

Relatório

EP1 - 2017
ACH2044 - Sistemas Operacionais

Integrantes:

Filipe Guimarães de Rosset Antônio	9844901
Leonardo Colman Lopes	9875490
Rafael Galvão Silveira	3495012

Resultados

Sabendo que os processos poderiam ter, no máximo, 21 comandos incluindo a SAIDA, e que era necessário criar testes distribuídos uniformemente num intervalo útil, optamos pelos seguintes quanta: 1, 3, 5, 7, 9, 11, 13, 15, 17, 19, 21.

Ao rodar o escalonador para cada um dos quanta acima, obtivemos 11 logs de saída, conforme tabela abaixo:

Quantum	Med_Trocas	Med_Instruções
1	16,2	0,7963
3	6,3	2,0476
5	4,6	2,8043
7	3,9	3,3077
9	3,8	3,3947
11	3,6	3,5833
13	3,5	3,6857
15	3,4	3,7941
17	3,4	3,7941
19	3,4	3,7941
21	3,3	3,9091

Para uma análise mais eficiente dos dados, fizemos 3 gráficos, mostrados abaixo:

Gráfico 1- Média de Trocas x Quantum

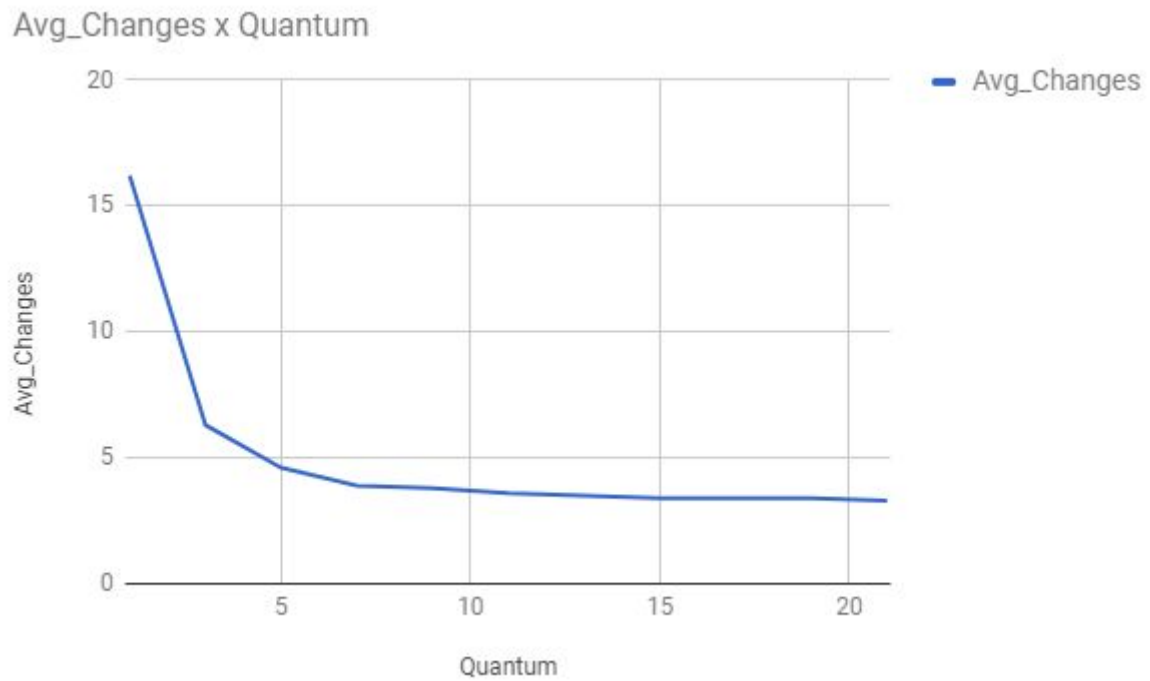


Gráfico 2- Média de Instruções x Quantum

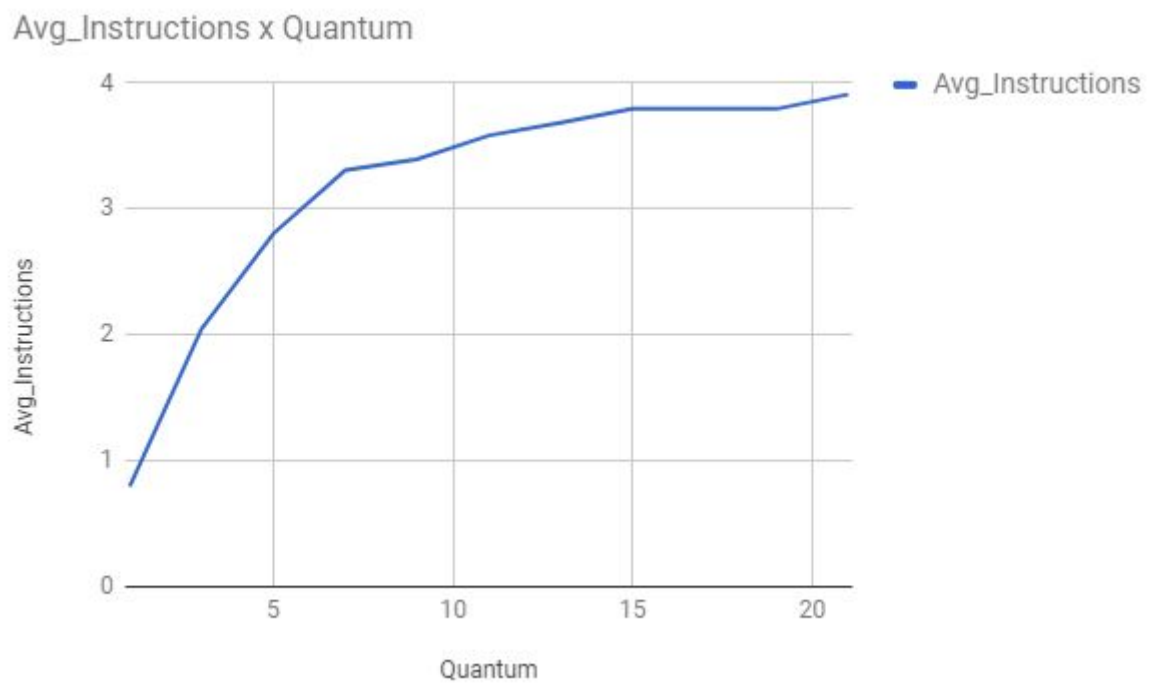
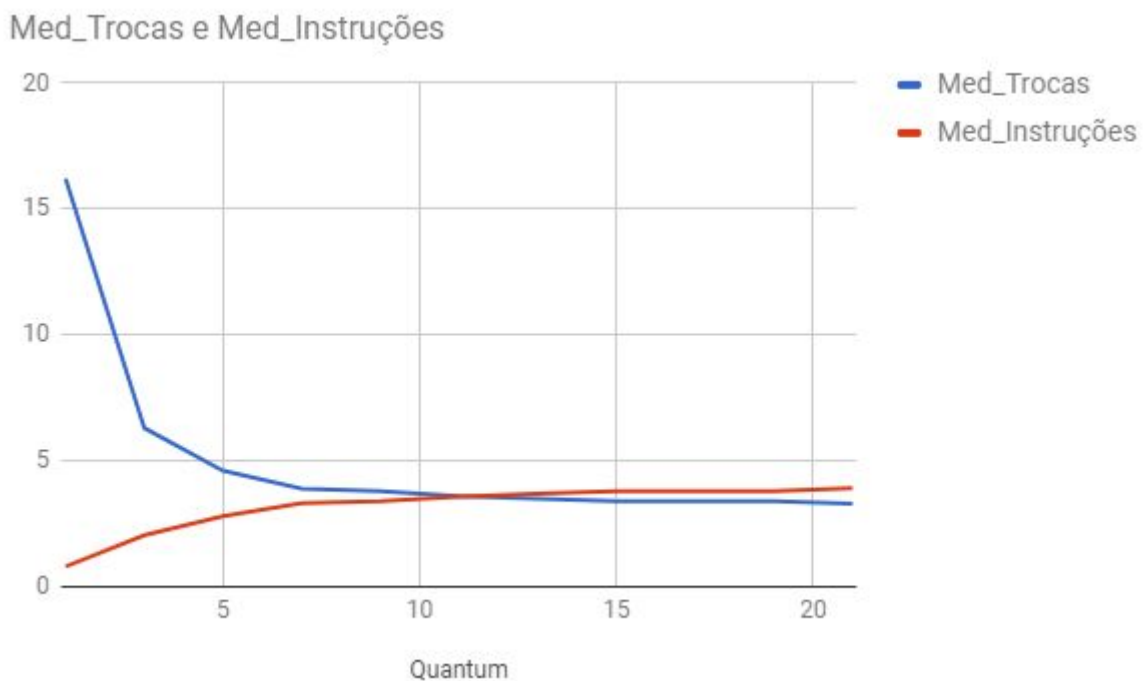


Gráfico 3- Média de Trocas e Média de Instruções x Quantum



Análise

Como podemos verificar, no Gráfico 1 obtivemos uma curva semelhante à logarítmica descendente. Ou seja, um pequeno aumento no número de quanta é capaz de diminuir muito o número de trocas médio, tornando-as quase constantes após um dado valor. No nosso gráfico, percebemos que acima de 5 quanta o número de trocas médio não diminui significativamente.

Além disso, no Gráfico 2 também obtivemos uma curva semelhante à logarítmica, só que ascendente. Nesse caso, percebemos que um pequeno aumento no número de quanta é capaz de aumentar a média de instruções executadas, e que esse número se torna quase constante após 7 quanta (no nosso caso).

Por fim, no último gráfico temos a combinação dos 2 anteriores. Através dele podemos notar que as curvas de Med_Trocas e Med_Instruções têm uma tendência a permanecer estáveis, mesmo dobrando ou triplicando o número de quanta.

Concluimos que, de acordo com o problema proposto, é necessário haver um equilíbrio de recursos, uma vez que não é muito eficiente manter uma alta quantidade de quanta em um processo se o objetivo for multiprogramação (pode demorar muito para executar outros processos). Simplesmente aumentar o quantum de um processo não é suficiente para deixá-lo mais eficiente, visto que sua taxa de instruções/quantum não aumenta significativamente depois de um certo número. Também não adianta deixar o quantum muito pequeno para aumentar a rotatividade do processo, pois serão executadas poucas instruções e muitas trocas, tornando o sistema ineficiente da mesma forma.