

# Aula Prática 10

## 1 - Planejar um conjunto de experimentos variando os parâmetros.

O objetivo dos nossos experimentos será encontrar um balanço ideal entre a ordem da árvore B ( $m$ ) e o tamanho dos registros (CARGA) em bytes.

Em teoria, quanto maior for a ordem da árvore, mais será valorizada a localidade de referência espacial. Dessa maneira, ao acessar uma página, um número de registros maior será feito, deixando a árvore mais achatada e provendo acesso mais rápido às suas folhas.

Inicialmente, será montado um protocolo no programa usando o struct exp. Com ele criaremos objetos que testarão:

- Números de operações de 0 até 100
- Chaves máximas de 0 até 100

A partir disso, iremos alterar os parâmetros  $m$  e CARGA do código com:

- $m$ : de 2 até 512.
- CARGA: de 10 até 10,000.

Com isso, buscaremos encontrar o compromisso entre  $m$  e CARGA.

## 2 - Justificar porque o plano de experimentos exercita o compromisso entre os parâmetros.

O plano de experimentos proposto busca encontrar um compromisso ideal entre a ordem da árvore B ( $m$ ) e o tamanho dos registros (CARGA). Através da variação de  $m$  de 2 até 512 e CARGA de 10 até 10,000, serão realizados experimentos com diferentes cenários, operações e chaves máximas. Isso permitirá avaliar como as alterações nesses parâmetros afetam a localidade de referência espacial e temporal, buscando otimizar o desempenho da árvore B.

Ao exercitar esse compromisso, será possível observar como um maior valor de  $m$  influencia a localidade de referência espacial, reduzindo o número de acessos a disco. Por outro lado, ao variar o tamanho dos registros, será possível analisar como isso afeta o uso de memória e a localidade de referência temporal. Com essa abordagem abrangente, será possível encontrar o equilíbrio ideal entre  $m$  e CARGA, visando otimizar a eficiência e o desempenho geral da árvore B em diferentes situações.

### 3 - Executar os experimentos planejados, medindo o tempo de execução e outros parâmetros que considerar pertinentes.

Para isso, adicionamos um loop na main do programa que testa nossos parâmetros sobre o número de Operações e chave máxima. A partir disso, medimos o tempo total gasto pelo programa para cada valor de m e CARGA que desejamos testar.

m	CARGA	Tempo(ms)
2	10	64420
64	1250	234599
128	2500	374635
192	3750	570423
256	5000	Killed
320	6250	SegFault
384	7500	SegFault
448	8750	SegFault
512	10000	SegFault

Apesar do programa não funcionar mais com valores tão grandes, pelos primeiros resultados já é possível avaliar que o aumento fornecido ao m não é suficiente para suprir o custo resultante do aumento da CARGA nessa escala.

Então, foram alterados os parâmetros:

m	CARGA	Tempo(ms)
2	10	63894
64	125	361555
128	250	147126
192	375	167353
256	500	218021
320	625	221210

384	750	233198
448	875	247437
512	1000	305657

Dessa vez, o tempo gasto aparentou crescer de forma razoavelmente linear! Podemos definir essa escala como uma boa regra. Vamos formaliza-la:

Para cada aumento de ordem em 64, devemos aumentar a CARGA em 125. Ou seja:

**A razão  $m/CARGA$  deve ser de  $64/125$  ou  $1/2$  aproximadamente.**