura do disco virtual seja gerenciada com eficiência e precisão

A estrutura do I-node é fundamental, uma vez que representa um conjunto de dados que define um sistema de arquivos. Esses dados desempenham um papel crucial na obtenção de informações essenciais sobre arquivos ou diretórios no sistema de

arquivos. A estrutura do I-node abrange uma gama significativa de informações, incluindo:

Proteção: Este atributo define qual é a estrutura que está sendo referenciado, podendo ser um diretório, um link ou um arquivo regular, as permissões de acesso, determinando quais permissões são concedidas a usuários, grupos e outros. As permissões podem incluir leitura, gravação, execução ou pode estar desabilitado.

Contador de Links Físicos: Esse valor indica o número de entradas no diretório que apontam para o mesmo Inode. Ele é utilizado para controlar as ligações físicas entre os arquivos e diretórios.

Proprietário: Este atributo especifica o usuário que é o proprietário do arquivo ou diretório.

Grupo: Indica o grupo responsável pelo acesso ao qual o arquivo ou diretório pertence.

Data de Criação, Último Acesso e Última Alteração: Esses registros fornecem informações importantes sobre a história do arquivo, incluindo quando ele foi criado, acessado pela última vez e modificado pela última vez.

Tamanho Original do Arquivo em Bytes: Este atributo determina o tamanho inicial do arquivo, medido em bytes.

Após esses atributos, a estrutura do I-node inclui os chamados 'blocos diretos'. São cinco posições que apontam para os blocos correspondentes no disco (representados no vetor). No entanto, caso esses blocos diretos não sejam suficientes para armazenar o arquivo ou diretório em questão, utilizamos uma estrutura adicional chamada 'bloco indireto'. Existem três tipos de blocos indiretos:

Bloco Indireto Simples: Adiciona oito posições extras para armazenamento de blocos no disco.

Bloco Indireto Duplo: Estende a capacidade de armazenamento ao adicionar níveis que permite o acesso a mais blocos.

Bloco Indireto Triplo: Caso as duas estruturas anteriores não sejam suficientes, o bloco indireto triplo oferece uma extensão ainda maior ao sistema, fornecendo acesso a uma quantidade significativa de blocos no disco. Caso essa estrutura ainda não atenda todos os blocos que representam o arquivo, é criada uma referência para um outro Inode.

No sistema de arquivos em questão, oferecemos suporte à utilização de links para arquivos e diretórios, permitindo que os usuários tirem proveito de duas variantes de links:

Link Físico: Essa modalidade de link estabelece uma referência direta ao bloco I-node do arquivo. Em outras palavras, os blocos são associados a uma pequena estrutura de dados conectada ao arquivo, enquanto os diretórios apontam apenas para essa estrutura de dados. Isso significa que vários usuários podem compartilhar o mesmo arquivo através dessa ligação direta, proporcionando um mecanismo eficaz de compartilhamento de recursos.

Link Simbólico: Por outro lado, os links simbólicos oferecem um atalho direto para um arquivo ou diretório, representando uma maneira conveniente de acessar rapidamente recursos no sistema. Eles são, na verdade, caminhos simples que apontam para o local real do arquivo ou diretório, fornecendo uma forma prática de navegação no sistema de arquivos.

3. Estruturas Utilizadas

Para representar o sistema de arquivos, adotamos uma abordagem baseada em uma estrutura estática em forma de vetor. Essa estrutura será responsável por representar todos os blocos que compõem um disco virtual. O tamanho dessa estrutura, ou seja, a quantidade de blocos disponíveis, será escolhido pelo usuário, permitindo flexibilidade na definição do espaço de armazenamento disponível.

Disco

R	R	F	F	A	B	F	B	A	B	A	B	A	F	A
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

Cada posição no disco pode conter as seguintes estruturas: Inode, Inode Indireto, Diretório, Lista de Blocos Livres, Link Simbólico e um inteiro representando um setor defeituoso (bad)

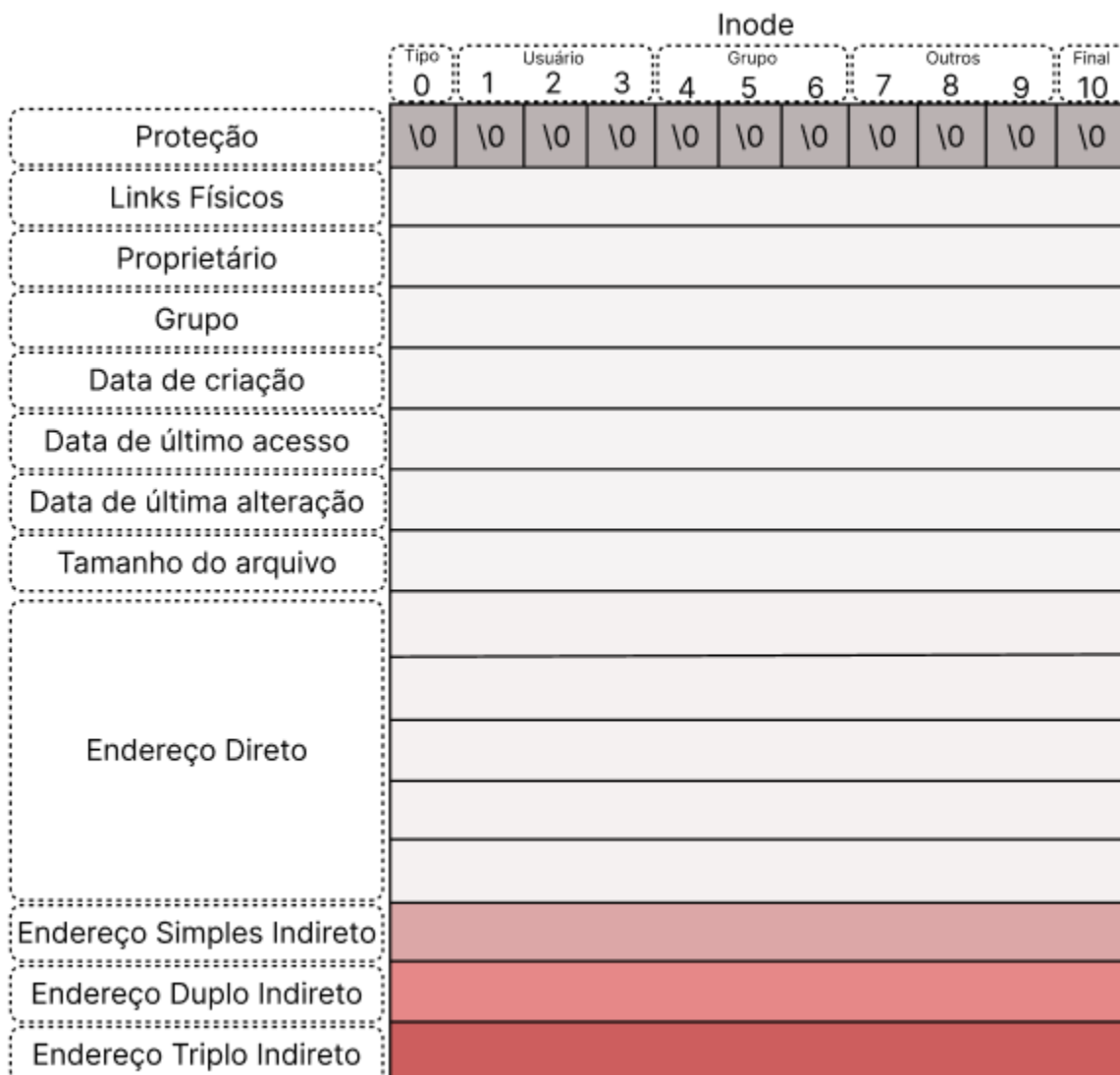
```

struct Disco
{
    // identificador se o disco está como bad ou não
    int bad;

    INode inode;
    INodeIndireto inodeIndireto;
    Diretorio diretorio;
    ListaBlocoLivre lbl;
    LinkSimbolico ls;
};

```

A estrutura do I-node desempenha um papel fundamental, representando um conjunto de dados que define integralmente o sistema de arquivos. Esta estrutura abriga atributos essenciais de um arquivo, incluindo permissões (Proteção), contador de links físicos, proprietário, grupo, registros de data (Criação, Último Acesso e Última Alteração) e tamanho original do arquivo em bytes. Além disso, o I-node também registra a localização física dos arquivos no disco, utilizando bloco direto e uma estrutura conhecida como I-node Indireto (com modalidades Simples, Duplo e Triplo), permitindo um gerenciamento eficiente do armazenamento dos dados



```

struct Inode
{ // Inode principal
    /*
        DEFINIÇÃO DO QUE O I-NODE ESTÁ APONTANDO:
        [0] 'd' = DIRETÓRIO, '-' = ARQUIVO, 'l' = LINK

        PERMISSÕES:
        [1] - [3] (rwx) -> OWNER
        [4] - [6] (rwx) -> GROUP
        [7] - [9] (rwx) -> OTHERS

        [10] - '\0'
    */
    char protecao[11];

    /*Contador do números de entradas no diretório que apontam para o i-node;*/
    int contadorLinkFisico;

    /*Código do proprietário do arquivo*/
    int proprietario;
    /*Código do grupo do arquivo*/
    int grupo;

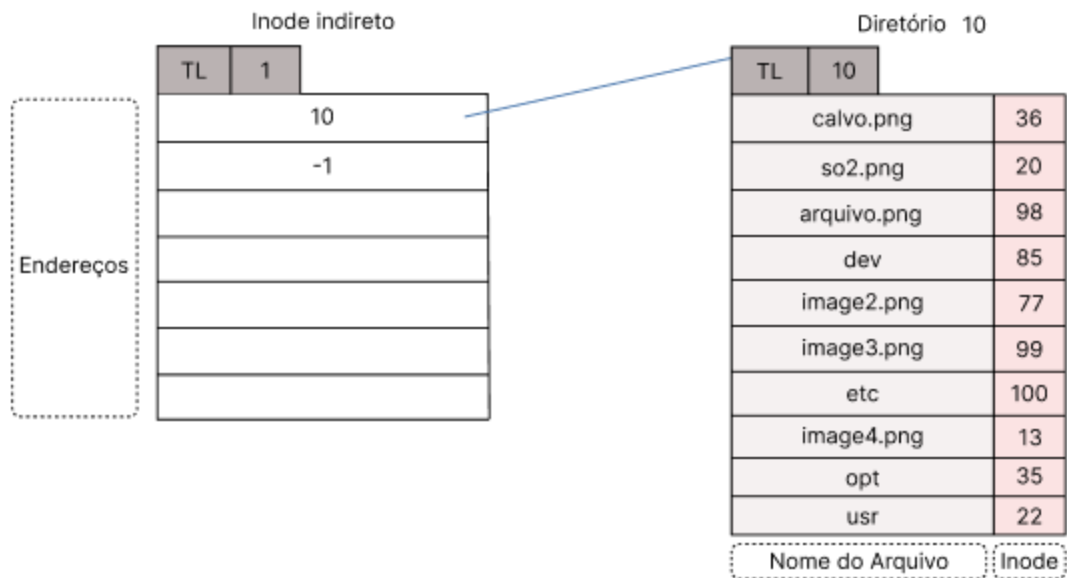
    /*data de criação do arquivo*/
    char dataCriacao[20];
    /*data de último acesso do arquivo*/
    char dataUltimoAcesso[20];
    /*data de última alteração do arquivo*/
    char dataUltimaAlteracao[20];

    /*Tamanho em bytes do arquivo apontado*/
    long long int tamanhoArquivo;

    /*Endereços de blocos que armazenam o conteúdo do arquivo */
    /*5 primeiros para alocação direta*/
    int enderecoDireto[5];
    /*6º ponteiro para alocação simples indireta*/
    int enderecoSimpleIndireto;
    /*7º para alocação dupla indireta*/
    int enderecoDuploIndireto;
    /*8º para alocação tripla-indireta*/
    int enderecoTriploIndireto;
};
typedef struct Inode Inode;

```

O I-node Indireto desempenha um papel crucial como uma extensão que possibilita o registro da localização física dos arquivos no disco. Essa estrutura oferece espaço adicional ao I-node, com a capacidade de armazenar até oito endereços de blocos do disco correspondentes ao arquivo.



```
struct INodeIndireto
{
    int endereco[MAX_INODEINDIRETO];
    int TL;
};
typedef INodeIndireto INodeIndireto;
```

A estrutura de um diretório é composta por entradas de diretório, cada uma com um tamanho de até dez entradas. Essas entradas podem armazenar informações sobre arquivos regulares ou diretórios, incluindo o nome do arquivo e o número do I-node que o representa. Quando as dez posições do diretório são preenchidas, o diretório alocará espaço adicional em outro bloco direto, e em caso de esgotamento dos blocos diretos, recorrerá aos blocos indiretos para acomodar mais entradas.

Diretório

TL	0		

Nome do Arquivo
Inode

```

struct Arquivo
{
    int enderecoINode;
    char nome[MAX_NOME_ARQUIVO];
};
typedef struct Arquivo Arquivo;

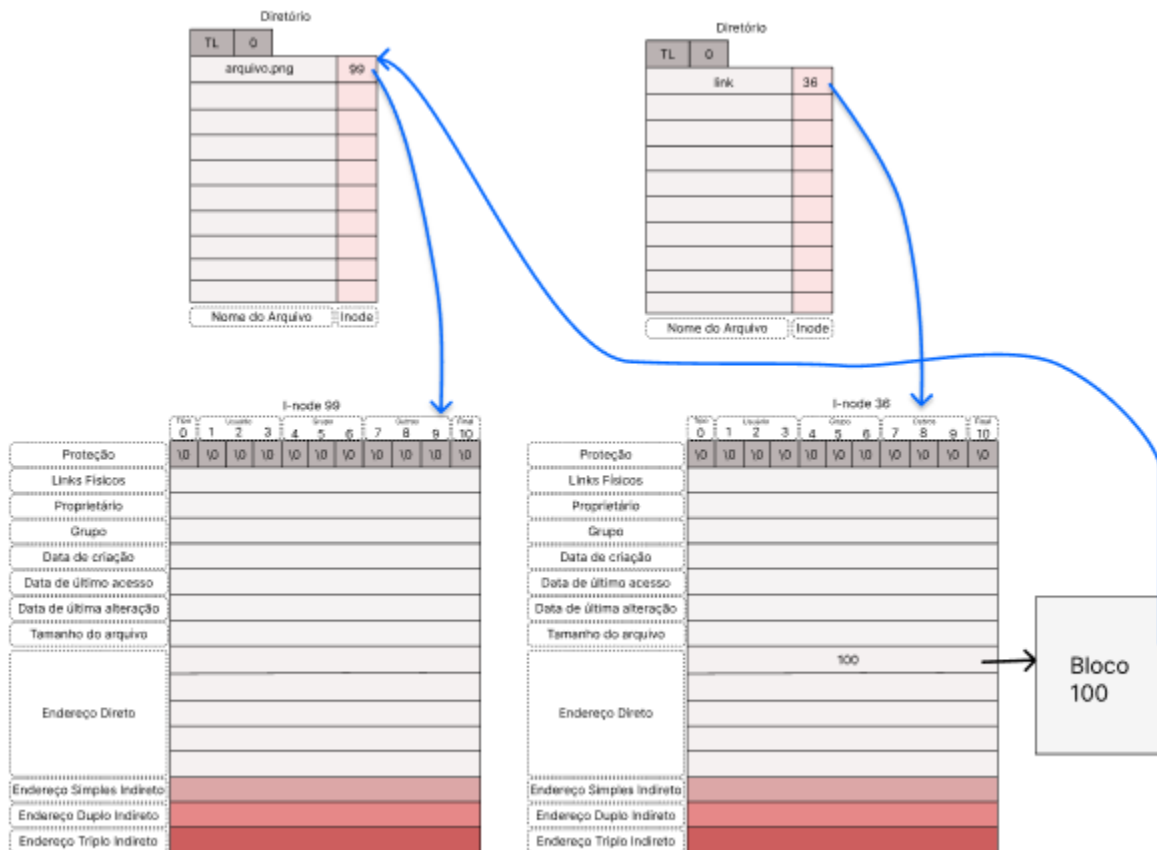
struct Diretorio
{
    Arquivo arquivo[DIRETORIO_LIMITE_ARQUIVOS];
    int TL;
};
typedef struct Diretorio Diretorio;

```

Link Simbólico (Soft Link)

Os links simbólicos oferecem um atalho direto para um arquivo ou diretório, representando uma maneira mais rápida de acessar os recursos no sistema. Eles são,

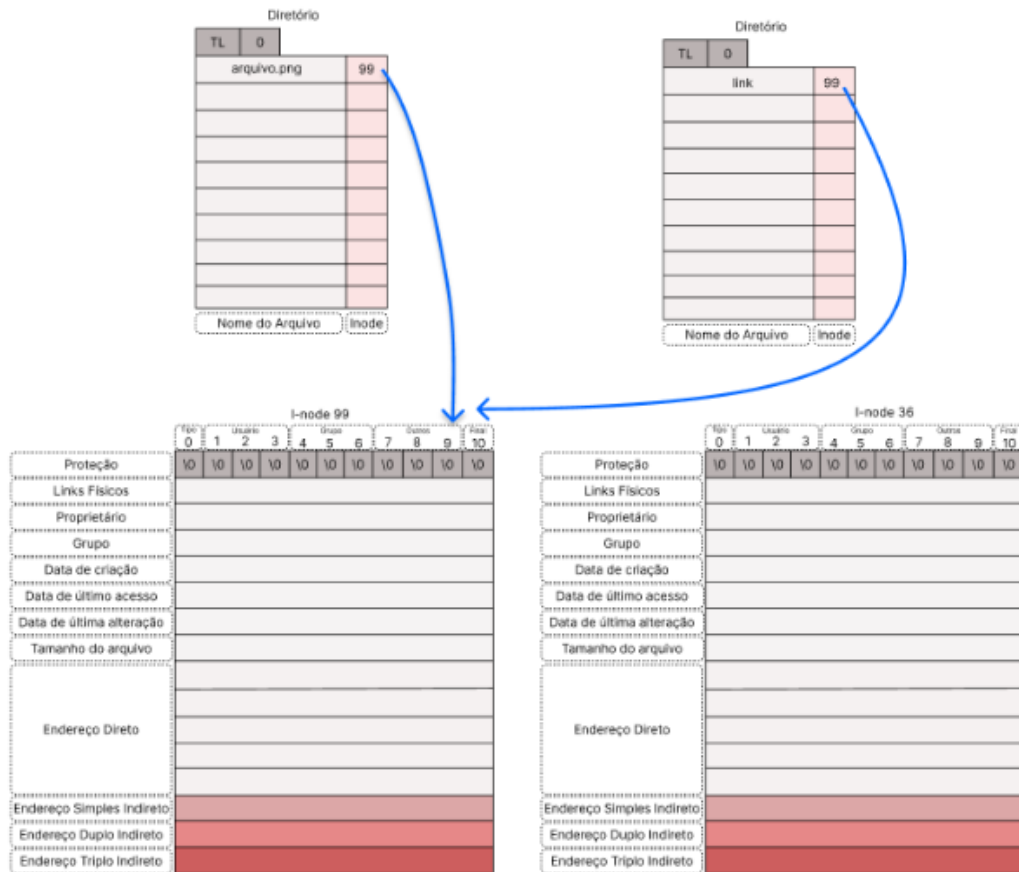
na verdade, caminhos simples que apontam para o local real do arquivo ou diretório, fornecendo uma forma prática de navegação no sistema de arquivos. A estrutura do link simbólico conterá uma string que representa o caminho de origem que será salvo deste link.



```
struct LinkSimbolico
{
    string caminho;
};
typedef struct LinkSimbolico LinkSimbolico;
```

Link Físico

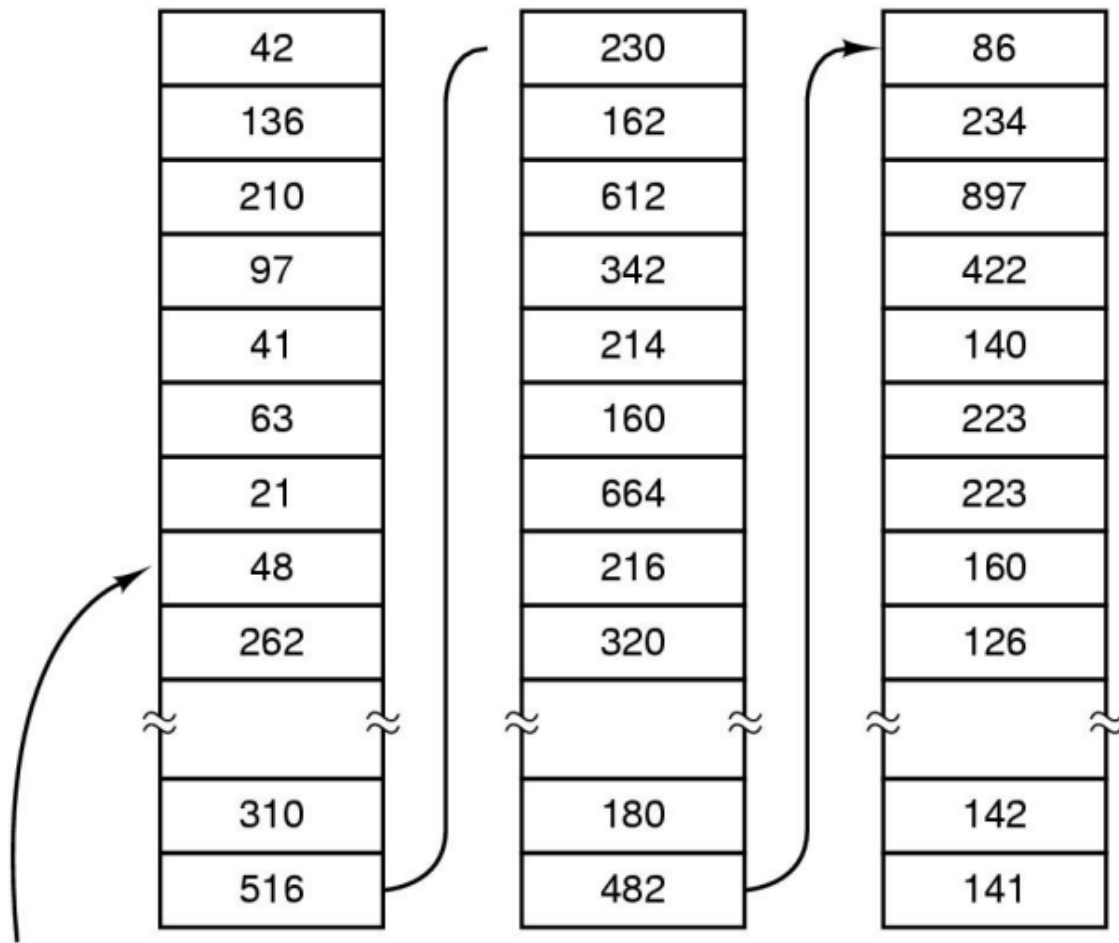
Essa modalidade de link estabelece uma referência direta ao bloco I-node do arquivo. Essa estrutura permite que em uma entrada de diretório do link de destino possa conter o mesmo número do Inode do arquivo de origem em que esta ação foi realizada.



Lista Ligada de Blocos Livres.

Para monitorar os blocos livres do disco, será utilizada uma estrutura de lista encadeada de blocos livres. Todos os blocos de disco disponíveis serão empilhados na lista, e quando um bloco for requisitado, a lista de blocos livres fornecerá a posição do topo que poderá ser utilizada. Se uma posição previamente ocupada for liberada, ela será empilhada novamente à lista de blocos livres, tornando-se disponível para futuras alocações.

FREE disk blocks: 10, 17, 18



```
struct ListaBlocoLivre
{
    int topo;
    int endereco[QUANTIDADE_LIMITE_ENDERECO_LISTA_BLOCO_LIVRE];
    // endereço do próximo bloco de pilha no disco
    int enderecoBlocoProx;
};
typedef struct ListaBlocoLivre ListaBlocoLivre;
```

4. Instruções do Sistema

Inicialização do Sistema:


```
Informe a Quantidade de blocos no disco desejada[ENTER p/ ignorar]:  
1000
```

Comandos Básicos:

- a. **chmod (+)(-)ugo RWX:** alterar as permissões de acesso a arquivos e diretórios

```
root@localhost:~$ ls -l  
-rw-r--r-- 1 root root 100 11/10/2023 02:24:06 arquivo.png  
-rw-r--r-- 1 root root 200 11/10/2023 02:24:12 arquivo.jpg  
-rw-r--r-- 1 root root 120 11/10/2023 02:24:25 arquivo.cpp  
-rw-r--r-- 1 root root 90 11/10/2023 02:24:42 vscode  
drwxr-xr-x 1 root root 1 11/10/2023 02:24:47 diretorio1  
drwxr-xr-x 1 root root 1 11/10/2023 02:24:51 diretorio2  
drwxr-xr-x 1 root root 1 11/10/2023 02:24:56 diretorio3
```

```
root@localhost:~$ chmod u+x arquivo.jpg  
root@localhost:~$ ls -l  
-rwxrwxrwx 1 root root 100 11/10/2023 02:24:06 arquivo.png  
-rwxr--r-- 1 root root 200 11/10/2023 02:24:12 arquivo.jpg  
-rw-r--r-- 1 root root 120 11/10/2023 02:24:25 arquivo.cpp  
-rw-r--r-- 1 root root 90 11/10/2023 02:24:42 vscode  
drwxr-xr-x 1 root root 1 11/10/2023 02:24:47 diretorio1  
drwxr-xr-x 1 root root 1 11/10/2023 02:24:51 diretorio2  
drwxr-xr-x 1 root root 1 11/10/2023 02:24:56 diretorio3
```

```
root@localhost:~$ chmod u-r arquivo.jpg
root@localhost:~$ ls -l
-rwxrwxrwx 1 root root 100 11/10/2023 02:24:06 arquivo.png
--wxr--r-- 1 root root 200 11/10/2023 02:24:12 arquivo.jpg
-rw-r--r-- 1 root root 120 11/10/2023 02:24:25 arquivo.cpp
-rw-r--r-- 1 root root 90 11/10/2023 02:24:42 vscode
drwxr-xr-x 1 root root 1 11/10/2023 02:24:47 diretorio1
drwxr-xr-x 1 root root 1 11/10/2023 02:24:51 diretorio2
drwxr-xr-x 1 root root 1 11/10/2023 02:24:56 diretorio3
```

```
root@localhost:~$ chmod 777 arquivo.png
root@localhost:~$ ls -l
-rwxrwxrwx 1 root root 100 11/10/2023 02:24:06 arquivo.png
-rw-r--r-- 1 root root 200 11/10/2023 02:24:12 arquivo.jpg
-rw-r--r-- 1 root root 120 11/10/2023 02:24:25 arquivo.cpp
-rw-r--r-- 1 root root 90 11/10/2023 02:24:42 vscode
drwxr-xr-x 1 root root 1 11/10/2023 02:24:47 diretorio1
drwxr-xr-x 1 root root 1 11/10/2023 02:24:51 diretorio2
drwxr-xr-x 1 root root 1 11/10/2023 02:24:56 diretorio3
```

```
root@localhost:~$ chmod 000 diretorio1
root@localhost:~$ ls -l
-rwxrwxrwx 1 root root 100 11/10/2023 02:24:06 arquivo.png
--wxr--r-- 1 root root 200 11/10/2023 02:24:12 arquivo.jpg
-rw-r--r-- 1 root root 120 11/10/2023 02:24:25 arquivo.cpp
-rw-r--r-- 1 root root 90 11/10/2023 02:24:42 vscode
d----- 1 root root 1 11/10/2023 02:24:47 diretorio1
drwxr-xr-x 1 root root 1 11/10/2023 02:24:51 diretorio2
drwxr-xr-x 1 root root 1 11/10/2023 02:24:56 diretorio3
```

```
root@localhost:~$ chmod o+rw diretorio1
root@localhost:~$ ls -l
-rwxrwxrwx 1 root root 100 11/10/2023 02:24:06 arquivo.png
--wxr--r-- 1 root root 200 11/10/2023 02:24:12 arquivo.jpg
-rw-r--r-- 1 root root 120 11/10/2023 02:24:25 arquivo.cpp
-rw-r--r-- 1 root root 90 11/10/2023 02:24:42 vscode
d-----rwx 1 root root 1 11/10/2023 02:24:47 diretorio1
drwxr-xr-x 1 root root 1 11/10/2023 02:24:51 diretorio2
drwxr-xr-x 1 root root 1 11/10/2023 02:24:56 diretorio3
```

- b. **vi**: nomeArquivo visualizar um arquivo regular

```
root@localhost:~$ vi arquivo.jpg
arquivo.jpg: 985-984-983-982-981-980-979-978-977-976-975-974-973-972-971-970-969-968-967-966-965-964-963-
962
Quantidade de enderecos: 24
```

- c. **ls**: listar os nomes dos arquivos no diretório

```
root@localhost:~$ ls
arquivo.png      arquivo.jpg      arquivo.cpp      vscode  diretorio1      diretorio2      diretorio3      a
rquivo2 linkfisico      linksimbolico
root@localhost:~$
```

- d. **ls -l**: listas os nomes dos arquivos com seus atributos

```
root@localhost:~$ ls -l
-rwxrwxrwx 1 root root 100 11/10/2023 02:24:06 arquivo.png
--wxr--r-- 1 root root 200 11/10/2023 02:24:12 arquivo.jpg
-rw-r--r-- 1 root root 120 11/10/2023 02:24:25 arquivo.cpp
-rw-r--r-- 1 root root 90 11/10/2023 02:24:42 vscode
d-----rwx 1 root root 1 11/10/2023 02:24:47 diretorio1
drwxr-xr-x 1 root root 1 11/10/2023 02:24:51 diretorio2
drwxr-xr-x 1 root root 1 11/10/2023 02:24:56 diretorio3
-rw-r--r-- 1 root root 8 11/10/2023 02:47:55 arquivo2
drwxr-xr-x 2 root root 1 11/10/2023 02:41:11 linkfisico
lrwxrwxrwx 1 root root 1 11/10/2023 02:53:46 linksimbolico -> diretorio1/diretorio2
```

- e. **mkdir NomeDir**: criar diretórios

```
root@localhost:~$ mkdir diretorio1
Diretorio criado
root@localhost:~$
```


f. **rmdir NomeDir**: deletar diretórios que estejam vazios

```
root@localhost:~$ ls -l
-rwxrwxrwx 1 root root 100 11/10/2023 02:24:06 arquivo.png
--wxr--r-- 1 root root 200 11/10/2023 02:24:12 arquivo.jpg
-rw-r--r-- 1 root root 120 11/10/2023 02:24:25 arquivo.cpp
-rw-r--r-- 1 root root 90 11/10/2023 02:24:42 vscode
d-----rwx 1 root root 1 11/10/2023 02:24:47 diretorio1
drwxr-xr-x 1 root root 1 11/10/2023 02:24:51 diretorio2
drwxr-xr-x 1 root root 1 11/10/2023 02:24:56 diretorio3
drwxr-xr-x 1 root root 1 11/10/2023 02:37:41 diretorio5
```

```
root@localhost:~$ rmdir diretorio 5
Nao foi possivel remover o diretorio
root@localhost:~$ rmdir diretorio5
Diretorio removido
root@localhost:~$ ls -l
-rwxrwxrwx 1 root root 100 11/10/2023 02:24:06 arquivo.png
--wxr--r-- 1 root root 200 11/10/2023 02:24:12 arquivo.jpg
-rw-r--r-- 1 root root 120 11/10/2023 02:24:25 arquivo.cpp
-rw-r--r-- 1 root root 90 11/10/2023 02:24:42 vscode
d-----rwx 1 root root 1 11/10/2023 02:24:47 diretorio1
drwxr-xr-x 1 root root 1 11/10/2023 02:24:51 diretorio2
drwxr-xr-x 1 root root 1 11/10/2023 02:24:56 diretorio3
```

g. **rm NomeArq**: deletar arquivos

```
root@localhost:~$ ls -l
-rwxrwxrwx 1 root root 100 11/10/2023 02:24:06 arquivo.png
--wxr--r-- 1 root root 200 11/10/2023 02:24:12 arquivo.jpg
-rw-r--r-- 1 root root 120 11/10/2023 02:24:25 arquivo.cpp
-rw-r--r-- 1 root root 90 11/10/2023 02:24:42 vscode
d-----rwx 1 root root 1 11/10/2023 02:24:47 diretorio1
drwxr-xr-x 1 root root 1 11/10/2023 02:24:51 diretorio2
drwxr-xr-x 1 root root 1 11/10/2023 02:24:56 diretorio3
-rw-r--r-- 1 root root 50 11/10/2023 02:39:16 arquivo5
```

```
root@localhost:~$ rm arquivo 5
Nao foi possivel remover o arquivo
root@localhost:~$ rm arquivo5
Arquivo removido
root@localhost:~$ ls -l
-rwxrwxrwx 1 root root 100 11/10/2023 02:24:06 arquivo.png
--wxr--r-- 1 root root 200 11/10/2023 02:24:12 arquivo.jpg
-rw-r--r-- 1 root root 120 11/10/2023 02:24:25 arquivo.cpp
-rw-r--r-- 1 root root 90 11/10/2023 02:24:42 vscode
d-----rwx 1 root root 1 11/10/2023 02:24:47 diretorio1
drwxr-xr-x 1 root root 1 11/10/2023 02:24:51 diretorio2
drwxr-xr-x 1 root root 1 11/10/2023 02:24:56 diretorio3
```

- h. **cd NomeDir** ou **.** ou **..** : navegar nos diretórios

```
root@localhost:~$ cd diretorio3
root@localhost:~/diretorio3$
```

- i. **link -h nomeArquivoOrigem NomeArquivoDestino**: criar link físico

```
root@localhost:~$ link -h diretorio1/diretorio2 linkfisico
root@localhost:~$ ls -l
-rwxrwxrwx 1 root root 100 11/10/2023 02:24:06 arquivo.png
--wxr--r-- 1 root root 200 11/10/2023 02:24:12 arquivo.jpg
-rw-r--r-- 1 root root 120 11/10/2023 02:24:25 arquivo.cpp
-rw-r--r-- 1 root root 90 11/10/2023 02:24:42 vscode
d-----rwx 1 root root 1 11/10/2023 02:24:47 diretorio1
drwxr-xr-x 1 root root 1 11/10/2023 02:24:51 diretorio2
drwxr-xr-x 1 root root 1 11/10/2023 02:24:56 diretorio3
drwxr-xr-x 2 root root 1 11/10/2023 02:41:11 linkfisico
```

- j. **link -s nomeArquivoOrigem NomeArquivoDestino**: criar link simbólico

```
root@localhost:~$ link -s diretorio1/diretorio2 linksimbolico
root@localhost:~$ ls -l
-rwxrwxrwx 1 root root 100 11/10/2023 02:24:06 arquivo.png
--wxr--r-- 1 root root 200 11/10/2023 02:24:12 arquivo.jpg
-rw-r--r-- 1 root root 120 11/10/2023 02:24:25 arquivo.cpp
-rw-r--r-- 1 root root 90 11/10/2023 02:24:42 vscode
d-----rwx 1 root root 1 11/10/2023 02:24:47 diretorio1
drwxr-xr-x 1 root root 1 11/10/2023 02:24:51 diretorio2
drwxr-xr-x 1 root root 1 11/10/2023 02:24:56 diretorio3
drwxr-xr-x 2 root root 1 11/10/2023 02:41:11 linkfisico
lrwxrwxrwx 1 root root 1 11/10/2023 02:42:17 linksimbolico -> diretorio1/diretorio2
```

k. **unlink -h**: remover link físico

```
root@localhost:~$ unlink -h linkfisico
root@localhost:~$ ls -l
-rwxrwxrwx 1 root root 100 11/10/2023 02:24:06 arquivo.png
--wxr--r-- 1 root root 200 11/10/2023 02:24:12 arquivo.jpg
-rw-r--r-- 1 root root 120 11/10/2023 02:24:25 arquivo.cpp
-rw-r--r-- 1 root root 90 11/10/2023 02:24:42 vscode
d-----rwx 1 root root 1 11/10/2023 02:24:47 diretorio1
drwxr-xr-x 1 root root 1 11/10/2023 02:24:51 diretorio2
drwxr-xr-x 1 root root 1 11/10/2023 02:24:56 diretorio3
lrwxrwxrwx 1 root root 1 11/10/2023 02:42:17 linksimbolico -> diretorio1/diretorio2
```

l. **unlink -s**: remover link simbólico

```
root@localhost:~$ unlink -s linksimbolico
root@localhost:~$ ls -l
-rwxrwxrwx 1 root root 100 11/10/2023 02:24:06 arquivo.png
--wxr--r-- 1 root root 200 11/10/2023 02:24:12 arquivo.jpg
-rw-r--r-- 1 root root 120 11/10/2023 02:24:25 arquivo.cpp
-rw-r--r-- 1 root root 90 11/10/2023 02:24:42 vscode
d-----rwx 1 root root 1 11/10/2023 02:24:47 diretorio1
drwxr-xr-x 1 root root 1 11/10/2023 02:24:51 diretorio2
drwxr-xr-x 1 root root 1 11/10/2023 02:24:56 diretorio3
```

m. **bad numeroBloco**: transformar um bloco em Bad

```
root@localhost:~$ bad 205
root@localhost:~$
```

n. **touch NomeArquivo TamanhoBytes:** Criar um arquivo Regular

```
root@localhost:~$ touch arquivo.png 150
Arquivo criado
root@localhost:~$
```

o. **df:** apresentar em bytes espaço livre e ocupados do disco

```
root@localhost:~$ df
Filesystem      Tamanho Usados  Disponivel   Uso%   Montado em
/dev/sda1       1000    141    856        14.10%   /
root@localhost:~$
```

p. **exit:** Finaliza a execução do sistema

```
root@localhost:~$ ls
arquivo.png  arquivo.jpg  arquivo.cpp  vscode  diretorio1  diretorio2  diretorio3  a
rquivo2 linkfisico  linksimbolico
root@localhost:~$ exit
PS C:\TrabalhoS02\S0II-Inode>
```

q. **clear:** Limpa a tela

```
Quantidade de blocos selecionada: 1000
root@localhost:~$ touch arquivo.png 100
Arquivo criado
root@localhost:~$ touch arquivo.jpg 200
Arquivo criado
root@localhost:~$ touch arquivo.cpp 120
Arquivo criado
root@localhost:~$ touch vscode 90
Arquivo criado
root@localhost:~$ mkdir diretorio1
Diretorio criado
root@localhost:~$ mkdir diretorio2
Diretorio criado
root@localhost:~$ mkdir diretorio3
Diretorio criado
root@localhost:~$ cd diretorio3
root@localhost:~/diretorio3$ clear
```

```
root@localhost:~/diretorio3$
```

Comandos de Relatórios:

- r. **trace disk**: Apresenta o estado atual de todos os blocos do disco

```
root@localhost:~$ trace disk
```

[0 R]	[1 R]	[2 R]	[3 R]	[4 R]	[5 R]	[6 R]	[7 R]	[8 R]	[9 R]	[10 R]
[11 R]	[12 R]	[13 R]	[14 R]	[15 R]	[16 R]	[17 R]	[18 R]	[19 R]	[20 R]	
[21 R]	[22 R]	[23 R]	[24 R]	[25 R]	[26 R]	[27 R]	[28 R]	[29 R]	[30 R]	
[31 R]	[32 R]	[33 R]	[34 R]	[35 R]	[36 R]	[37 R]	[38 R]	[39 R]	[40 R]	
[41 R]	[42 R]	[43 R]	[44 R]	[45 R]	[46 R]	[47 R]	[48 R]	[49 R]	[50 R]	
[51 R]	[52 R]	[53 R]	[54 R]	[55 R]	[56 R]	[57 R]	[58 R]	[59 R]	[60 R]	
[61 R]	[62 R]	[63 R]	[64 R]	[65 R]	[66 R]	[67 R]	[68 R]	[69 R]	[70 R]	
[71 R]	[72 R]	[73 R]	[74 R]	[75 R]	[76 R]	[77 R]	[78 R]	[79 R]	[80 R]	
[81 R]	[82 R]	[83 R]	[84 R]	[85 R]	[86 R]	[87 R]	[88 R]	[89 R]	[90 R]	
[91 R]	[92 R]	[93 R]	[94 R]	[95 R]	[96 R]	[97 R]	[98 R]	[99 R]	[100 F]	
[101 F]	[102 F]	[103 F]	[104 F]	[105 F]	[106 F]	[107 F]	[108 F]	[109 F]	[110 F]	
[111 F]	[112 F]	[113 F]	[114 F]	[115 F]	[116 F]	[117 F]	[118 F]	[119 F]	[120 F]	
[121 F]	[122 F]	[123 F]	[124 F]	[125 F]	[126 F]	[127 F]	[128 F]	[129 F]	[130 F]	
[131 F]	[132 F]	[133 F]	[134 F]	[135 F]	[136 F]	[137 F]	[138 F]	[139 F]	[140 F]	
[141 F]	[142 F]	[143 F]	[144 F]	[145 F]	[146 F]	[147 F]	[148 F]	[149 F]	[150 F]	
[151 F]	[152 F]	[153 F]	[154 F]	[155 F]	[156 F]	[157 F]	[158 F]	[159 F]	[160 F]	
[161 F]	[162 F]	[163 F]	[164 F]	[165 F]	[166 F]	[167 F]	[168 F]	[169 F]	[170 F]	
[171 F]	[172 F]	[173 F]	[174 F]	[175 F]	[176 F]	[177 F]	[178 F]	[179 F]	[180 F]	
[181 F]	[182 F]	[183 F]	[184 F]	[185 F]	[186 F]	[187 F]	[188 F]	[189 F]	[190 F]	
[191 F]	[192 F]	[193 F]	[194 F]	[195 F]	[196 F]	[197 F]	[198 F]	[199 F]	[200 F]	
[201 F]	[202 F]	[203 F]	[204 F]	[205 B]	[206 F]	[207 F]	[208 F]	[209 F]	[210 F]	
[211 F]	[212 F]	[213 F]	[214 F]	[215 F]	[216 F]	[217 F]	[218 F]	[219 F]	[220 F]	
[221 F]	[222 F]	[223 F]	[224 F]	[225 F]	[226 F]	[227 F]	[228 F]	[229 F]	[230 F]	
[231 F]	[232 F]	[233 F]	[234 F]	[235 F]	[236 F]	[237 F]	[238 F]	[239 F]	[240 F]	
[241 F]	[242 F]	[243 F]	[244 F]	[245 F]	[246 F]	[247 F]	[248 F]	[249 F]	[250 F]	
[251 F]	[252 F]	[253 F]	[254 F]	[255 F]	[256 F]	[257 F]	[258 F]	[259 F]	[260 F]	
[261 F]	[262 F]	[263 F]	[264 F]	[265 F]	[266 F]	[267 F]	[268 F]	[269 F]	[270 F]	
[271 F]	[272 F]	[273 F]	[274 F]	[275 F]	[276 F]	[277 F]	[278 F]	[279 F]	[280 F]	
[281 F]	[282 F]	[283 F]	[284 F]	[285 F]	[286 F]	[287 F]	[288 F]	[289 F]	[290 F]	
[291 F]	[292 F]	[293 F]	[294 F]	[295 F]	[296 F]	[297 F]	[298 F]	[299 F]	[300 F]	

[511 F]	[512 F]	[513 F]	[514 F]	[515 F]	[516 F]	[517 F]	[518 F]	[519 F]	[520 F]
[521 F]	[522 F]	[523 F]	[524 F]	[525 F]	[526 F]	[527 F]	[528 F]	[529 F]	[530 F]
[531 F]	[532 F]	[533 F]	[534 F]	[535 F]	[536 F]	[537 F]	[538 F]	[539 F]	[540 F]
[541 F]	[542 F]	[543 F]	[544 F]	[545 F]	[546 F]	[547 F]	[548 F]	[549 F]	[550 F]
[551 F]	[552 F]	[553 F]	[554 F]	[555 F]	[556 F]	[557 F]	[558 F]	[559 F]	[560 F]
[561 F]	[562 F]	[563 F]	[564 F]	[565 F]	[566 F]	[567 F]	[568 F]	[569 F]	[570 F]
[571 F]	[572 F]	[573 F]	[574 F]	[575 F]	[576 F]	[577 F]	[578 F]	[579 F]	[580 F]
[581 F]	[582 F]	[583 F]	[584 F]	[585 F]	[586 F]	[587 F]	[588 F]	[589 F]	[590 F]
[591 F]	[592 F]	[593 F]	[594 F]	[595 F]	[596 F]	[597 F]	[598 F]	[599 F]	[600 F]
[601 F]	[602 F]	[603 F]	[604 F]	[605 F]	[606 F]	[607 F]	[608 F]	[609 F]	[610 F]
[611 F]	[612 F]	[613 F]	[614 F]	[615 F]	[616 F]	[617 F]	[618 F]	[619 F]	[620 F]
[621 F]	[622 F]	[623 F]	[624 F]	[625 F]	[626 F]	[627 F]	[628 F]	[629 F]	[630 F]
[631 F]	[632 F]	[633 F]	[634 F]	[635 F]	[636 F]	[637 F]	[638 F]	[639 F]	[640 F]
[641 F]	[642 F]	[643 F]	[644 F]	[645 F]	[646 F]	[647 F]	[648 F]	[649 F]	[650 F]
[651 F]	[652 F]	[653 F]	[654 F]	[655 F]	[656 F]	[657 F]	[658 F]	[659 F]	[660 F]
[661 F]	[662 F]	[663 F]	[664 F]	[665 F]	[666 F]	[667 F]	[668 F]	[669 F]	[670 F]
[671 F]	[672 F]	[673 F]	[674 F]	[675 F]	[676 F]	[677 F]	[678 F]	[679 F]	[680 F]
[681 F]	[682 F]	[683 F]	[684 F]	[685 F]	[686 F]	[687 F]	[688 F]	[689 F]	[690 F]
[691 F]	[692 F]	[693 F]	[694 F]	[695 F]	[696 F]	[697 F]	[698 F]	[699 F]	[700 F]
[701 F]	[702 F]	[703 F]	[704 F]	[705 F]	[706 F]	[707 F]	[708 F]	[709 F]	[710 F]
[711 F]	[712 F]	[713 B]	[714 B]	[715 B]	[716 B]	[717 F]	[718 F]	[719 F]	[720 F]
[721 F]	[722 F]	[723 F]	[724 F]	[725 F]	[726 F]	[727 F]	[728 F]	[729 F]	[730 F]
[731 F]	[732 F]	[733 F]	[734 F]	[735 F]	[736 F]	[737 F]	[738 F]	[739 F]	[740 F]
[741 F]	[742 F]	[743 F]	[744 F]	[745 F]	[746 F]	[747 F]	[748 F]	[749 F]	[750 F]
[751 F]	[752 F]	[753 F]	[754 F]	[755 F]	[756 F]	[757 F]	[758 F]	[759 F]	[760 F]
[761 F]	[762 F]	[763 F]	[764 F]	[765 F]	[766 F]	[767 F]	[768 F]	[769 F]	[770 F]
[771 F]	[772 F]	[773 F]	[774 F]	[775 F]	[776 F]	[777 F]	[778 F]	[779 F]	[780 F]
[781 F]	[782 F]	[783 F]	[784 F]	[785 F]	[786 F]	[787 F]	[788 F]	[789 F]	[790 F]
[791 F]	[792 F]	[793 F]	[794 F]	[795 F]	[796 F]	[797 F]	[798 F]	[799 F]	[800 F]
[801 F]	[802 F]	[803 F]	[804 F]	[805 F]	[806 F]	[807 F]	[808 F]	[809 F]	[810 F]
[811 F]	[812 F]	[813 F]	[814 F]	[815 F]	[816 F]	[817 F]	[818 F]	[819 F]	[820 F]
[821 F]	[822 F]	[823 F]	[824 F]	[825 F]	[826 F]	[827 F]	[828 F]	[829 F]	[830 F]
[831 F]	[832 F]	[833 F]	[834 F]	[835 F]	[836 F]	[837 F]	[838 F]	[839 F]	[840 F]
[841 F]	[842 F]	[843 F]	[844 F]	[845 F]	[846 F]	[847 F]	[848 F]	[849 F]	[850 F]
[851 F]	[852 F]	[853 F]	[854 F]	[855 F]	[856 F]	[857 F]	[858 F]	[859 F]	[860 F]
[861 F]	[862 F]	[863 F]	[864 F]	[865 F]	[866 F]	[867 F]	[868 F]	[869 F]	[870 F]
[871 F]	[872 F]	[873 F]	[874 F]	[875 F]	[876 F]	[877 F]	[878 F]	[879 F]	[880 F]
[881 F]	[882 F]	[883 F]	[884 F]	[885 F]	[886 F]	[887 F]	[888 F]	[889 F]	[890 F]
[891 F]	[892 F]	[893 F]	[894 F]	[895 F]	[896 F]	[897 F]	[898 F]	[899 F]	[900 F]
[901 F]	[902 F]	[903 F]	[904 F]	[905 F]	[906 F]	[907 F]	[908 F]	[909 F]	[910 F]
[911 F]	[912 F]	[913 F]	[914 F]	[915 F]	[916 F]	[917 F]	[918 F]	[919 F]	[920 F]
[921 F]	[922 F]	[923 F]	[924 F]	[925 A]	[926 A]	[928 F]	[928 F]	[929 F]	[930 I]
[931 A]	[932 I]	[933 A]	[934 I]	[935 A]	[936 I]	[937 A]	[938 A]	[939 A]	[940 A]
[941 I]	[942 A]	[943 A]	[944 A]	[945 A]	[946 A]	[947 I]	[948 A]	[949 A]	[950 A]

- s. **check files:** Realiza uma varredura e exibe o estado do arquivo do disco, se encontra-se corrompido ou íntegro

```
root@localhost:~$ check files
Nome          Status
arquivo.png   [CORROMPIDO]
arquivo.jpg   [INTEGRO]
arquivo.cpp   [CORROMPIDO]
vscode        [INTEGRO]
diretorio1    [INTEGRO]
diretorio2    [INTEGRO]
diretorio3    [INTEGRO]
arquivo2      [INTEGRO]
```

- t. **lost block:** Apresenta a quantidade de blocos que foram não foram usados (perdidos), são os blocos que não apresentam-se como livres e nem como defeituoso

```
root@localhost:~$ lost block
Foram perdidos 100 (1000 em Bytes) de 1000 blocos (10.00% de perca)
```

- u. **max file:** Estima o máximo de blocos que poderão ser utilizados pelo arquivo.

```
root@localhost:~$ max file
Pode ser usado 713 de 821 blocos para inserir um arquivo(86.85% aproveitamento)
root@localhost:~$
```


- v. **ls -e**: Apresenta todos os arquivos do diretório atual, seu tipo e os blocos

```
root@localhost:~$ ls -e
Nome                Tipo                Blocos
arquivo.png          Arquivo             997-996-995-994-993-992-991-990-989-988-987-986
arquivo.jpg          Arquivo             985-984-983-982-981-980-979-978-977-976-975-974-973-972-971-970-969-968-967-966-965-964-963-962
arquivo.cpp          Arquivo             961-960-959-958-957-956-955-954-953-952-951-950-949-948
vscode              Arquivo             947-946-945-944-943-942-941-940-939-938-937
diretorio1           Diretorio            936-935
diretorio2           Diretorio            934-933
diretorio3           Diretorio            932-931
arquivo2             Arquivo             928-928
```

utilizados. Similar a uma visão de lista do Windows Explorer

- w. **ls -li**: Apresenta todos os links do diretório atual, seu tipo, o endereço do inode e a quantidade de links físicos.

```
root@localhost:~$ ls -li
Nome                Tipo                Num. Inode                Qtd. Link Físico
linkfisico          Físico              930                        2
linksimbolico       Simbolico           929                        1
```

- x. **ls -la**: Mostra todos arquivos e seu atributos incluindo os arquivos ocultos com o nome iniciando com '.' na listagem.

```
root@localhost:~$ ls -la
drwxr-xr-x 1 root root 1 11/10/2023 03:02:23 .
drwxr-xr-x 1 root root 1 11/10/2023 03:02:23 ..
drwxr-xr-x 1 root root 1 11/10/2023 03:02:32 diretorio1
drwxr-xr-x 1 root root 1 11/10/2023 03:02:35 diretorio2
drwxr-xr-x 1 root root 1 11/10/2023 03:02:38 diretorio3
drwxr-xr-x 1 root root 1 11/10/2023 03:02:42 diretorio4
-rw-r--r-- 1 root root 10 11/10/2023 03:02:50 arquivo1
-rw-r--r-- 1 root root 20 11/10/2023 03:02:55 arquivo2
-rw-r--r-- 1 root root 40 11/10/2023 03:03:00 arquivo3
-rw-r--r-- 1 root root 500 11/10/2023 03:03:06 arquivo4
```

5. Relatórios

1. O número de blocos ocupados por um arquivo escolhido por usuário;

```
root@localhost:~$ vi vscode
vscode: 947-946-945-944-943-942-941-940-939-938-937
Quantidade de endereços: 11
```

2. O tamanho (em blocos) do maior arquivo que ainda pode ser criado nesse disco;

```
root@localhost:~$ max file
Pode ser usado 713 de 821 blocos para inserir um arquivo(86.85% aproveitamento)
root@localhost:~$
```

3. Quais arquivos estão íntegros e quais estão corrompidos por blocos defeituosos (badblocks);

```
root@localhost:~$ check files
Nome          Status
arquivo.png   [CORROMPIDO]
arquivo.jpg   [INTEGRO]
arquivo.cpp   [CORROMPIDO]
vscode        [INTEGRO]
diretorio1    [INTEGRO]
diretorio2    [INTEGRO]
diretorio3    [INTEGRO]
arquivo2      [INTEGRO]
```

4. Apresentar quantos blocos do disco estão perdidos e quais são eles, ou seja, não são usados por arquivos e nem estão marcados como livres ou defeituosos. Não esqueça de apresentar o espaço de disco perdido em bytes;

```
root@localhost:~$ lost block
Foram perdidos 100 (1000 em Bytes) de 1000 blocos (10.00% de perca)
```

5. Imprima todos os blocos em seu estado atual, igual a figura b.

```
root@localhost:~$ trace disk

[0 R]  [1 R]  [2 R]  [3 R]  [4 R]  [5 R]  [6 R]  [7 R]  [8 R]  [9 R]  [10 R]
[11 R] [12 R] [13 R] [14 R] [15 R] [16 R] [17 R] [18 R] [19 R] [20 R]
[21 R] [22 R] [23 R] [24 R] [25 R] [26 R] [27 R] [28 R] [29 R] [30 R]
[31 R] [32 R] [33 R] [34 R] [35 R] [36 R] [37 R] [38 R] [39 R] [40 R]
[41 R] [42 R] [43 R] [44 R] [45 R] [46 R] [47 R] [48 R] [49 R] [50 R]
[51 R] [52 R] [53 R] [54 R] [55 R] [56 R] [57 R] [58 R] [59 R] [60 R]
[61 R] [62 R] [63 R] [64 R] [65 R] [66 R] [67 R] [68 R] [69 R] [70 R]
[71 R] [72 R] [73 R] [74 R] [75 R] [76 R] [77 R] [78 R] [79 R] [80 R]
```

6. Visualize os arquivos e diretórios alocados, apresentando o nome e seus números do bloco correspondentes identificando o tipo. Esta visualização deve ser algo semelhante ao Windows Explorer. Não precisa ser gráfico.

7. Visualizar a árvore de diretório igual a figura D, apresentado as estruturas com seus atribuídos, nomes e números;

[511 F]	[512 F]	[513 F]	[514 F]	[515 F]	[516 F]	[517 F]	[518 F]	[519 F]	[520 F]
[521 F]	[522 F]	[523 F]	[524 F]	[525 F]	[526 F]	[527 F]	[528 F]	[529 F]	[530 F]
[531 F]	[532 F]	[533 F]	[534 F]	[535 F]	[536 F]	[537 F]	[538 F]	[539 F]	[540 F]
[541 F]	[542 F]	[543 F]	[544 F]	[545 F]	[546 F]	[547 F]	[548 F]	[549 F]	[550 F]
[551 F]	[552 F]	[553 F]	[554 F]	[555 F]	[556 F]	[557 F]	[558 F]	[559 F]	[560 F]
[561 F]	[562 F]	[563 F]	[564 F]	[565 F]	[566 F]	[567 F]	[568 F]	[569 F]	[570 F]
[571 F]	[572 F]	[573 F]	[574 F]	[575 F]	[576 F]	[577 F]	[578 F]	[579 F]	[580 F]
[581 F]	[582 F]	[583 F]	[584 F]	[585 F]	[586 F]	[587 F]	[588 F]	[589 F]	[590 F]
[591 F]	[592 F]	[593 F]	[594 F]	[595 F]	[596 F]	[597 F]	[598 F]	[599 F]	[600 F]
[601 F]	[602 F]	[603 F]	[604 F]	[605 F]	[606 F]	[607 F]	[608 F]	[609 F]	[610 F]
[611 F]	[612 F]	[613 F]	[614 F]	[615 F]	[616 F]	[617 F]	[618 F]	[619 F]	[620 F]
[621 F]	[622 F]	[623 F]	[624 F]	[625 F]	[626 F]	[627 F]	[628 F]	[629 F]	[630 F]
[631 F]	[632 F]	[633 F]	[634 F]	[635 F]	[636 F]	[637 F]	[638 F]	[639 F]	[640 F]
[641 F]	[642 F]	[643 F]	[644 F]	[645 F]	[646 F]	[647 F]	[648 F]	[649 F]	[650 F]
[651 F]	[652 F]	[653 F]	[654 F]	[655 F]	[656 F]	[657 F]	[658 F]	[659 F]	[660 F]
[661 F]	[662 F]	[663 F]	[664 F]	[665 F]	[666 F]	[667 F]	[668 F]	[669 F]	[670 F]
[671 F]	[672 F]	[673 F]	[674 F]	[675 F]	[676 F]	[677 F]	[678 F]	[679 F]	[680 F]
[681 F]	[682 F]	[683 F]	[684 F]	[685 F]	[686 F]	[687 F]	[688 F]	[689 F]	[690 F]
[691 F]	[692 F]	[693 F]	[694 F]	[695 F]	[696 F]	[697 F]	[698 F]	[699 F]	[700 F]
[701 F]	[702 F]	[703 F]	[704 F]	[705 F]	[706 F]	[707 F]	[708 F]	[709 F]	[710 F]
[711 F]	[712 F]	[713 B]	[714 B]	[715 B]	[716 B]	[717 F]	[718 F]	[719 F]	[720 F]
[721 F]	[722 F]	[723 F]	[724 F]	[725 F]	[726 F]	[727 F]	[728 F]	[729 F]	[730 F]
[731 F]	[732 F]	[733 F]	[734 F]	[735 F]	[736 F]	[737 F]	[738 F]	[739 F]	[740 F]
[741 F]	[742 F]	[743 F]	[744 F]	[745 F]	[746 F]	[747 F]	[748 F]	[749 F]	[750 F]
[751 F]	[752 F]	[753 F]	[754 F]	[755 F]	[756 F]	[757 F]	[758 F]	[759 F]	[760 F]
[761 F]	[762 F]	[763 F]	[764 F]	[765 F]	[766 F]	[767 F]	[768 F]	[769 F]	[770 F]
[771 F]	[772 F]	[773 F]	[774 F]	[775 F]	[776 F]	[777 F]	[778 F]	[779 F]	[780 F]
[781 F]	[782 F]	[783 F]	[784 F]	[785 F]	[786 F]	[787 F]	[788 F]	[789 F]	[790 F]
[791 F]	[792 F]	[793 F]	[794 F]	[795 F]	[796 F]	[797 F]	[798 F]	[799 F]	[800 F]
[801 F]	[802 F]	[803 F]	[804 F]	[805 F]	[806 F]	[807 F]	[808 F]	[809 F]	[810 F]
[811 F]	[812 F]	[813 F]	[814 F]	[815 F]	[816 F]	[817 F]	[818 F]	[819 F]	[820 F]
[821 F]	[822 F]	[823 F]	[824 F]	[825 F]	[826 F]	[827 F]	[828 F]	[829 F]	[830 F]
[831 F]	[832 F]	[833 F]	[834 F]	[835 F]	[836 F]	[837 F]	[838 F]	[839 F]	[840 F]
[841 F]	[842 F]	[843 F]	[844 F]	[845 F]	[846 F]	[847 F]	[848 F]	[849 F]	[850 F]
[851 F]	[852 F]	[853 F]	[854 F]	[855 F]	[856 F]	[857 F]	[858 F]	[859 F]	[860 F]
[861 F]	[862 F]	[863 F]	[864 F]	[865 F]	[866 F]	[867 F]	[868 F]	[869 F]	[870 F]
[871 F]	[872 F]	[873 F]	[874 F]	[875 F]	[876 F]	[877 F]	[878 F]	[879 F]	[880 F]
[881 F]	[882 F]	[883 F]	[884 F]	[885 F]	[886 F]	[887 F]	[888 F]	[889 F]	[890 F]
[891 F]	[892 F]	[893 F]	[894 F]	[895 F]	[896 F]	[897 F]	[898 F]	[899 F]	[900 F]
[901 F]	[902 F]	[903 F]	[904 F]	[905 F]	[906 F]	[907 F]	[908 F]	[909 F]	[910 F]
[911 F]	[912 F]	[913 F]	[914 F]	[915 F]	[916 F]	[917 F]	[918 F]	[919 F]	[920 F]
[921 F]	[922 F]	[923 F]	[924 F]	[925 A]	[926 A]	[928 F]	[928 F]	[929 F]	[930 I]
[931 A]	[932 I]	[933 A]	[934 I]	[935 A]	[936 I]	[937 A]	[938 A]	[939 A]	[940 A]
[941 I]	[942 A]	[943 A]	[944 A]	[945 A]	[946 A]	[947 I]	[948 A]	[949 A]	[950 A]

8. Visualizar os links simbólicos e físicos criados, detalhando as entradas de diretório, inodes e números

```
root@localhost:~$ ls -li
```

Nome	Tipo	Num. Inode	Qtd. Link Físico
linkfisico	Físico	930	2
linksimbolico	Simbolico	929	1

6. Conclusão

Durante a implementação do trabalho foi possível notar as diferentes peças-chaves na implementação de sistemas de arquivos são essenciais para a organização e a segurança de dados em computadores.

Ao longo de todo o estudo, abordamos o desenvolvimento, configuração e personalização de um ambiente, bem como a importância de compreender seus princípios fundamentais. ficou evidente que os I-Nodes servem como a espinha dorsal de um sistema de arquivos, permitindo a alocação eficiente de espaço em disco, o rastreamento de metadados essenciais, como permissões, timestamps e links rígidos, além de possibilitar a recuperação rápida de dados.

Compreender a estrutura e o funcionamento dos I-Nodes é crucial para qualquer profissional de TI que lida com administração de sistemas ou desenvolvimento de software.

Em suma, este trabalho ressalta a relevância dos I-Nodes na implementação de arquivos e sistemas de armazenamento, enfatizando sua organização, segurança e o desempenho de sistemas de computação.