

## Relatório de Acompanhamento FINAL

### Equipe:

Nome: Lisiane de Aguiar Campos

Nro. Cartão: 00252739

Nome: Marcely Zanon Boito

Nro. Cartão: 00228454

### Atividades a serem executadas

1. Elaborar uma matriz em que cada linha corresponde a uma das funções da API de *T2FS* e com três colunas indicando se a função da API foi implementada (sim, não, parcial); se a função está correta (sim ou não) e, em caso de funcionamento parcial, uma breve descrição do o quê ela está fazendo e o quê ela não está fazendo.
2. Responder aos questionamentos abaixo listados (máximo 15 linhas).

### Questões a serem respondidas (entregáveis do relatório final)

1. Tabela de funcionamento das primitivas da API.

Primitiva da API	Implementada (Sim ou não)	Funcionamento (sim, não, parcial)	Em caso de implementada e não funcional, dizer o que NÃO está sendo feito pela primitiva.
<i>create2()</i>	sim	sim	
<i>delete2()</i>	sim	sim	
<i>open2()</i>	sim	sim	
<i>close2()</i>	sim	sim	
<i>read2()</i>	sim	sim	
<i>write2()</i>	sim	sim	
<i>truncate2()</i>	sim	sim	
<i>seek2()</i>	sim	sim	
<i>mkdir2()</i>	sim	sim	
<i>rmdir2()</i>	sim	sim	
<i>chdir2()</i>	sim	sim	
<i>getcwd2()</i>	sim	sim	
<i>opendir2()</i>	sim	sim	
<i>readdir2()</i>	sim	sim	
<i>closedir2()</i>	sim	sim	

## ESCOLHAS DE IMPLEMENTAÇÃO:

**read2:** Um read2 após um seek2(handle, -1) retornará -1.

**opendir2:** Como não foi definido um número máximo de arquivos no enunciado, utilizamos o mesmo que para arquivos: 10.

**chdir2 e getcwd2:** se o usuário fizer um chdir2 com um path relativo (por exemplo: ../../dir/dir2/./dir3/), o getcwd2 retornará a exata mesma string (não fazemos uma conversão para a string de caminho absoluto, mas o caminhamento de diretórios funciona perfeitamente).

**opendir2 e readdir2:** na abertura do diretório com opendir2, uma estrutura é alocada em memória contendo as entradas do diretório, por causa disso, se após abrir um diretório o usuário atualizar arquivos (delete2, write2, create2), essas informações não aparecerão na readdir2. Para acessá-las, é necessário abrir o diretório novamente, para a informação ser recuperada do disco.

**read2 e write2:** estou devolvendo -1 no caso do tamanho do buffer enviado ser menor que o size informado.

### 2. Principais dificuldades encontradas na realização do trabalho (max. 15 linhas).

De início, foi meio difícil planejar e entender como encaixar as coisas, e até mesmo testar o que estávamos fazendo. Também começar tardiamente nos afetou bastante, pois esse trabalho é BEM mais difícil que o anterior, e necessita de muita atenção e foco, principalmente no início, onde está tudo muito abstrato.

Algumas funções (principalmente a nossa função interna de percorrer diretórios) exigiam um maior tempo de depuração, o que fez o desenvolvimento do trabalho estagnar em alguns pontos.

De resto, ainda batalhamos um pouco com a programação C, mas agora nos sentimos muito mais preparadas para o trabalho de compiladores no próximo semestre. :)

### 3. Lições aprendidas: se fosse começar de novo, o que seria feito de forma diferente? (max. 15 linhas)

O desenvolvimento do trabalho devia ter sido iniciado na semana em que foi disponibilizado, para que as dúvidas fossem reunidas e para que existisse uma melhor divisão das tarefas.

ESTUDAR PARA A PROVA == ESTUDAR PARA O TRABALHO

### 4. Carga horária SEMANAL

Nome do componente/cartão	Horas gastas
Lisiane de Aguiar Campos	5h
Marcely Zanon Boito	0 nas primeiras 3 semanas / 30h nas últimas duas
Total de horas:	85

Nos desorganizamos bastante nesse trabalho, principalmente comigo (Marcely) ficando hospitalizada e perdendo uma semana de aula e duas provas, o que desestabilizou o nosso ritmo de trabalho até a metade da terceira semana.

### 5. Carga horária TOTAL (de todas semanas) dispendida por membro do grupo.

Nome do componente/cartão	Horas gastas	Percentual sobre total
Lisiane de Aguiar Campos	25h	30%
Marcelly Zanon Boito	60h	70%
Total de horas:	85	

EXTRA: Link do github: <https://github.com/mzboito/sisop2>