# **Cenários de Teste (TP de SO) GR**UPO: ARTHUR MENDES, BERNARDO ALVAREZ, BRUNO FERRAZ

**Como usar:**

1. Inicie o sistema de arquivos como root ou um usuário específico, por exemplo: java -cp . filesys.Main root
2. Utilize as opções numéricas do menu (1 a 9) para chamar os comandos, fornecendo os caminhos e parâmetros solicitados.
3. Observe a saída no console para verificar se o comportamento esperado (sucesso, erro com exceção) foi atingido.

## **1. mkdir (Criar Diretório)**

**Objetivo:** Testar a criação de diretórios, incluindo caminhos intermediários e validações.

* **Cenário 1: Criar diretório simples**
  + **Usuário:** root
  + **Comando:** mkdir /pasta1
  + **Saída Esperada:** Diretório 'pasta1' criado em '/'.
* **Cenário 2: Criar diretórios intermediários (comportamento -p padrão)**
  + **Usuário:** root
  + **Comando:** mkdir /projetos/equipeA/docs
  + **Saída Esperada:** Diretório 'projetos' criado em '/'. Diretório 'equipeA' criado em '/projetos'. Diretório 'docs' criado em '/projetos/equipeA'.
* **Cenário 3: Tentar criar diretório já existente**
  + **Usuário:** root
  + **Pré-condição:** /pasta1 já existe.
  + **Comando:** mkdir /pasta1
  + **Saída Esperada:** Erro: Não é possível criar diretório '/pasta1': já existe um diretório com este nome. (CaminhoJaExistenteException)
* **Cenário 4: Tentar criar diretório onde já existe um arquivo**
  + **Usuário:** root
  + **Pré-condição:** Criar /arquivo.txt (touch /arquivo.txt)
  + **Comando:** mkdir /arquivo.txt
  + **Saída Esperada:** Erro: Não é possível criar diretório 'arquivo.txt': já existe um arquivo com este nome. (CaminhoJaExistenteException)
* **Cenário 5: Tentar criar diretório sem permissão de escrita no pai**
  + **Usuário:** root
  + **Pré-condição:** Criar /privado (mkdir /privado). Alterar permissão: chmod /privado root other --- (remover escrita para 'other').
  + **Trocar Usuário:** su joao
  + **Comando:** mkdir /privado/secreto
  + **Saída Esperada:** Erro: Permissão negada: não pode criar diretório em '/privado' para o usuário 'joao'. (PermissaoException)

## **2. touch (Criar Arquivo Vazio ou Atualizar Timestamp)**

**Objetivo:** Testar a criação de arquivos e a atualização de metadados.

* **Cenário 1: Criar arquivo vazio**
  + **Usuário:** root
  + **Comando:** touch /documento.pdf
  + **Saída Esperada:** Arquivo 'documento.pdf' criado em '/'.
* **Cenário 2: "Tocar" em arquivo existente (atualizar timestamp)**
  + **Usuário:** root
  + **Pré-condição:** /documento.pdf existe.
  + **Comando:** touch /documento.pdf
  + **Saída Esperada:** Arquivo 'documento.pdf' já existe, tempo de modificação atualizado.
* **Cenário 3: Tentar criar arquivo onde já existe um diretório**
  + **Usuário:** root
  + **Pré-condição:** /config já existe (mkdir /config).
  + **Comando:** touch /config
  + **Saída Esperada:** Erro: Não é possível criar arquivo 'config': já existe um diretório com este nome. (CaminhoJaExistenteException)
* **Cenário 4: Tentar criar arquivo em diretório pai inexistente**
  + **Usuário:** root
  + **Comando:** touch /naoexiste/arquivo.txt
  + **Saída Esperada:** Erro: Caminho pai '/naoexiste' não existe ou não é um diretório. (CaminhoNaoEncontradoException)
* **Cenário 5: Tentar criar arquivo sem permissão de escrita no pai**
  + **Usuário:** root
  + **Pré-condição:** Criar /publico (mkdir /publico). Alterar permissão: chmod /publico root other r-x (remover escrita para 'other').
  + **Trocar Usuário:** su maria
  + **Comando:** touch /publico/relatorio.txt
  + **Saída Esperada:** Erro: Permissão negada: não pode criar arquivo em '/publico' para o usuário 'maria'. (PermissaoException)

## **3. write (Escrever em Arquivo)**

**Objetivo:** Testar a escrita de dados em arquivos, incluindo sobrescrita e anexação.

* **Cenário 1: Escrever em arquivo vazio (sobrescrever)**
  + **Usuário:** root
  + **Pré-condição:** /log.txt existe (touch /log.txt).
  + **Comando:** write /log.txt false "Primeira linha do log"
  + **Saída Esperada:** <número> bytes escritos em '/log.txt'. (Verifique com read).
* **Cenário 2: Anexar dados em arquivo**
  + **Usuário:** root
  + **Pré-condição:** /log.txt contém "Primeira linha do log".
  + **Comando:** write /log.txt true "Segunda linha do log"
  + **Saída Esperada:** <número> bytes escritos em '/log.txt'. (Verifique com read se o conteúdo é "Primeira linha do logSegunda linha do log").
* **Cenário 3: Escrever em arquivo existente (sobrescrever)**
  + **Usuário:** root
  + **Pré-condição:** /log.txt contém "Primeira linha do logSegunda linha do log".
  + **Comando:** write /log.txt false "Conteudo totalmente novo"
  + **Saída Esperada:** <número> bytes escritos em '/log.txt'. (Verifique com read se o conteúdo é apenas "Conteudo totalmente novo").
* **Cenário 4: Tentar escrever em diretório (erro de tipo)**
  + **Usuário:** root
  + **Pré-condição:** /home existe.
  + **Comando:** write /home false "dados"
  + **Saída Esperada:** Erro: '/home' não é um arquivo ou não existe para escrita. (CaminhoNaoEncontradoException)
* **Cenário 5: Tentar escrever em arquivo inexistente**
  + **Usuário:** root
  + **Comando:** write /naoexiste.txt false "dados"
  + **Saída Esperada:** Erro: '/naoexiste.txt' não é um arquivo ou não existe para escrita. (CaminhoNaoEncontradoException)
* **Cenário 6: Tentar escrever sem permissão de escrita no arquivo**
  + **Usuário:** root
  + **Pré-condição:** Criar /somente\_leitura.txt (touch /somente\_leitura.txt). Alterar permissão: chmod /somente\_leitura.txt root other r--.
  + **Trocar Usuário:** su tiago
  + **Comando:** write /somente\_leitura.txt false "tentativa"
  + **Saída Esperada:** Erro: Permissão negada: não pode escrever em '/somente\_leitura.txt' para o usuário 'tiago'. (PermissaoException)

## **4. read (Ler Arquivo)**

**Objetivo:** Testar a leitura de dados de arquivos.

* **Cenário 1: Ler arquivo existente**
  + **Usuário:** root
  + **Pré-condição:** Escrever conteúdo em /meu\_arquivo.txt (write /meu\_arquivo.txt false "Conteudo de teste para leitura.").
  + **Comando:** read /meu\_arquivo.txt
  + **Saída Esperada:** Conteúdo lido: "Conteudo de teste para leitura."
* **Cenário 2: Tentar ler arquivo inexistente**
  + **Usuário:** root
  + **Comando:** read /naoexiste\_arquivo.txt
  + **Saída Esperada:** Erro: '/naoexiste\_arquivo.txt' não é um arquivo ou não existe para leitura. (CaminhoNaoEncontradoException)
* **Cenário 3: Tentar ler diretório (erro de tipo)**
  + **Usuário:** root
  + **Pré-condição:** /dados existe (mkdir /dados).
  + **Comando:** read /dados
  + **Saída Esperada:** Erro: '/dados' não é um arquivo ou não existe para leitura. (CaminhoNaoEncontradoException)
* **Cenário 4: Tentar ler sem permissão de leitura no arquivo**
  + **Usuário:** root
  + **Pré-condição:** Criar /confidencial.txt (touch /confidencial.txt). Alterar permissão: chmod /confidencial.txt root other ---. Escrever conteúdo: write /confidencial.txt false "Dados secretos"
  + **Trocar Usuário:** su carla
  + **Comando:** read /confidencial.txt
  + **Saída Esperada:** Erro: Permissão negada: não pode ler '/confidencial.txt' para o usuário 'carla'. (PermissaoException)

## **5. ls (Listar Conteúdo de Diretório)**

**Objetivo:** Testar a listagem de arquivos e subdiretórios, recursivamente e com validação de permissão.

* **Cenário 1: Listar conteúdo da raiz**
  + **Usuário:** root
  + **Comando:** ls /
  + **Saída Esperada:** Lista de filesys, exception, lib, tests, users, etc.
* **Cenário 2: Listar diretório específico (não recursivo)**
  + **Usuário:** root
  + **Pré-condição:** Criar /tmp/tempfile.txt e /tmp/subdir
  + **Comando:** ls /tmp
  + **Saída Esperada:** Lista apenas tempfile.txt e subdir dentro de /tmp.
* **Cenário 3: Listar diretório recursivamente**
  + **Usuário:** root
  + **Pré-condição:** Criar /home/user/docs/report.txt e /home/user/code/main.java
  + **Comando:** ls /home true
  + **Saída Esperada:** Listagem em árvore de /home e todo seu conteúdo.
* **Cenário 4: Listar informações de um arquivo (ls em arquivo)**
  + **Usuário:** root
  + **Pré-condição:** /documento.txt existe.
  + **Comando:** ls /documento.txt
  + **Saída Esperada:** documento.txt (arquivo, <tamanho> bytes) - <permissões> <dono>
* **Cenário 5: Tentar listar caminho inexistente**
  + **Usuário:** root
  + **Comando:** ls /diretorio\_fantasma
  + **Saída Esperada:** Erro: ls: '/diretorio\_fantasma': Nenhum arquivo ou diretório encontrado. (CaminhoNaoEncontradoException)
* **Cenário 6: Tentar listar sem permissão de leitura (read) no diretório**
  + **Usuário:** root
  + **Pré-condição:** Criar /secretaria (mkdir /secretaria). Alterar permissão: chmod /secretaria root other --x (não permite leitura, só navegação).
  + **Trocar Usuário:** su atendente
  + **Comando:** ls /secretaria
  + **Saída Esperada:** Erro: Permissão negada: não pode ler diretório '/secretaria' para o usuário 'atendente'. (PermissaoException)
* **Cenário 7: Tentar listar sem permissão de execução (execute/navegação) no diretório**
  + **Usuário:** root
  + **Pré-condição:** Criar /restrito (mkdir /restrito). Alterar permissão: chmod /restrito root other rw- (permite leitura/escrita, mas não navegação).
  + **Trocar Usuário:** su visitante
  + **Comando:** ls /restrito
  + **Saída Esperada:** Erro: Permissão negada: não pode navegar (executar) no diretório '/restrito' para o usuário 'visitante'. (PermissaoException)

## **6. mv (Mover/Renomear Arquivo/Diretório)**

**Objetivo:** Testar a movimentação e renomeação de itens, considerando todos os casos e permissões.

* **Cenário 1: Renomear arquivo**
  + **Usuário:** root
  + **Pré-condição:** touch /velho\_nome.txt
  + **Comando:** mv /velho\_nome.txt /novo\_nome.txt
  + **Saída Esperada:** Item '/velho\_nome.txt' movido/renomeado para '/novo\_nome.txt'.
* **Cenário 2: Mover arquivo para outro diretório**
  + **Usuário:** root
  + **Pré-condição:** mkdir /destino\_pasta. touch /origem\_arquivo.txt
  + **Comando:** mv /origem\_arquivo.txt /destino\_pasta/origem\_arquivo.txt
  + **Saída Esperada:** Arquivo '/origem\_arquivo.txt' movido para '/destino\_pasta/origem\_arquivo.txt'.
* **Cenário 3: Renomear diretório**
  + **Usuário:** root
  + **Pré-condição:** mkdir /temp\_dir\_velho
  + **Comando:** mv /temp\_dir\_velho /temp\_dir\_novo
  + **Saída Esperada:** Diretório '/temp\_dir\_velho' movido e/ou renomeado para '/temp\_dir\_novo'.
* **Cenário 4: Mover diretório para outro diretório**
  + **Usuário:** root
  + **Pré-condição:** mkdir /projetoX. mkdir /pastas\_comuns.
  + **Comando:** mv /projetoX /pastas\_comuns/projetoX
  + **Saída Esperada:** Diretório '/projetoX' movido para '/pastas\_comuns/projetoX'.
* **Cenário 5: Tentar mover caminho de origem inexistente**
  + **Usuário:** root
  + **Comando:** mv /nao\_existe\_origem /algum\_destino
  + **Saída Esperada:** Erro: mv: '/nao\_existe\_origem': Nenhum arquivo ou diretório encontrado. (CaminhoNaoEncontradoException)
* **Cenário 6: Tentar mover para diretório pai de destino inexistente**
  + **Usuário:** root
  + **Pré-condição:** touch /temp\_file.txt
  + **Comando:** mv /temp\_file.txt /destino\_inexistente/novo\_nome.txt
  + **Saída Esperada:** Erro: Caminho de destino pai não existe para '/destino\_inexistente/novo\_nome.txt'. (CaminhoNaoEncontradoException)
* **Cenário 7: Tentar mover a raiz**
  + **Usuário:** root
  + **Comando:** mv / /nova\_raiz
  + **Saída Esperada:** Erro: Não é possível mover o diretório raiz. (PermissaoException)
* **Cenário 8: Tentar mover sem permissão de escrita no diretório pai de origem**
  + **Usuário:** root
  + **Pré-condição:** mkdir /source\_protected. touch /source\_protected/my\_doc.txt. chmod /source\_protected root other r-x (sem escrita para 'other').
  + **Trocar Usuário:** su func
  + **Comando:** mv /source\_protected/my\_doc.txt /tmp/new\_doc.txt (Assumindo /tmp tem escrita para func)
  + **Saída Esperada:** Erro: Permissão negada: não pode remover item em '/source\_protected' para o usuário 'func'. (PermissaoException)
* **Cenário 9: Tentar mover sem permissão de escrita no diretório pai de destino**
  + **Usuário:** root
  + **Pré-condição:** mkdir /dest\_protected. chmod /dest\_protected root other r-x. touch /meu\_doc\_src.txt.
  + **Trocar Usuário:** su func
  + **Comando:** mv /meu\_doc\_src.txt /dest\_protected/meu\_doc\_dest.txt (Assumindo func pode remover /meu\_doc\_src.txt)
  + **Saída Esperada:** Erro: Permissão negada: não pode escrever no diretório de destino '/dest\_protected' para o usuário 'func'. (PermissaoException)
* **Cenário 10: Tentar mover arquivo para destino onde já existe um diretório**
  + **Usuário:** root
  + **Pré-condição:** touch /my\_file. mkdir /my\_dir.
  + **Comando:** mv /my\_file /my\_dir
  + **Saída Esperada:** Arquivo '/my\_file' movido para '/my\_dir/my\_file'. (Neste caso, o mv move o arquivo *para dentro* do diretório. Se você tentasse mv /my\_file /my\_dir/my\_dir\_name, onde my\_dir\_name já é um diretório, o erro seria CaminhoJaExistenteException encapsulado em PermissaoException).
  + **Cenário 10.1: Tentar mover arquivo para nome que já é um diretório**
    - **Usuário:** root
    - **Pré-condição:** touch /arquivo\_src. mkdir /diretorio\_destino.
    - **Comando:** mv /arquivo\_src /diretorio\_destino
    - **Saída Esperada:** Arquivo '/arquivo\_src' movido para '/diretorio\_destino/arquivo\_src'. (Comportamento padrão: move para dentro)
* **Cenário 11: Tentar mover diretório para destino onde já existe um arquivo**
  + **Usuário:** root
  + **Pré-condição:** mkdir /dir\_src. touch /file\_dest.txt.
  + **Comando:** mv /dir\_src /file\_dest.txt
  + **Saída Esperada:** Erro: Não é possível mover diretório para um arquivo existente: '/file\_dest.txt'. (PermissaoException encapsulando CaminhoJaExistenteException)
* **Cenário 12: Tentar mover um diretório para dentro de um de seus subdiretórios**
  + **Usuário:** root
  + **Pré-condição:** mkdir /parent/child
  + **Comando:** mv /parent /parent/child
  + **Saída Esperada:** Erro: Não é possível mover um diretório para um de seus subdiretórios. (IllegalArgumentException)
* **Cenário 13: Mover arquivo sobre arquivo existente (sobrescrever)**
  + **Usuário:** root
  + **Pré-condição:** touch /file\_to\_move.txt. write /file\_to\_move.txt false "Conteudo original.". touch /target\_file.txt. write /target\_file.txt false "Conteudo a ser sobrescrito."
  + **Comando:** mv /file\_to\_move.txt /target\_file.txt
  + **Saída Esperada:** Arquivo '/file\_to\_move.txt' movido e renomeado para '/target\_file.txt'. (Conteúdo de /target\_file.txt deve ser "Conteudo original.").

## **7. cp (Copiar Arquivo/Diretório)**

**Objetivo:** Testar a cópia de itens, recursivamente e com validação de permissão.

* **Cenário 1: Copiar arquivo (origem para novo arquivo)**
  + **Usuário:** root
  + **Pré-condição:** touch /original.txt. write /original.txt false "Dados importantes."
  + **Comando:** cp /original.txt /copia.txt
  + **Saída Esperada:** Arquivo 'original.txt' copiado para '/copia.txt'.
* **Cenário 2: Copiar arquivo (origem para diretório existente)**
  + **Usuário:** root
  + **Pré-condição:** touch /documento.txt. mkdir /backup.
  + **Comando:** cp /documento.txt /backup
  + **Saída Esperada:** Arquivo 'documento.txt' copiado para '/backup/'. (Resulta em /backup/documento.txt)
* **Cenário 3: Copiar arquivo (origem para arquivo existente - sobrescrever)**
  + **Usuário:** root
  + **Pré-condição:** touch /src.txt. write /src.txt false "Conteudo A". touch /dest.txt. write /dest.txt false "Conteudo B"
  + **Comando:** cp /src.txt /dest.txt
  + **Saída Esperada:** Arquivo 'src.txt' copiado para '/dest.txt' (sobrescrito). (Verifique se dest.txt agora tem "Conteudo A")
* **Cenário 4: Copiar diretório (recursivo)**
  + **Usuário:** root
  + **Pré-condição:** mkdir /projeto\_antigo/src. touch /projeto\_antigo/main.c. touch /projeto\_antigo/src/helper.h. mkdir /novo\_repositorio.
  + **Comando:** cp /projeto\_antigo /novo\_repositorio
  + **Saída Esperada:** Diretório '/projeto\_antigo' e seu conteúdo copiados para '/novo\_repositorio'. (Resulta em /novo\_repositorio/projeto\_antigo com todo o conteúdo)
* **Cenário 5: Tentar copiar origem inexistente**
  + **Usuário:** root
  + **Comando:** cp /nao\_existe\_origem /destino
  + **Saída Esperada:** Erro: cp: '/nao\_existe\_origem': Nenhum arquivo ou diretório encontrado. (CaminhoNaoEncontradoException)
* **Cenário 6: Tentar copiar para diretório pai de destino inexistente**
  + **Usuário:** root
  + **Pré-condição:** touch /temp\_copy.txt
  + **Comando:** cp /temp\_copy.txt /inexistente\_dir/copia\_final.txt
  + **Saída Esperada:** Erro: cp: Destino '/inexistente\_dir/copia\_final.txt': Caminho pai não existe. (CaminhoNaoEncontradoException)
* **Cenário 7: Tentar copiar sem permissão de leitura na origem**
  + **Usuário:** root
  + **Pré-condição:** touch /dados\_senssiveis.txt. chmod /dados\_senssiveis.txt root other ---.
  + **Trocar Usuário:** su intruso
  + **Comando:** cp /dados\_senssiveis.txt /intruso\_home/dados\_roubados.txt
  + **Saída Esperada:** Erro: Permissão negada: não pode ler arquivo de origem '/dados\_senssiveis.txt' para o usuário 'intruso'. (PermissaoException)
* **Cenário 8: Tentar copiar sem permissão de escrita no destino**
  + **Usuário:** root
  + **Pré-condição:** mkdir /area\_restrita. chmod /area\_restrita root other r-x. touch /meu\_arquivinho.txt.
  + **Trocar Usuário:** su dev
  + **Comando:** cp /meu\_arquivinho.txt /area\_restrita/novo\_arquivo.txt
  + **Saída Esperada:** Erro: Permissão negada: não pode escrever no diretório de destino '/area\_restrita' para o usuário 'dev'. (PermissaoException)
* **Cenário 9: Tentar copiar diretório para um arquivo existente**
  + **Usuário:** root
  + **Pré-condição:** mkdir /source\_dir\_cp. touch /target\_file\_cp.txt.
  + **Comando:** cp /source\_dir\_cp /target\_file\_cp.txt
  + **Saída Esperada:** Erro: Não é possível copiar diretório para um arquivo existente: '/target\_file\_cp.txt'. (CaminhoJaExistenteException)
* **Cenário 10: Tentar copiar um diretório para dentro de um de seus subdiretórios**
  + **Usuário:** root
  + **Pré-condição:** mkdir /principal/sub
  + **Comando:** cp /principal /principal/sub
  + **Saída Esperada:** Erro: Não é possível copiar um diretório para um de seus subdiretórios. (IllegalArgumentException)

## **8. chmod (Alterar Permissões)**

**Objetivo:** Testar a alteração de permissões para arquivos e diretórios.

* **Cenário 1: Alterar permissão de um arquivo como dono**
  + **Usuário:** root
  + **Pré-condição:** touch /my\_script.sh. write /my\_script.sh false "echo 'Hello'"
  + **Comando:** chmod /my\_script.sh root rwx
  + **Saída Esperada:** Permissões de '/my\_script.sh' para o usuário 'root' alteradas para 'rwx'.
* **Cenário 2: Alterar permissão de um diretório como dono**
  + **Usuário:** root
  + **Pré-condição:** mkdir /shared\_folder.
  + **Comando:** chmod /shared\_folder root rwx
  + **Saída Esperada:** Permissões de '/shared\_folder' para o usuário 'root' alteradas para 'rwx'.
* **Cenário 3: Alterar permissão como root (em item que não é seu)**
  + **Usuário:** root
  + **Pré-condição:** Logue como user1 e crie /user1\_file.txt.
  + **Trocar Usuário:** su root
  + **Comando:** chmod /user1\_file.txt user1 r--
  + **Saída Esperada:** Permissões de '/user1\_file.txt' para o usuário 'user1' alteradas para 'r--'.
* **Cenário 4: Tentar alterar permissão de arquivo/diretório que não existe**
  + **Usuário:** root
  + **Comando:** chmod /nao\_existe.txt root rwx
  + **Saída Esperada:** Erro: chmod: '/nao\_existe.txt': Nenhum arquivo ou diretório encontrado. (CaminhoNaoEncontradoException)
* **Cenário 5: Tentar alterar permissão sem ser dono ou root**
  + **Usuário:** root
  + **Pré-condição:** touch /file\_owned\_by\_root.txt.
  + **Trocar Usuário:** su estranho
  + **Comando:** chmod /file\_owned\_by\_root.txt estranho rwx
  + **Saída Esperada:** Erro: Permissão negada: Somente o dono ou 'root' pode alterar permissões de '/file\_owned\_by\_root.txt'. (PermissaoException)
* **Cenário 6: Tentar alterar permissão com formato inválido**
  + **Usuário:** root
  + **Pré-condição:** touch /temp\_file\_perms.txt.
  + **Comando:** chmod /temp\_file\_perms.txt root rwx\_extra
  + **Saída Esperada:** Erro: Formato de permissão inválido. Use 'rwx', 'rw-', 'r-x', '---', etc. (IllegalArgumentException)