

Documentação

Desenvolvedores: Lucas Silva e Nicácio Arruda

Sistema Operacional Utilizado:

Desenvolvimento: Debian 8 “Jessie” 64 bits

Testes: Ubuntu 15.10 64 bits

Compilador: GCC (GNU Compiler Collection) Versão 4.2.9

Editor de texto: Atom

Controle de versão: Git

Instruções de uso: Para executar o programa é necessário que o sistema operacional tenha o compilador GCC instalado. Após baixar o projeto, abra o terminal e entre na pasta do mesmo. Dentro da pasta trabalho03 execute o comando ‘make’ e após o término da compilação, execute ./main acompanhado do nome do arquivo em que deseja salvar os dados gerados do programa, da seguinte forma:

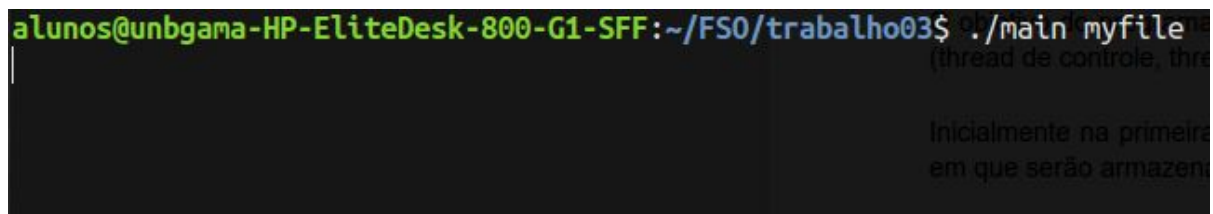


Imagem 1 - Tela inicial

Descrição do Programa:

O objetivo do programa é lançar um processo que irá disparar três tipos de threads (thread de controle, thread produtora e thread consumidora).

Inicialmente ao executar o programa o usuário irá informar o nome do arquivo em que serão armazenadas as mensagens de registro. Caso o usuário não passe o nome por parâmetro, a seguinte mensagem será exibida:

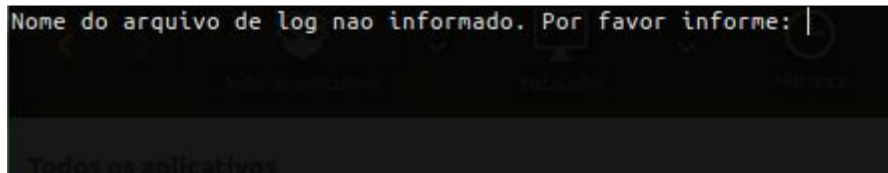


Imagem 2 - Mensagem para inserir o nome do arquivo

Após a criação do arquivo, o programa irá criar as threads, neste caso serão criadas uma thread de controle, uma thread produtora e duas threads consumidoras(a,b). A thread produtora será responsável por produzir inteiros aleatórios e registrar cada um no arquivo. As mensagens gravadas pela thread produtora estarão no arquivo no seguinte formato:

```
1 [producao]: Numero gerado: -436520072
2 [producao]: Numero gerado: 548483589
```

Imagem 3 - Mensagens gravadas pela thread produtora

Enquanto que as threads consumidoras irão ler as informações produzidas pela thread produtora e armazenar no arquivo. Além disso, as threads consumidoras também devem armazenar o maior e o menor número inteiro gerado durante a execução da aplicação; e quantos inteiros foram gerados e armazenados para aguardar pelo consumo. As mensagens gravadas pelas threads consumidoras estarão no arquivo no seguinte formato:

```
3 [consumo a]: Numero lido: 548483589
4 [consumo b]: Numero lido: -436520072
```

Imagem 4 - Mensagens gravadas pelas threads consumidoras

A sequência de mensagens escritas pelas threads produtoras e consumidoras seguirão o seguinte formato (podem haver pequenas alterações):

```
1 [producao]: Numero gerado: -689980853
2 [consumo a]: Numero lido: -689980853
3 [producao]: Numero gerado: 475932365
4 [consumo a]: Numero lido: 475932365
5 [producao]: Numero gerado: -1426364099
6 [consumo a]: Numero lido: -1426364099
7 [producao]: Numero gerado: -1116540274
8 [producao]: Numero gerado: 836358185
9 [consumo a]: Numero lido: 836358185
10 [consumo b]: Numero lido: -1116540274
11 [producao]: Numero gerado: -45508687
12 [consumo a]: Numero lido: -45508687
13 [producao]: Numero gerado: 551249390
```

Imagem 5 - Imagem do arquivo gerado

A thread de controle irá monitorar o que o usuário digitar no terminal durante a execução do programa. Quando o usuário digitar CTRL+C na linha de comando, a thread de controle deve registrar no final do arquivo o maior e o menor número gerado, a maior ocupação no buffer e uma mensagem de encerramento da aplicação. O final do arquivo será semelhante a imagem abaixo:

```
6597 [producao]: Numero gerado: -936367298
6598 [consumo a]: Numero lido: -936367298
6599 [producao]: Numero gerado: 1589788364
6600 [producao]: Numero gerado: 1034273887
6601 [consumo a]: Numero lido: 1034273887
6602 [consumo b]: Numero lido: 1589788364
6603 [producao]: Numero gerado: -653285261
6604 [consumo a]: Numero lido: -653285261
6605 [producao]: Numero gerado: -560480383
6606 [producao]: Numero gerado: 734273960
6607 [consumo a]: Numero lido: 734273960
6608 [consumo b]: Numero lido: -560480383
6609 [producao]: Numero gerado: -1928005983
6610 [consumo a]: Numero lido: -1928005983
6611 [producao]: Numero gerado: -52736974
6612 [aviso]: Termino solicitado. Aguardando threads...
6613 [aviso]: Maior numero gerado: 2070009761
6614 [aviso]: Menor numero gerado: -2110828502
6615 [aviso]: Maior ocupacao de buffer: 2
6616 [aviso]: Aplicacao encerrada.
```

Imagem 6 - Mensagens gravadas pela thread de controle

Além de registrar no arquivo essas mensagens, elas também devem ser exibidas na tela. aplicação deverá mostrar uma mensagem na tela semelhante a imagem abaixo:

```
[aviso]: Termino solicitado. Aguardando threads...
[aviso]: Maior numero gerado: 1954947578
[aviso]: Menor numero gerado: -2093718231
[aviso]: Maior ocupacao de buffer: 2
[aviso]: Aplicacao encerrada.
```

Imagem 7 - Imagem do término do programa

Casos de teste

Para testar o programa, inicialmente executou-se o programa por aproximadamente 15 segundos e acrescentando 5 segundos após cada execução até o total de 25 segundos. Para finalizar executou-se o programa por 3 minutos. Durante a execução, procurou-se minimizar o consumo de CPU, deixando apenas o terminal aberto. Os resultados obtidos podem ser visualizados abaixo:

```
329 [producao]: Numero gerado: 126970468
330 [consumo a]: Numero lido: 126970468
331 [producao]: Numero gerado: 507029967
332 [producao]: Numero gerado: 202581574
333 [aviso]: Termino solicitado. Aguardando threads...
334 [aviso]: Maior numero gerado: 1788266826
335 [aviso]: Menor numero gerado: -1873717542
336 [aviso]: Maior ocupacao de buffer: 2
337 [aviso]: Aplicacao encerrada.
```

Imagem 8 - Resultados obtidos após 15 segundos

```
433 [producao]: Numero gerado: 402067735
434 [producao]: Numero gerado: 271488844
435 [consumo b]: Numero lido: 271488844
436 [consumo a]: Numero lido: 402067735
437 [producao]: Numero gerado: -705134535
438 [consumo a]: Numero lido: -705134535
439 [producao]: Numero gerado: -138698675
440 [producao]: Numero gerado: 199298272
441 [aviso]: Termino solicitado. Aguardando threads...
442 [aviso]: Maior numero gerado: 1886138224
443 [aviso]: Menor numero gerado: -1944952836
444 [aviso]: Maior ocupacao de buffer: 2
445 [aviso]: Aplicacao encerrada.
```

Imagem 9 - Resultados obtidos após 20 segundos

```
495 [consumo b]: Numero lido: -1047280651
496 [consumo a]: Numero lido: 83925849
497 [producao]: Numero gerado: -397393297
498 [consumo a]: Numero lido: -397393297
499 [producao]: Numero gerado: -204836347
500 [producao]: Numero gerado: -1216275015
501 [consumo a]: Numero lido: -1216275015
502 [consumo b]: Numero lido: -204836347
503 [aviso]: Termino solicitado. Aguardando threads...
504 [aviso]: Maior numero gerado: 2033910083
505 [aviso]: Menor numero gerado: -2004373293
506 [aviso]: Maior ocupacao de buffer: 2
507 [aviso]: Aplicacao encerrada.
```

Imagem 10 - Resultados obtidos após 25 segundos

```
3625 [producao]: Numero gerado: -525885281
3626 [consumo b]: Numero lido: -525885281
3627 [producao]: Numero gerado: -1079153790
3628 [producao]: Numero gerado: 1595292006
3629 [consumo b]: Numero lido: 1595292006
3630 [consumo a]: Numero lido: -1079153790
3631 [producao]: Numero gerado: 908120691
3632 [aviso]: Terminou solicitado. Aguardando threads...
3633 [aviso]: Maior numero gerado: 2030885227
3634 [aviso]: Menor numero gerado: -2114938300
3635 [aviso]: Maior ocupacao de buffer: 2
3636 [aviso]: Aplicacao encerrada.
```

Imagem 11 - Resultados obtidos após 3 minutos

Detalhes técnicos:

| Máquina de testes | |
|----------------------|---------------------------------------|
| Memória Ram: | 7,7GB |
| Processador: | Intel Core I5-4570 CPU 3.20GHZ x 4 |
| Sistema Operacional: | Ubuntu 15.10, versão 64 bits |