



UNIVERSIDADE FEDERAL
DO RIO DE JANEIRO

SIMULAÇÃO DE ESCALONAMENTO DE PROCESSOS

Departamento de Ciência da Computação
Sistemas Operacionais I – 2019/1
Trabalho 1

Alunos: Matheus Panno - 116036387
Thiago Coelho - 116016400

1. Introdução

Round-robin (RR) é um dos algoritmos empregados por escalonadores de processo e de rede, em computação. Fatias de tempo iguais de uso do processador são atribuídas a cada processo e em ordem circular, manipulando todos os processos sem prioridade. É usado em projetos de sistemas operacionais multitarefa, e foi projetado especialmente para sistemas time-sharing, pois este algoritmo depende de um temporizador.

2. Objetivo

Desenvolver um simulador do escalonador de processos de um sistema operacional que utiliza o algoritmo de Round-robin. O programa simulará um ambiente concorrente, com vários processos sendo intercalados entre si.

3. Definições e Premissas:

- **Processo:**

- É representado no simulador por uma *struct* contendo:
 - pid
 - ppid
 - prioridade
 - status (NEW, READY, RUNNING, BLOCKED e EXIT)
 - lista de IO
 - tempo de início
 - tempo de serviço
 - além de outros atributos de controle para o funcionamento do simulador
- Foi definido um número fixo de processos igual a 10.
- O número de IO por processo é randômico, com o limite de até 3.
- O tempo de início dos processos é randômico. O primeiro processo inicia até, no máximo, o tempo 10. Os processos seguintes iniciam em até 10 unidades de tempo após o processo anterior.
- Processos podem ter tempo de início iguais.
- A quantidade e os tipos de IO que o processo fará são escolhidos de forma randômica.
- O tempo de serviço dos processos é randômico, sendo limitado entre 2 a 20 ut.
- Um processo não tem mais de um IO executando no mesmo tempo, isso é controlado na criação dos processos.

- **IO:**

- É representado no simulador por uma *struct* contendo:
 - Tipo
 - Tempo de duração
 - Prioridade da fila para qual o processo retorna após o fim do IO
- Existem 3 tipos de IO: Disco, Impressora e Fita Magnética
- O tempo de duração dos IO's é fixo:
 - Disco: 5 ut

- Impressora: 12 ut
- Fita Magnética: 8 ut

- **Aspectos gerais do Simulador:**

- Foi definido uma fatia de tempo fixa igual a 4.
- Existe uma variável que representa o tempo atual. Ela é incrementada a cada iteração do simulador.
- Há 5 filas diferentes:
 - Fila de alta e de baixa prioridade. São utilizadas para guardar os processos que estão prontos e serão posteriormente colocados em execução.
 - Uma fila para cada tipo de IO, sendo três no total. Cada fila guarda os processos que estão fazendo determinado tipo de IO. São utilizadas para saber qual processo deve retornar do IO em determinada unidade de tempo.

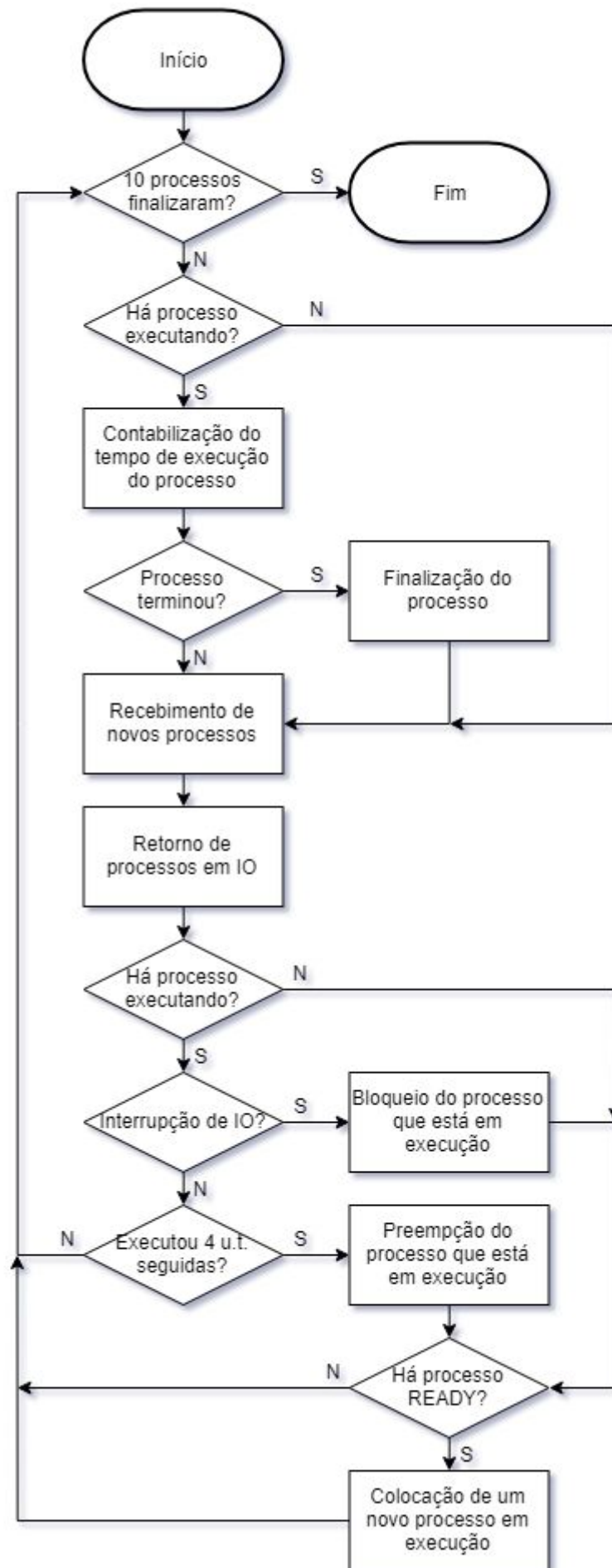
4. Fluxo da Simulação

O programa consiste de um loop principal que simula cada unidade de tempo. O loop será finalizado quando todos os processos agendados terminarem. A cada iteração, são feitas, em ordem, as seguintes operações:

- Contabilização do tempo de execução do processo que está rodando atualmente, se houver algum, e verificação do seu término. Se o processo terminar, seu status será alterado para EXIT;
- Recebimento de novos processos, colocando-os na fila de alta prioridade. Novos processos terão status igual a NEW e quando forem colocados na fila, terão seu status alterado para READY;
- Recebimento de processos que retornaram do IO, colocando-os na fila de prioridade correspondente ao IO do qual ele retornou. Isso é verificado observando o primeiro processo na fila do IO em questão. Essa lógica foi possível porque o cada dispositivo tem um tempo fixo de execução, então o primeiro processo a entrar em sua fila também será o primeiro a sair. Ao retornar do IO para alguma fila de execução, o processo terá seu status alterado para READY;
- Tratamento de interrupção de IO do processo que está em execução, caso ela ocorra no tempo em questão. Nesse caso, o processo será bloqueado (seu status será alterado para BLOCKED), tirado de execução e colocado na fila do IO em questão.
- Caso o processo em execução tenha executado o tempo máximo do *time slice*, ele sofrerá preempção. Ao sofrer essa preempção, seu status será alterado para READY e ele será colocado na fila de baixa prioridade.
- Escalonamento de um novo processo, caso seja possível. O novo processo é buscado primeiramente na fila de alta prioridade e, caso esta fila esteja vazia, será procurado na de baixa prioridade e, caso esta também esteja vazia, nenhum processo será colocado em execução. Processos colocados em execução terão seu status alterado para RUNNING.

Fluxograma

Fluxo do loop principal do simulador de escalonamento construído



5. Resultados

A saída do programa consiste de 3 arquivos:

- log_execucao.txt: Contém o log completo da execução do simulador. A cada momento, diz qual era o processo que estava em execução no início daquela unidade de tempo. Depois informa todas as operações que ocorreram naquela unidade de tempo, como processos novos chegando, processo terminando execução, interrupção de IO, preempção, etc. Por fim, as 5 filas existentes no simulador são impressas no estado em que estavam ao fim da unidade de tempo em questão.
- log_turnarounds.txt: Contém os *turnarounds* dos processos.
- log_init_val.txt: Contém os dados iniciais dos processos, como: pid, ppid, tempo de início, tempo de serviço, tipo de IO que irá executar, etc.

Alguns exemplos de captura de tela dos arquivos de log:

- Dados iniciais de um processo:

```
Processo PID: 100
PPID: 1
Prioridade: 2
Tempo Inicio: 0
Tempo Servico: 5
Tipo IO: mag_tape
Tempo de inicio do IO: 4
Tipo IO: printer
Tempo de inicio do IO: 2
```

- Processo em IO (A saída foi alterada, pois havia muitas unidades de tempo)

```
-----
Tempo 47: O processo de pid = 108 esta em execucao
```

```
- Processo de pid = 108 sofreu uma interrupcao de IO do tipo mag_tape e foi bloqueado
- Processo de pid = 101 saiu da fila de baixa prioridade e foi colocado em execucao
Fila de alta prioridade:      [  ]
Fila de baixa prioridade:    [ 104 | 105 | 106 | 103 ]
Fila de IO do tipo disk:     [  ]
Fila de IO do tipo printer:  [  ]
Fila de IO do tipo mag_tape: [ 107 | 108 ]
```

```
-----
.
.
.
```

```
-----
Tempo 55: O processo de pid = 107 esta em execucao
```

```
- Processo de pid = 107 terminou sua execucao
- Processo de pid = 108 voltou de um IO do tipo mag_tape e foi colocado na fila de alta prioridade
- Processo de pid = 109 saiu da fila de alta prioridade e foi colocado em execucao
Fila de alta prioridade:      [ 108 ]
Fila de baixa prioridade:    [ 104 | 105 | 106 | 103 | 101 ]
Fila de IO do tipo disk:     [  ]
Fila de IO do tipo printer:  [  ]
Fila de IO do tipo mag_tape: [  ]
```

- Processo sofrendo preempção

```

-----
Tempo 60: O processo de pid = 104 esta em execucao

Fila de alta prioridade:    [ ]
Fila de baixa prioridade:  [ 106 | 103 ]
Fila de IO do tipo disk:   [ ]
Fila de IO do tipo printer: [ 102 ]
Fila de IO do tipo mag_tape: [ 109 ]

-----

Tempo 61: O processo de pid = 104 esta em execucao

Fila de alta prioridade:    [ ]
Fila de baixa prioridade:  [ 106 | 103 ]
Fila de IO do tipo disk:   [ ]
Fila de IO do tipo printer: [ 102 ]
Fila de IO do tipo mag_tape: [ 109 ]

-----

Tempo 62: O processo de pid = 104 esta em execucao

Fila de alta prioridade:    [ ]
Fila de baixa prioridade:  [ 106 | 103 ]
Fila de IO do tipo disk:   [ ]
Fila de IO do tipo printer: [ 102 ]
Fila de IO do tipo mag_tape: [ 109 ]

-----

Tempo 63: O processo de pid = 104 esta em execucao

- Processo de pid = 104 executou o maximo de unidades de tempo seguidas (4) e foi colocado na fila de baixa prioridade
- Processo de pid = 106 saiu da fila de baixa prioridade e foi colocado em execucao
Fila de alta prioridade:    [ ]
Fila de baixa prioridade:  [ 103 | 104 ]
Fila de IO do tipo disk:   [ ]
Fila de IO do tipo printer: [ 102 ]
Fila de IO do tipo mag_tape: [ 109 ]

-----

Tempo 64: O processo de pid = 106 esta em execucao

- Processo de pid = 109 voltou de um IO do tipo mag_tape e foi colocado na fila de alta prioridade
Fila de alta prioridade:    [ 109 ]
Fila de baixa prioridade:  [ 103 | 104 ]
Fila de IO do tipo disk:   [ ]
Fila de IO do tipo printer: [ 102 ]
Fila de IO do tipo mag_tape: [ ]

-----

```

- Processo finalizando sua execução

```

-----
Tempo 59: O processo de pid = 109 esta em execucao

- Processo de pid = 109 terminou sua execucao
- Processo de pid = 108 saiu da fila de alta prioridade e foi colocado em execucao
Fila de alta prioridade:    [ ]
Fila de baixa prioridade:  [ 104 | 105 | 106 | 103 | 101 ]
Fila de IO do tipo disk:   [ ]
Fila de IO do tipo printer: [ ]
Fila de IO do tipo mag_tape: [ ]

-----

Tempo 60: O processo de pid = 108 esta em execucao

- Processo de pid = 108 sofreu uma interrupcao de IO do tipo disk e foi bloqueado
- Processo de pid = 104 saiu da fila de baixa prioridade e foi colocado em execucao
Fila de alta prioridade:    [ ]
Fila de baixa prioridade:  [ 105 | 106 | 103 | 101 ]
Fila de IO do tipo disk:   [ 108 ]
Fila de IO do tipo printer: [ ]
Fila de IO do tipo mag_tape: [ ]

-----

```

- Turnarounds

O tempo do final da execução de cada processo foi armazenado para que fosse calculado seu *turnaround*.

```
Turnarounds:
Processo 100: 25
Processo 101: 67
Processo 102: 18
Processo 103: 89
Processo 104: 67
Processo 105: 59
Processo 106: 52
Processo 107: 19
Processo 108: 39
Processo 109: 6
```

6. Bibliografia

- Stallings, William (2015). *Operating Systems: Internals and Design Principles*. [S.l.]: Pearson.
- <https://www.draw.io>. Ferramenta utilizada para construir o fluxograma.