

UNIVERSIDADE ESTADUAL DE PONTA GROSSA  
SETOR DE CIÊNCIAS AGRÁRIAS E DE TECNOLOGIA  
DEPARTAMENTO DE INFORMÁTICA

EDUARDO LUIZ SCHADE SOARES

**AVALIAÇÃO DE TEMPO DE CRIAÇÃO E CÓPIA DE ARQUIVOS**

PONTA GROSSA

2018

## - Introdução

Para a realização deste trabalho, foram utilizados dois métodos de criação e cópia de arquivos - **Função** e **Syscall**. A linguagem utilizada foi C++. Com isso foi possível chegar a uma média de tempo necessária para a criação e cópia de arquivos de arquivos de tamanho: *1B, 1KB, 1MB e 1 GB*.

## - Configuração da Máquina

Intel Core i5-5200U 2,2 GHz (2 núcleos, 4 threads)  
8 GB de RAM DDR3 1600 MHz  
HDD 1TB 5400RPM  
Linux Deepin 64 Bits

## - Implementação da criação e cópia de arquivos

Para a criação e cópia de arquivos, foram utilizados dois métodos a partir da linguagem C++, **função** e **syscall**.

### Função para criação:

- filename - Nome do arquivo
- size - Tamanho do arquivo em Bytes
- c - Caracter para ser escrito no arquivo

```
file = fopen(fileName, "w");  
for(int i=0; i < size; i++) fputc(c, file);  
fclose(file);
```

### Syscall para criação:

- filename - Nome do arquivo
- size - Tamanho do arquivo em Bytes
- c - Caracter para ser escrito no arquivo

```
file=open(fileName,O_WRONLY|O_CREAT,S_IRUSR|S_IWUSR);  
for(int i=0; i < size; i++) write(file, &c, 1);  
close(file);
```

### Função para cópia:

- filename1 - Nome do arquivo de leitura;
- filename2 - Nome do arquivo de gravação;
- c - Caracter para ser escrito no arquivo;

```

file = fopen(fileName1, "r");
fileCopied = fopen(fileName2, "w");
while((ch=fgetc(file))!=EOF) fputc(ch,fileCopied);
fclose(file);
fclose(fileCopied);

```

### **Syscall para cópia:**

- filename1 - Nome do arquivo de leitura;
- filename2 - Nome do arquivo de gravação;
- c - Caracter para ser escrito no arquivo;

```

file = open(fileName1, O_RDONLY);
fileCopied=open(fileName2,O_WRONLY|O_CREAT,S_IRUSR|S_IWUSR);
while(read(file,&ch,1) == 1) write(fileCopied,&c,1);
close(file);
close(fileCopied);

```

### **- Cálculo do tempo**

Para a realização do cálculo do tempo decorrido na criação dos arquivos, foi utilizada uma função da linguagem C++ chamada *gettimeofday*.

### **Implementação:**

```

// Declaração da struct
struct timeval tempo1, tempo2;
struct timezone tzp;

// Pegar o tempo inicial
gettimeofday(&tempo1, &tzp);

// Rotina para ser realizado o cálculo do tempo

// Pegar o tempo final
gettimeofday(&tempo2, &tzp);

// Tempo é armazenado na variável tempo
tempo=(double) (tempo2.tv_sec-tempo1.tv_sec)+(((double)
(tempo2.tv_usec - tempo1.tv_usec))/1000000);

```

### **- Resultados**

Para obter-se o resultado do tempo de execução das funções de cópia e de criação, cada função fora executada um total de 5 vezes, após isso, obteve-se a média.

- **Função de criação - *Tempo em segundos***

	FUNÇÃO CRIAÇÃO			
	1 B	1 KB	1 MB	1 GB
Referência 1	0,00003	0,00009	0,03902	38,33714
Referência 2	0,00002	0,00002	0,03401	34,51507
Referência 3	0,00002	0,00002	0,04002	33,66538
Referência 4	0,00003	0,00002	0,04413	35,99270
Referência 5	0,00004	0,00003	0,04177	33,71204
Média	0,00003	0,00004	0,03979	35,24447

- **Syscall de criação - *Tempo em segundos, minutos e horas***

	SYSCALL CRIAÇÃO					
	1 B	1 KB	1 MB	1 GB / S	1 GB / M	1 GB / H
Referência 1	0,00006	0,03509	40,57507	32.548	542,47	9,04
Referência 2	0,00048	0,04506	46,62232	38.193	636,55	10,61
Referência 3	0,00008	0,04730	48,67812	39.877	664,62	11,08
Referência 4	0,00006	0,06068	54,65559	44.774	746,23	12,44
Referência 5	0,00008	0,04950	43,33337	35.499	591,64	9,86
Média	0,000152	0,047526	46,772894	38.178	636,30	10,61

- **Função de cópia - *Tempo em segundos***

	FUNÇÃO CÓPIA			
	1 B	1 KB	1 MB	1 GB
Referência 1	0,00009	0,00010	0,04999	67,77995
Referência 2	0,00011	0,00013	0,05326	68,58634
Referência 3	0,00010	0,00013	0,05144	59,69185
Referência 4	0,00006	0,00008	0,06260	66,13003
Referência 5	0,00013	0,00011	0,06539	54,11354
Média	0,000098	0,00011	0,056536	63,260342

- **Syscall de cópia - *Tempo em segundos, minutos e horas***

	SYSCALL CRIAÇÃO					
	1 B	1 KB	1 MB	1 GB / S	1 GB / M	1 GB / H
Referência 1	0,00057	0,04173	40,06190	32.819	546,98	9,12
Referência 2	0,00008	0,06559	46,26325	37.899	631,65	10,53
Referência 3	0,00008	0,05753	45,84180	37.554	625,89	10,43
Referência 4	0,00007	0,05592	41,05865	33.635	560,59	9,34
Referência 5	0,00007	0,05123	44,09470	36.122	602,04	10,03
Média	0,000174	0,0544	43,46406	35.606	593,43	9,89

- **Conclusão**

Através dos resultados de tempo obtidos nas execuções das funções e syscall, pode-se dizer que o método **função** é significativamente mais rápido que o método **syscall**, tanto para criação, quanto para a cópia de arquivos.

Enquanto a função fputc (função) trabalha criando arquivos de 1MB, em média, a cada meio segundo, o write (syscall), gera um arquivo de 1MB, em média, a cada 45 segundo. Assim, para gerar arquivos grandes esse método se torna bastante ineficiente.

Quanto a cópia de arquivos, ambos os métodos se mostraram ligeiramente mais lentos que a função de criação.