

SIMULADOR DE SISTEMA DE ARQUIVOS

Autor 1: Matheus Moura de Lima.

Palavras-chave: Java, arquivo, diretório, log, CLI.

Resumo

Este trabalho propõe o desenvolvimento de um simulador, para facilitar a compreensão do funcionamento de um sistema de arquivos.

Introdução

Primeiramente, sistemas de arquivos são componentes essenciais dos sistemas operacionais, responsáveis pela organização e gerenciamento dos dados armazenados em dispositivos de armazenamento. Vale salientar que compreender o funcionamento interno de um sistema de arquivos é crucial para estudantes e profissionais da área de ciência da computação e engenharia de software. Este trabalho apresenta o desenvolvimento de um simulador de sistema de arquivos em Java, objetivando ilustrar as operações básicas e a lógica de um sistema de arquivos real.

Metodologia

A metodologia adotada para o desenvolvimento do simulador envolveu a implementação de uma série de classes em Java, cada uma responsável por diferentes aspectos do sistema de arquivos:

- **FileSystemSimulator:** esta classe implementa as operações principais do sistema de arquivos, manipulando um arquivo de texto (filesystem.txt) que simula o armazenamento. As operações incluem criação, cópia, renomeação e exclusão de arquivos e diretórios.
- **CustomFile:** representa um arquivo no sistema de arquivos. Suas responsabilidades incluem armazenar informações sobre o arquivo, como seu nome e caminho, e fornecer métodos para manipulação de arquivos.
- **Directory:** representa um diretório no sistema de arquivos. Suas responsabilidades incluem armazenar informações sobre o diretório, como seu nome e caminho, e manter uma lista de arquivos e subdiretórios contidos dentro dele. Ela também fornece métodos para manipulação de diretórios.
- **Journal:** classe responsável por gerenciar o log das operações realizadas no sistema de arquivos, fornecendo um histórico para auditoria e depuração.
- **Shell:** implementa a interface de linha de comando (CLI) que permite ao usuário interagir com o simulador, digitando comandos para realizar as diversas operações disponíveis.

As operações implementadas incluem:

- Criação de arquivos e diretórios.
- Cópia de arquivos.
- Renomeação de arquivos e diretórios.
- Exclusão de arquivos e diretórios.
- Listagem de conteúdo de diretórios.

Além disso, um aspecto central da metodologia foi a criação e uso de um arquivo chamado `filesystem.txt` para simular o armazenamento do sistema de arquivos. Esse arquivo de texto armazena as entradas de todos os arquivos e diretórios do sistema de arquivos simulado, fornecendo uma representação persistente do estado do sistema. Vale ressaltar que cada linha do arquivo representa um arquivo ou diretório no sistema:

- Arquivos: Representados pelo prefixo `FILE` seguido pelo caminho completo do arquivo.
- Diretórios: Representados pelo prefixo `DIR` seguido pelo caminho completo do diretório.

Resultados e Discussão

A seguir, são apresentados exemplos de comandos e suas saídas esperadas ao utilizar o simulador:

- Criação de um diretório e um arquivo:

```
C:\Users\matheus.moura\.jdk\openjdk-20\bin\java.exe "-javaagent:C:\Program Files\JetBrains\
Simulador de Sistema de Arquivos!
fs-shell> mkdir diretorio
fs-shell> touch arquivo.txt
fs-shell>
```

Esses comandos criam um diretório chamado “diretorio” e um arquivo chamado “arquivo.txt”.

- Renomeação de um Diretório:

```
C:\Users\matheus.moura\.jdk\openjdk-20\bin\java.exe "-javaagent:C:\Program Files\JetBrains\
Simulador de Sistema de Arquivos!
fs-shell> renomedir Pasta pasta
fs-shell> list pasta
File: pasta/teste.txt
fs-shell> |
```

O diretório “Pasta” é renomeado para “pasta”, atualizando todos os arquivos e subdiretórios.

- Listagem do Conteúdo de um Diretório:

```
C:\Users\matheus.moura\.jdk\openjdk-20\bin\java.exe "-javaagent:C:\Program Files\JetBrains\
Simulador de Sistema de Arquivos!
fs-shell> touch diretorio/teste.txt
fs-shell> list diretorio
File: diretorio/teste.txt
fs-shell>
```

Lista todos os arquivos e subdiretórios dentro de “diretorio”.

- Exclusão de um Arquivo e um Diretório:

```
C:\Users\matheus.moura\.jdk\openjdk-20\bin\java.exe "-javaagent:C:\Program Files\JetBrains\
Simulador de Sistema de Arquivos!
fs-shell> delete pasta/teste.txt
fs-shell> list pasta
fs-shell>
```

Remove o arquivo teste.txt e o diretório “pasta”.

Assim, a implementação demonstrou ser eficaz para simular as operações básicas de um sistema de arquivos, permitindo ao usuário realizar as operações de criação, cópia, renomeação e exclusão de arquivos e diretórios de forma intuitiva. É importante lembrar que a utilização de um arquivo de texto (filesystem.txt) como armazenamento simulado facilitou a visualização e compreensão das operações realizadas.

Conclusão

Portanto, pode-se concluir que o simulador de sistema de arquivos desenvolvido em Java cumpriu o objetivo de ilustrar o funcionamento básico de um sistema de arquivos, pois, por meio dele, é possível entender como os sistemas de arquivos gerenciam dados, realizam operações de manipulação de arquivos e diretórios, e mantêm a integridade das estruturas de dados armazenadas. Dessa forma, este trabalho destaca a importância de um sistema de arquivos eficiente e robusto, essencial para a gestão eficaz de dados em qualquer sistema operacional.

Referências

FILE Systems in Operating System, 2023. Disponível em: <https://www.geeksforgeeks.org/file-systems-in-operating-system/> Acesso em: 14 de jun. de 2024.