



PUC
CAMPINAS
PONTIFÍCIA UNIVERSIDADE CATÓLICA

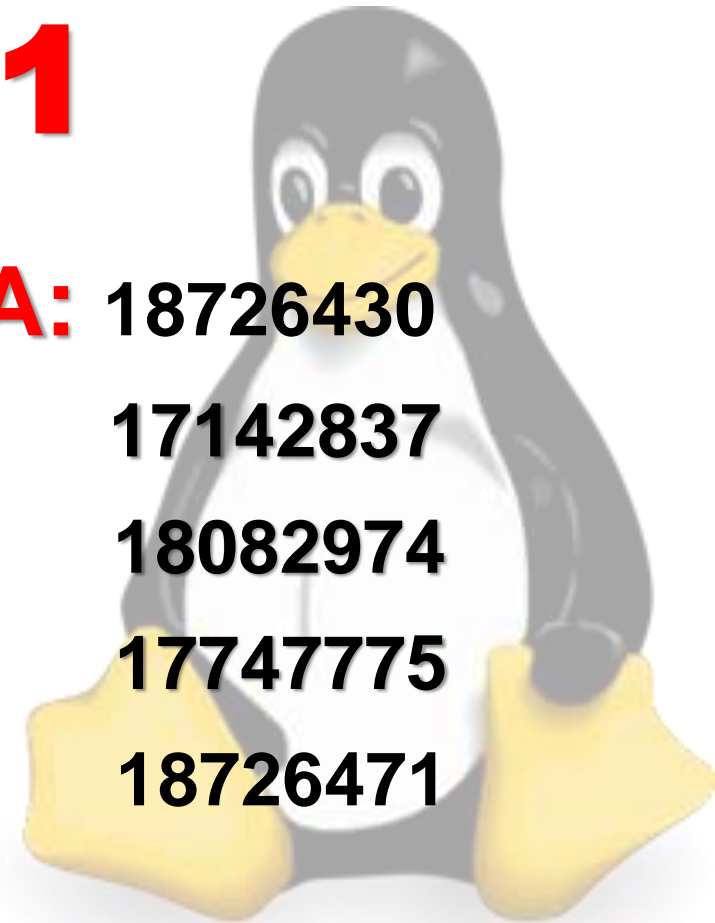
PONTIFÍCIA UNIVERSIDADE CATÓLICA DE CAMPINAS
CURSO DE ENGENHARIA DE COMPUTAÇÃO
21804 – SISTEMAS OPERACIONAIS A

EXPERIMENTO 1

EQUIPE: CHRISTOPHER OLIVEIRA
GIULIANO SANFINS
MATHEUS MORETTI
MURILO ARAUJO
VICTOR REIS

GRUPO NO. 03

RA: 18726430
17142837
18082974
17747775
18726471



INTRODUÇÃO

O experimento tem como objetivo aprender e se familiarizar com:

- ambiente Linux;
- compilador GCC;
- corrigindo o código;
- compreensão do funcionamento e criação de processos pai e filhos;
- desempenho do programa;
- *Funções: `fork()`, `exec()`, `wait()`, `usleep()` e `exit()`.*





ERROS E SOLUÇÕES NO PROGRAMA EXEMPLO

```
111  */
112  INT RTN;
113  rtn = 1;
114  for( count = 0; count < NO_OF_CHILDREN; count+- ) {
115      if( rtn == 0 ) {
116          rtn = fork();
117      } else {
118          break;
119      }
120  }
121  /*
122  * Verifica-se rtn para determinar se o processo eh pai ou filho
123  */
124
```

COUNT ++

RTN != 0



ERROS E SOLUÇÕES NO PROGRAMA EXEMPLO

```
168      */
169
170     } else {
171         /*
172         * Sou pai, aguardo o termino dos filhos
173         */
174         for( count = 0; count > NO_OF_CHILDREN; count++ ) {
175             wait(NULL);
176         }
177     }
178
179     exit(0);
180 }
181
```

A red box highlights the comparison operator `>` in the `for` loop condition on line 174. A red arrow points from this box to a red `<` symbol, indicating the correct comparison operator should be `<`.



PUC
CAMPINAS
PONTIFÍCIA UNIVERSIDADE CATÓLICA

CÓDIGO FONTE: CARGA

```
1  #include <stdio.h>
2
3
4  int main(){
5      long long int count=0;
6
7      while(1){
8          count++;
9
10     }
11
12     return 0;
13 }
14
```



PUC
CAMPINAS
PONTIFÍCIA UNIVERSIDADE CATÓLICA

RESPOSTAS ÀS QUESTÕES

➤ Pergunta 1

O que o compilador *gcc* faz com o arquivo *.h*, cujo nome aparece após o *include*?

R: Os arquivos fonte contém diretivas do tipo *#include* e *#define* que são processadas pelo pré-processador ***c*pp**, que faz a expansão dos macros, a inclusão dos arquivos com cabeçalhos, e remove os comentários. A saída do ***c*pp** é entregue ao compilador ***g*cc**, que faz a tradução de C para linguagem de montagem.

RESPOSTAS ÀS QUESTÕES

➤ Pergunta 2

Apresentar (parcialmente) e explicar o que há em `<stdio.h>`.

R: *stdio.h* é um cabeçalho da biblioteca padrão do C, "cabeçalho padrão de entrada/saída". Possui definições de sub-rotinas relativas às operações de entrada/saída, como leitura de dados digitados no teclado e exibição de informações na tela do programa de computador. Também possui numerosas definições de constantes, variáveis e tipos. É um dos cabeçalhos mais populares da linguagem de programação C, intensivamente utilizado tanto por programadores iniciantes como por experientes.

Abaixo temos 4 funções desta biblioteca que são muito utilizadas:

- printf()* Função usada para imprimir dados na tela.
- scanf()* Função usada para capturar dados do usuário.
- fprintf()* Função usada para imprimir dados em arquivo.
- fscanf()* Função usada para ler dados de arquivos.

RESPOSTAS ÀS QUESTÕES

➤ Pergunta 3

Qual é a função da diretiva include (linha que começa com #), com relação ao compilador?

R: O que o compilador faz é simplificarmente o mesmo que você copiar e colar o texto que está dentro do include para dentro do texto que está na fonte principal. Ele está importando todo seu conteúdo. Normalmente esses conteúdos são apenas declarações de estruturas de dados (classes, estruturas, enumerações), constantes, cabeçalho de funções/métodos, macros e eventualmente algum código quando se deseja que uma função seja importada direto na fonte ao invés de ser chamada (*source inline*). O mais comum é ele não ter código. Normalmente ele é usado para declarar as estruturas de dados, mas não definir os algoritmos.

RESPOSTAS ÀS QUESTÕES

➤ Pergunta 4

O que são e para que servem *argc* e *argv*? Não esqueça de considerar o * antes de *argv*.

R: ***argc*** – é um valor inteiro que indica a quantidade de argumentos que foram passados ao chamar o programa.

argv – é um vetor de char que contém os argumentos, um para cada *string* passada na linha de comando.

RESPOSTAS ÀS QUESTÕES

➤ Pergunta 5

Qual a relação: entre SLEEP_TIME e o desvio, nenhuma, direta * ou indiretamente proporcional?

R: O desvio é um método para medir a dispersão e analisar o quanto os valores de um conjunto se afastam de uma regularidade. E verificar o quanto os elementos de um conjunto respeitam um padrão. Em nosso experimento é analisado a partir do tempo de execução, ou seja, o *sleep* está diretamente ligado a este valor, aumentando este ou diminuindo afeta proporcionalmente o desvio, em um caso isolado, porém ele não é o único fator implicante no desvio, há fatores externos na própria máquina, como outros processos utilizando do processador, causando ainda mais sobrecarga afetando assim ainda mais o valor do desvio no final, portanto pode se concluir que o *sleep* é apenas um fato diretamente ligado ao desvio, mas não determinante.

RESPOSTAS ÀS QUESTÕES

➤ Pergunta 1

Apresente a linha de comando para compilar o programa exemplo, de tal maneira que o executável gerado receba o nome de "experimento1" (sem extensão).

R: GCC exemplo.c -o experimento1.

Após a execução do programa sem nenhum erro, foi executado o programa 10 vezes, variando a carga em cinco em cinco, começando inicialmente com nenhuma carga e finalizando a análise da execução com 45 cargas. Abaixo pode ser observado uma tabela demonstrativa que indica a variação das cargas e o valor do desvio total e médio em segundos de cada filho do processo.



PUC
CAMPINAS
PONTIFÍCIA UNIVERSIDADE CATÓLICA

RESPOSTAS ÀS QUESTÕES

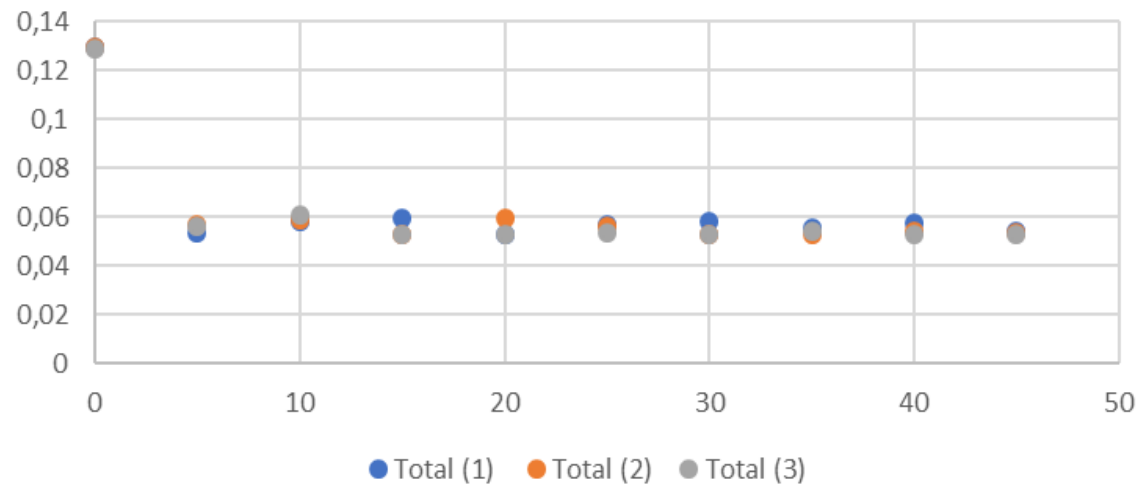
	Filho 1		Filho 2		Filho 3	
Carga	Total	Médio	Total	Médio	Total	Médio
0	0,12931	0,00012931	0,12923	0,00012923	0,12915	0,00012915
5	0,05359	0,00005359	0,05687	0,00005687	0,05601	0,00005601
10	0,05796	0,00005796	0,05849	0,00005849	0,06073	0,00006073
15	0,05949	0,00005949	0,0526	0,0000526	0,05281	0,00005281
20	0,05268	0,00005268	0,05914	0,00005914	0,05281	0,00005281
25	0,05704	0,00005704	0,05636	0,00005636	0,05341	0,00005341
30	0,058	0,000058	0,05281	0,00005281	0,05281	0,00005281
35	0,05537	0,00005537	0,05261	0,00005261	0,05393	0,00005393
40	0,05738	0,00005738	0,05378	0,00005378	0,0528	0,0000528
45	0,05395	0,00005395	0,05371	0,00005371	0,0528	0,0000528



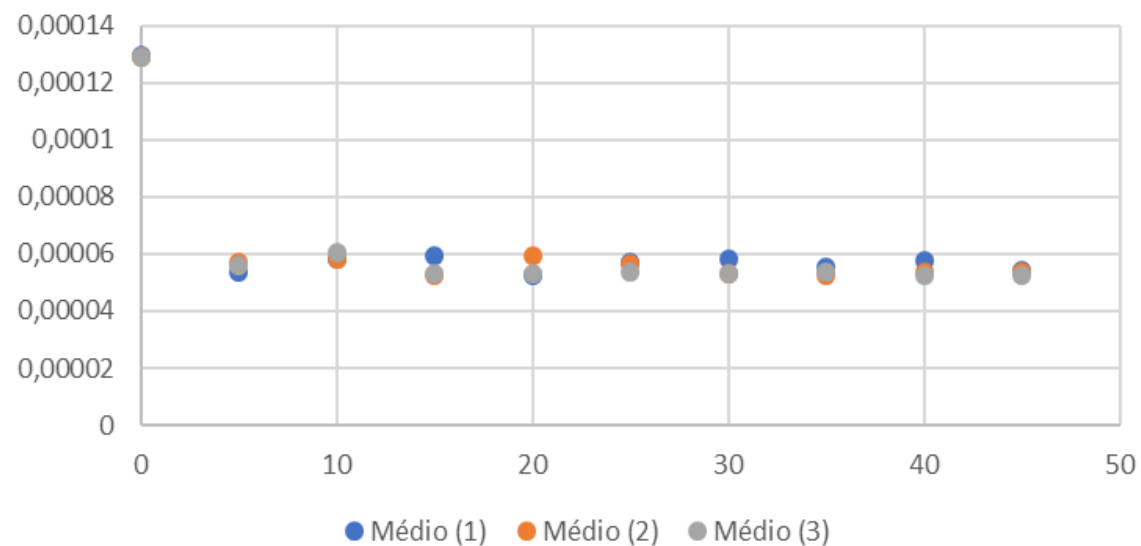
PUC
CAMPINAS
PONTIFÍCIA UNIVERSIDADE CATÓLICA

RESPOSTAS ÀS QUESTÕES

Desvio Total dos Filhos (Duração X Carga)



Desvio Médio dos Filhos (Duração X Carga)



RESPOSTAS ÀS QUESTÕES

➤ Pergunta 2

Descreva o efeito da diretiva &.

R: Permite que o programa execute em segundo plano (*Background*), deixando o prompt de comando livre para outros comandos.

RESPOSTAS ÀS QUESTÕES

➤ Pergunta 3

Qual é o motivo do uso do “./”?

Explique porque a linha de comando para executar o `gcc` não requer o “./”.

R: Usar “./” não tem a ver com o *shell script*. O Linux procura executáveis (*shell scripts* ou quaisquer outros) na variável de ambiente `PATH`. Então, se seu executável não estiver no `PATH`, o Linux não executa. O “ponto-barra” serve para indicar o caminho do executável: o ponto refere-se ao diretório atual e a barra, naturalmente, separa o diretório do nome do arquivo.

Então, a regra é: ao executar qualquer programa, se o mesmo estiver no `PATH`, basta usar o nome do executável. Caso contrário, é necessário indicar o caminho completo que, no caso do diretório atual, é “./<Executável>”.

RESPOSTAS ÀS QUESTÕES

➤ Pergunta 4

Apresente as características da CPU do computador usado no experimento.





RESPOSTAS ÀS QUESTÕES

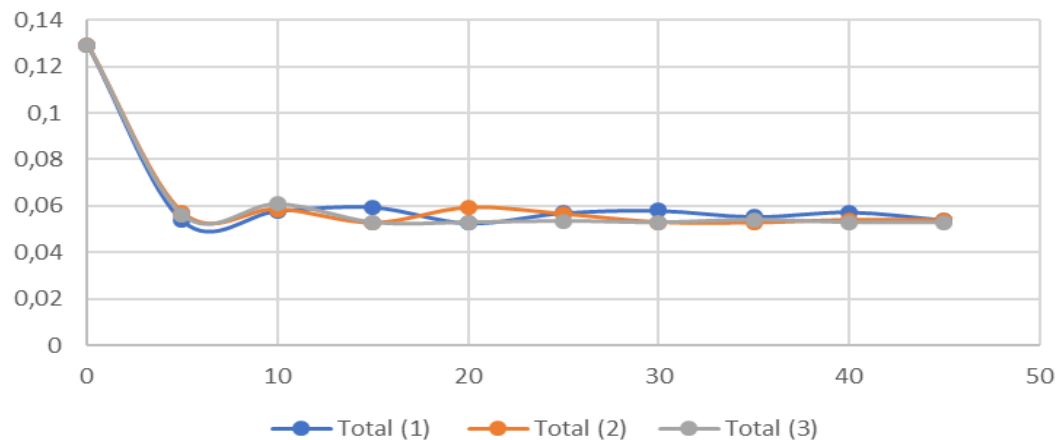
Filho / Carga	Total (seg)	Média (seg)
1/0	0,12931	0,00012931
2/0	0,12923	0,00012923
3/0	0,12915	0,00012915
1/5	0,05359	0,00005359
2/5	0,05687	0,00005687
3/5	0,05601	0,00005601
1/10	0,05796	0,00005796
2/10	0,05849	0,00005849
3/10	0,06073	0,00006073
1/15	0,05949	0,00005949
2/15	0,0526	0,0000526
3/15	0,05281	0,00005281
1/20	0,05268	0,00005268
2/20	0,05914	0,00005914
3/20	0,05281	0,00005281

1/25	0,05704	0,00005704
2/25	0,05636	0,00005636
3/25	0,05341	0,00005341
1/30	0,058	0,000058
2/30	0,05281	0,00005281
3/30	0,05281	0,00005281
1/35	0,05537	0,00005537
2/35	0,05261	0,00005261
3/35	0,05393	0,00005393
1/40	0,05738	0,00005738
2/40	0,05378	0,00005378
3/40	0,0528	0,0000528
1/45	0,05395	0,00005395
2/45	0,05371	0,00005371
3/45	0,0528	0,0000528

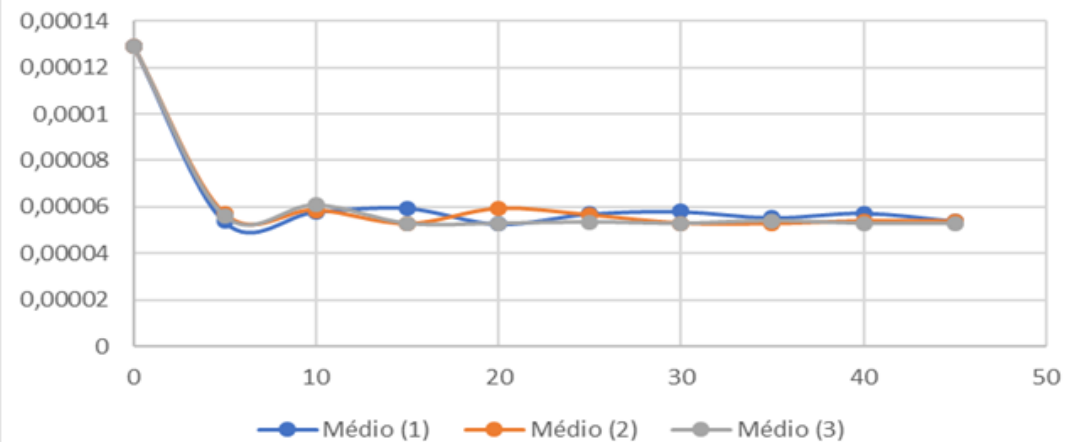


RESPOSTAS ÀS QUESTÕES

Comparação Desvio Total dos Filhos (Duração X Carga)



Comparação Desvio Médio dos Filhos (Duração X Carga)

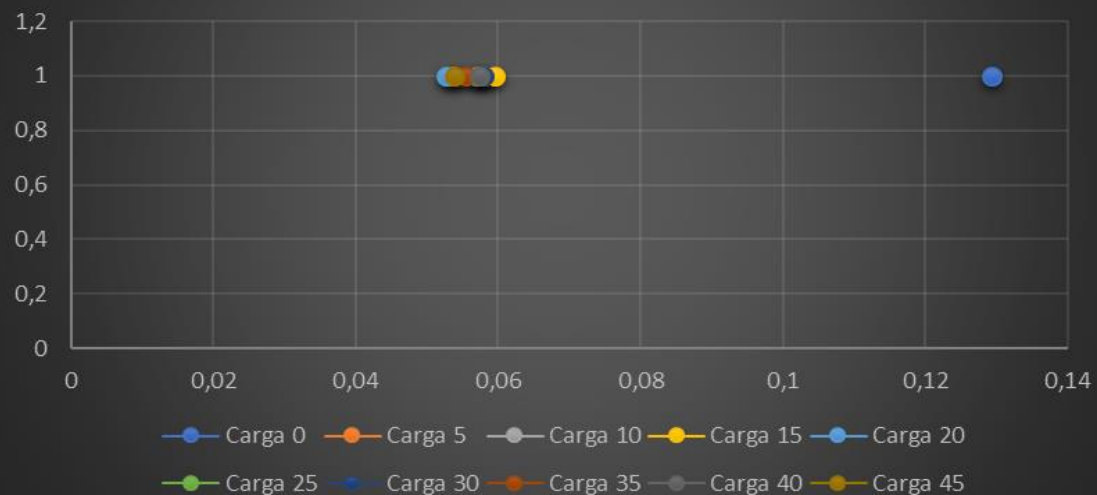




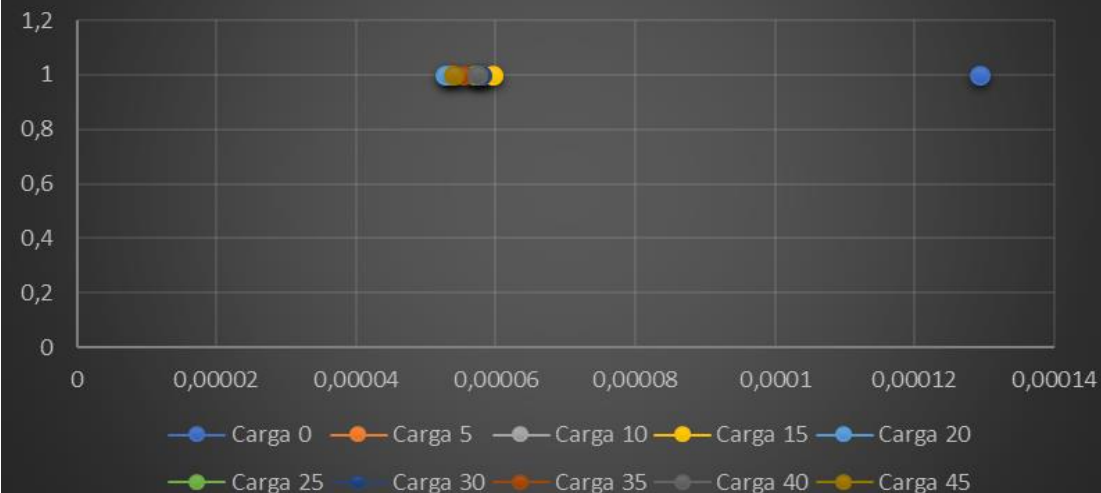
PUC
CAMPINAS
PONTIFÍCIA UNIVERSIDADE CATÓLICA

RESPOSTAS ÀS QUESTÕES

Número de cargas iguais dos três filhos
(Total)



Número de cargas iguais dos três filhos
(Médio)





RESPOSTAS ÀS QUESTÕES

“NO_OF_ITERATIONS”, de 1000, para 2000.
“SLEEP_TIME”, de 1000, para 2000.

Carga	Filho 1		Filho 2		Filho 3	
	Total	Médio	Total	Médio	Total	Médio
0	0,2726	0,0001363	0,27251	0,00013626	0,27244	0,00013622
20	0,10801	0,000054	0,18655	0,00009328	0,10813	0,00005407
40	0,15577	0,00007789	0,15546	0,0000773	0,15451	0,00007725

Carga	Filho 1		Filho 2		Filho 3	
	Total	Médio	Total	Médio	Total	Médio
0	0,12931	0,00012931	0,12923	0,00012923	0,12915	0,00012915
20	0,05268	0,00005268	0,05914	0,00005914	0,05281	0,00005281
40	0,05738	0,00005738	0,05378	0,00005378	0,0528	0,0000528

Carga	Total (CA 1)	Total (CA 2)	Total (CA 3)	Total (SA 1)	Total (SA 2)	Total (SA 3)
0	0,2726	0,27251	0,27244	0,12931	0,12923	0,12915
20	0,10801	0,18655	0,10813	0,05268	0,05914	0,05281
40	0,15577	0,15546	0,15451	0,05738	0,05378	0,0528

RESPOSTAS ÀS QUESTÕES

➤ Pergunta 5

O que significa um processo ser determinístico?

R: Dada uma certa entrada, ela produzirá sempre a mesma saída, com a máquina responsável sempre passando pela mesma sequência de estados.

RESPOSTAS ÀS QUESTÕES

➤ Pergunta 6

Como é possível conseguir as informações de todos os processos existentes na máquina, em um determinado instante?

R: Executando o comando “*ps -aux (a- all with tty, including other users; u- effective user id or name; x- register format)*”.



PUC
CAMPINAS
PONTIFÍCIA UNIVERSIDADE CATÓLICA

RESULTADOS E ANÁLISE





PUC
CAMPINAS
PONTIFÍCIA UNIVERSIDADE CATÓLICA

CONCLUSÃO

