

Guião Laboratório 2: Avaliação Experimental

É comum quando se tem mais do que um algoritmo para resolver um determinado problema que se pretenda saber qual é o algoritmo mais eficiente. Normalmente é realizada uma análise da complexidade assintótica do ponto de vista teórico, mas também é realizada uma avaliação experimental que nos poderá ajudar a escolher o algoritmo.

Nesta aula iremos realizar a avaliação experimental dos algoritmos *Fibonnaci* e *MaximalSubarray*. Iremos começar pelo algoritmo *MaximalSubarray*, nomeadamente a sua versão quadrática e a sua versão linear.

O Problema do Subarray Maximal

O problema do *subarray* maximal consiste na tarefa de, existindo um *array* de números, obter um *subarray* contíguo em que a soma dos seus valores é a maior possível.

Ambiente de Teste:

Sempre que se realiza uma avaliação experimental, é necessário indicar no relatório as características do ambiente de teste, isto é características do computador, nomeadamente do processador e da memória.

Relativamente às variáveis de teste a considerar numa determinada avaliação experimental, estas estão dependentes do algoritmo a testar. Por exemplo, como no problema do *subarray* maximal temos como parâmetro um *array*, é necessário pensar se a distribuição dos números poderá influenciar o algoritmo e escolher um conjunto de distribuições de números adequado para aferir o comportamento do algoritmo. Além disso, é necessário pensar para que dimensões de *array* será testado o algoritmo. Neste guião serão fixadas estas variáveis de teste, mas em cada avaliação experimental realizada terá de ser feito um levantamento destas variáveis, que são dependentes do problema em questão e descrevê-las no relatório.

As variáveis que dependem do problema do *subarray* maximal e consideradas neste guião são:

- 1- Distribuição dos valores;
 - 2- Domínio dos valores;
 - 3- Dimensões do *array*.
-
1. Distribuições aleatórias dos valores (pode utilizar o método **Math.random()**):
 - 1.1. Completamente aleatória;
 2. Domínio dos valores:
 - 2.1 Inteiros positivos e negativos, distintos;
 - 2.2 Inteiros positivos e negativos, com elevado número de repetidos (50%).
 3. Dimensões a testar do array:
 - 3.1 $100^2 \rightarrow 10000$
 - 3.2 $200^2 \rightarrow 40000$
 - 3.3 $300^2 \rightarrow 90000$
 - 3.4 $400^2 \rightarrow 120000$

Nota importante: Observem que as dimensões consideradas estão a ser colocadas ao quadrado. Deste modo, se o algoritmo for quadrático, o crescimento com esta escala será linear.

Medição dos tempos de execução: o tempo de cada caso de teste para cada versão do algoritmo (quadrática e linear) deve corresponder à média/mediana de várias execuções (por exemplo 11). Para medir os tempos de execução poderá utilizar o **java.lang.System.currentTimeMillis()**.

Medição do espaço em memória: existem várias ferramentas disponíveis, como por exemplo o **JProfiler**. No entanto, não será incluído este estudo para esta aula.

Resultados:

Para apresentar os resultados dos testes, deverá apresentar tabela(s) comparativa(s) e gráfico(s).

Neste caso a tabela deverá ser semelhante à seguinte tabela, onde deverão ser colocadas as médias/medianas das 11 execuções realizadas para cada caso (por exemplo 11, mas poderá ser apenas 5 para esta aula).

		dados aleatórios	
		sem repetições	com repetições
Algoritmo 1	10000		
	40000		
	90000		
	120000		
Algoritmo 2	10000		
	40000		
	90000		
	120000		

Deverá ser realizado um gráfico por cada coluna (2ª, 3ª). Cada gráfico deverá ter no eixo dos y o valor do tempo (em ms) e no eixo dos x a dimensão do *array*. Em cada gráfico deverá ter duas linhas correspondentes ao tempo de execução de cada algoritmo, indicando na legenda qual é o algoritmo que corresponde a cada linha.

A discussão dos resultados também engloba uma análise comparativa entre os resultados obtidos através da avaliação experimental e a complexidade assintótica de cada algoritmo.

Nº-ésimo número de Fibonnaci

O ambiente de teste para a avaliação experimental dos algoritmos para obtenção do nº-ésimo número de Fibonnaci apresentados nas aulas é mais simples por existir apenas um único dado de entrada. Sugere-se que faça também a avaliação experimental, mas adaptando a este problema.