



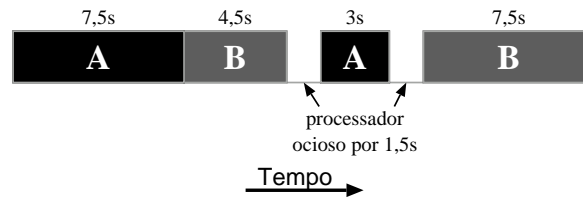
Curso de Tecnologia em Sistemas de Computação
Disciplina de Sistemas Operacionais
Professores: Valmir C. Barbosa e Felipe M. G. França
Assistente: Alexandre H. L. Porto

Quarto Período
Gabarito da AD1 - Primeiro Semestre de 2015

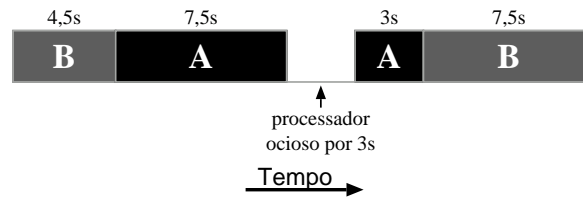
Atenção: Cada aluno é responsável por redigir suas próprias respostas. Provas iguais umas às outras terão suas notas diminuídas. As diminuições nas notas ocorrerão em proporção à similaridade entre as respostas. Exemplo: Três alunos que respondam identicamente a uma mesma questão terão, cada um, $1/3$ dos pontos daquela questão.

Nome -
Assinatura -

-
1. (1,5) Na figura a seguir mostramos como dois processos, A e B, são executados em um sistema operacional no qual a multiprogramação é usada somente para evitar a ociosidade do processador durante a execução de operações de E/S. O que ocorrerá com o tempo de ociosidade se agora B executar antes de A, supondo que o tempo da operação de E/S de A tenha sido reduzido pela metade e o tempo da operação de E/S de B tenha dobrado? E se a multiprogramação deixar de ser usada? Justifique a sua resposta.

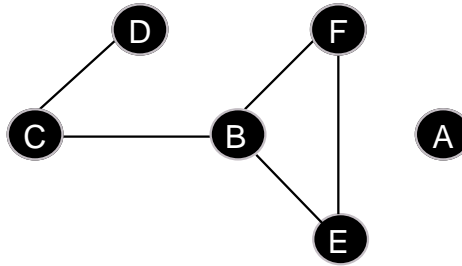


Resp.: -Como podemos ver pela figura, o tempo de execução do programa A no processador foi de 10,5s e o tempo da sua operação de E/S foi de 6s, e o tempo de execução do programa B no processador foi de 12s e o tempo da sua operação de E/S também foi de 6s. Na figura a seguir mostramos a ordem de execução com B executando antes de A, sendo que, segundo o enunciado, os novos tempos das operações de E/S de A e B são de, respectivamente, 3s e 12s. Como podemos ver pela figura, o tempo de ociosidade se mantém em 3s.

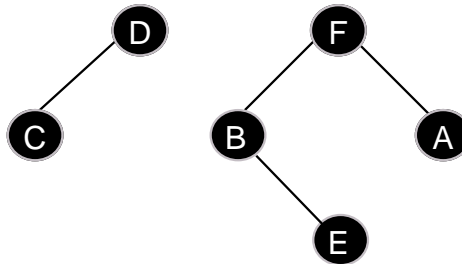


-Se a multiprogramação não for usada, o processador ficará ocioso enquanto as operações de E/S forem executadas, ou seja, durante as operações de E/S de A e B, independentemente de qual programa execute primeiro. Assim, o processador ficará ocioso por 12s.

2. (1,0) Um aluno de sistemas operacionais alegou que a hierarquia dada a seguir relaciona os processos A, B, C, D, E e F em execução no sistema operacional. A alegação do aluno está correta? Justifique a sua resposta.



Resp.: A alegação do aluno não está correta porque existem três erros na sua figura. O primeiro erro é que, estando no mesmo nível, B e C não podem estar relacionados um com outro. O segundo erro é que a aresta entre E e F não faz sentido, já que o pai de E tem que estar num nível acima dele. O terceiro erro é que o processo A não pode estar isolado no segundo nível. Uma possível alternativa à figura é a seguinte:



3. (1,5) Suponha que o sistema operacional esteja executando diretamente sobre o hardware de um computador cujas operações de E/S demorem 0,5ms. Suponha ainda que um processo tenha executado por 8,7s e que, durante a sua execução, tenha feito 7000 operações de E/S. Se o sistema operacional agora executar sobre uma máquina virtual que reduza a velocidade do processador em 35%, qual será o tempo de uma operação de E/S na máquina virtual se o programa fizer somente metade das operações de E/S e se executar agora em 10,8s? Justifique a sua resposta.

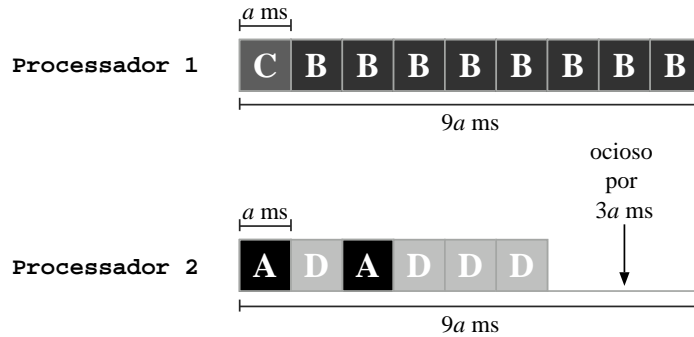
Resp.: Como cada operação de E/S demora 0,5ms e como o processo faz 7000 operações, então 3500ms do tempo de execução de 8,7s ou 8700ms desse processo são gastos em operações de E/S, quando ele executa no sistema operacional sobre o hardware do computador. Logo,

o processo executa no processador do hardware por $8\,700 - 3\,500 = 5\,200\text{ms}$. Note que a velocidade do processador ser reduzida em 35% significa que a velocidade do processador virtual é 65% da velocidade do processador real, o que por sua vez significa, durante os 5 200ms, que somente 65% das instruções serão executadas. Com isso, quando o processo executou no sistema operacional sobre a máquina virtual, o tempo de execução dele no processador virtual foi de $\frac{5\,200}{0,65} = 8\,000\text{ms}$. Agora, como o processo executou por 10,8s ou 10800ms na máquina virtual, isso significa que o tempo total de E/S na máquina virtual foi de $10800 - 8000 = 2800\text{ms}$. Finalmente, como o processo executou metade das operações de E/S, ou seja, 3500 operações, então o tempo de cada operação de E/S na máquina virtual é de $2\,800/3\,500 = 0,8\text{ms}$.

4. (1,5) Suponha que o escalonador substitua o processo em execução em um processador a cada a milissegundos, e que um processo somente possa executar novamente quando cada um dos outros processos prontos tiver sido executado também por a milissegundos em algum processador. Suponha que existam quatro processos em execução, A, B, C e D, e que eles precisem executar no processador por, respectivamente, $2a$, $8a$, a e $4a$ milissegundos. Qual será o tempo decorrido após a execução de todos os processos se somente 1 processador existir no hardware? E se existirem 2 processadores? E se existirem 4 processadores? Justifique a sua resposta.

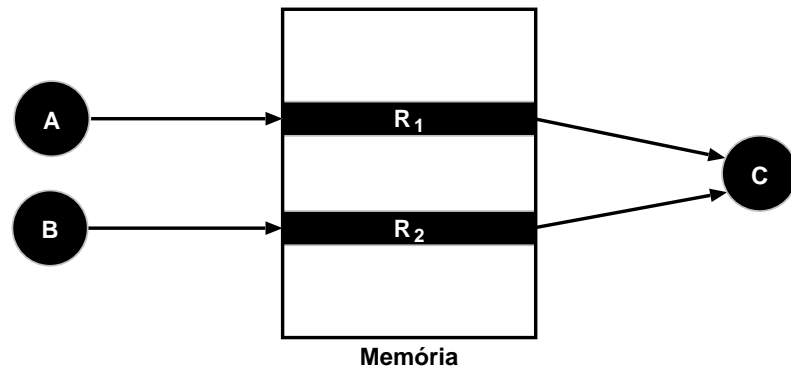
Resp.: -Se existir somente 1 processador então o tempo decorrido será igual ao tempo necessário para executar todos os quatro processos, e não dependerá do algoritmo de escalonamento usado. Logo, o tempo total de execução será $2a + 8a + a + 4a$ milissegundos, ou seja, $15a$ milissegundos.

-Agora, se existem 2 processadores, 2 processos poderão executar paralelamente, o que significa que um processo poderá voltar a executar em um processador depois de um outro processo ter executado em algum processador. Mais de uma solução, resultando no menor tempo total possível, pode existir; por exemplo, a ordem de execução dada na figura a seguir supondo que B e C sempre executam no Processador 1, e A e D sempre executam no Processador 2. Como podemos ver pela figura, o tempo total de execução de A, B, C e D será de $9a$ milissegundos.



-Finalmente, no caso de 4 processadores, podemos supor, sem perda de generalidade, que cada processo executa em um dos processadores até terminar e, com isso, o tempo total de execução de A, B, C e D será o maior tempo de execução desses processos no processador, ou seja, $8a$ milissegundos.

5. (2,0) Suponha que tenhamos duas regiões de memória, R_1 e R_2 , compartilhadas por três processos, A, B e C, como mostrado na figura dada a seguir. O processo A deposita um valor em R_1 , o qual sempre deverá ser consumido pelo processo C. O processo B, por sua vez, deposita um valor em R_2 , o qual também sempre deverá ser consumido pelo processo C. Finalmente, o processo C somente lê os valores de R_1 e R_2 e imprime o seu produto. Como 4 semáforos binários podem ser usados para garantir o correto funcionamento dos processos A, B e C, supondo que eles repitam o procedimento acima indefinidamente, e que as regiões R_1 e R_2 estejam inicialmente vazias? Justifique a sua resposta.



Resp.: Para a posição R_1 , precisamos do semáforo binário $Livre_1$ para bloquear A caso C ainda não tenha lido o valor de R_1 anteriormente colocado por A, e do semáforo binário $Disponivel_1$ para bloquear C caso A ainda não tenha colocado um novo valor em R_1 após C ler o último valor de R_1 . Similarmente, para a posição R_2 , precisamos do semáforo binário $Livre_2$ para bloquear B caso o processo C ainda não tenha lido o valor de R_2 anteriormente colocado por B, e do semáforo binário $Disponivel_2$ para bloquear C caso B ainda não tenha colocado um novo valor em R_2 após C ler o último valor de R_2 . Como inicialmente as posições R_1 e R_2 estão vazias então os valores dos semáforos $Livre_1$, $Livre_2$, $Disponivel_1$ e $Disponivel_2$ são de, respectivamente, 1, 1, 0 e 0. A seguir mostramos os códigos dos processos A, B e C que garantem o correto funcionamento dos processos:

```

void ProcessoA()
{
    while (1)
    {
        P(Livre1);
        // Código para copiar um valor para R1.
        V(Disponivel1);
    }
}

```

```

void ProcessoB()
{
    while (1)
    {
        P(Livre2);
        // Código para copiar um valor para R2.
        V(Disponivel2);
    }
}

void ProcessoC()
{
    while (1)
    {
        P(Disponivel1);
        P(Disponivel2);
        // Código para copiar o valores de R1 e R2 e imprimir o seu produto.
        V(Livre1);
        V(Livre2);
    }
}

```

6. (2,5) Cinco processos, A, B, C, D e E, foram inicializados e têm tempos de execução de, respectivamente, 15, 9, 3, 12 e 8 segundos. Suas prioridades são, respectivamente, 5, 3, 1, 7 e 2, sendo que o processo mais prioritário tem a maior prioridade. Para cada um dos seguintes algoritmos de escalonamento, determine a média dos tempos de término dos processos. Ignore o acréscimo (*overhead*) da comutação de processos e suponha que nenhum processo faça operações de E/S. Justifique a sua resposta.
- (a) (1,0) *Round robin* com um quantum de 2 segundos de duração, e com os processos inicialmente executando na ordem B, E, D, C e A.

Resp.: Pelo enunciado, vemos que a ordem de execução dos processos é como a dada nas tabelas as seguir. Nestas tabelas mostramos como os processos são escolhidos pelo algoritmo, sendo que cada coluna refere-se à execução de um processo dando o tempo

de início de cada quantum e o processo correspondente. Note que os processos A, B e C somente precisam executar pela metade, 1s, do último quantum porque os seus tempos de execução são números ímpares. Pelas tabelas, vemos que os tempos de término dos processos A, B, C, D e E, são de, respectivamente, 47, 37, 17, 42 e 31 segundos, o que nos dá um tempo médio de 34,8 segundos.

0	2	4	6	8	10	12	14	16	17	19	21	23
B	E	D	C	A	B	E	D	C	A	B	E	D

25	27	29	31	33	35	37	38	40	42	44	46
A	B	E	D	A	B	D	A	D	A	A	A

- (b) (1,0) Escalonamento por prioridades, supondo que a prioridade do processo em execução seja reduzida por 1 unidade a cada segundo, e que o processo deixe de executar somente quando um outro processo passa a ter a maior prioridade.

Resp.: Como existem várias possíveis ordens de execução devido a empates nas prioridades, vemos nas tabelas a seguir uma possível ordem de execução dos processos. Estas tabelas são similares a do item anterior e possuem somente mais uma linha, a terceira, que mostra a prioridade de cada processo antes de ele executar na unidade de tempo dada na mesma coluna desse processo. Pelas tabelas, vemos que os tempos de término dos processos A, B, C, D e E, são de, respectivamente, 47, 42, 27, 37 e 41 segundos, o que nos dá um tempo médio de 38,8 segundos.

0	1	2	3	4	5	6	7	8	9	10	11
D	D	D	A	A	D	D	B	A	A	E	D
7	6	5	5	4	4	3	3	3	2	2	2

12	13	14	15	16	17	18	19	20	21	22	23
B	B	C	A	E	D	D	B	C	A	E	E
2	1	1	1	1	1	0	0	0	0	0	-1

24	25	26	27	28	29	30	31	32	33	34	35
D	B	C	A	A	E	D	B	B	A	E	D
-1	-1	-1	-1	-2	-2	-2	-2	-3	-3	-3	-3

36	37	38	39	40	41	42	43	44	45	46
D	B	A	E	E	B	A	A	A	A	A
-4	-4	-4	-4	-5	-5	-5	-6	-7	-8	-9

(c) (0,5) Trabalho mais curto primeiro.

Resp.: Neste caso, a ordem de execução dos trabalhos é a seguinte: C, E, B, D e A. Como cada processo executa exclusivamente no processador até o seu término, C executa por 3 segundos, seguido pelos processos E, B, D e A, que executam por, respectivamente, 8, 9, 12 e 15 segundos. Logo, os tempos de término dos processos de A a E são, respectivamente, de 47, 20, 3, 32 e 11, o que nos dá um tempo médio de 22,6 segundos.