

Algoritmos e Estruturas de Dados

Erros Gerais Mais Frequentes nos Trabalhos de AED

DI – FCT/UNL
18 de Outubro de 2010

1 Introdução

Este documento apresenta alguns erros gerais encontrados com frequência nos trabalhos de Algoritmos e Estruturas de Dados. Para além de poderem existir outros erros gerais, que aqui não são referidos, há sempre erros específicos de cada trabalho concreto. Este texto não identifica os erros específicos do trabalho de 2010/11.

A nota do trabalho depende da versão implementada (I ou J) e de 3 números:

- o número total de erros de gravidade muito elevada (chamados **erros graves**);
- a soma dos pesos de todos os erros de gravidade média (chamados **erros médios**);
- a soma dos pesos das **bonificações**.

A Secção 2 descreve alguns erros graves, a Secção 3 identifica alguns erros médios mais comuns e especifica os seus pesos, e a Secção 4 contém a tabela que permite calcular a nota do trabalho.

2 Erros Graves

A lista que se segue indica alguns erros gerais graves mais comuns.

- As únicas EDs são vectores ou listas ligadas.
- Pesquisas, inserções, remoções ou percursos em listas ligadas com complexidade superior a linear.
- Pesquisas, inserções ou remoções em tabelas de dispersão com complexidade superior a constante, no caso esperado.
- Percursos em tabelas de dispersão com complexidade superior a linear.
- Pesquisas, inserções ou remoções em árvores binárias de pesquisa com complexidade superior à altura da árvore.
- Percursos em árvores binárias de pesquisa com complexidade superior a linear.
- Cálculo da dimensão de uma ED com complexidade superior a constante.

3 Erros Médios

A lista que se segue indica alguns erros gerais médios mais comuns. Regra geral, cada erro tem peso 1, mas há casos (assinalados) em que o peso pode ser diferente.

Consoante o contexto, o nome *Database* denota o TAD principal da aplicação, a respectiva interface ou a classe que a implementa. É um nome genérico, escolhido porque se aplica a todos os trabalhos de AED, mas que não deve ser usado devido, precisamente, à sua pouca expressividade.

- [Peso de 1 a 6] Relatório com partes omissas, incompletas ou com erros.
- Interfaces ou classes da aplicação no pacote *dataStructures*.
- Interfaces ou classes da aplicação no mesmo ficheiro.
- Ausência da interface *Database*.
- Ausência da classe *Database*.
- Funcionalidade da *Database* diferente da especificada no enunciado.
- Leitura de dados de entrada de um comando após a chamada do método correspondente da *Database*.
- Escrita de resultados de um comando antes do fim da execução do método correspondente da *Database*.
- A *Database* ou alguma classe usada (directa ou indirectamente) pela *Database* conhece as *Strings* a escrever na saída padrão (por exemplo, passadas em mensagens de excepções) ou a formatação especificada no enunciado.
- Ausência de interface para os objectos retornados pela *Database*.
- Objectos retornados pela *Database* alteráveis do exterior.
- [Peso de 1 a 2] Nas classes da aplicação, a funcionalidade está repartida apenas entre a *Main* e a *Database*, ou seja, todas as outras classes têm pouca funcionalidade, não havendo descentralização de tarefas.
- Uso de interfaces ou de classes com tipos genéricos ignorando os tipos genéricos.
- Uso inadequado do mecanismo de herança.
- Repetição de atributos ou de métodos por falta de herança.
- [Peso de 1 a 2] Violação da privacidade das classes (i.e., acesso exterior aos atributos de uma classe não herdada).
- Violação de invariantes das classes.
- [Peso 2] Uso de EDs com invariantes assegurados por classes da aplicação.
- [Peso 2] Classes de EDs adulteradas para contemplar detalhes da aplicação.
- Uso das classes (em vez das interfaces) na definição dos tipos de variáveis e de parâmetros.
- [Peso 1 por Método] Métodos das interfaces não (ou mal) implementados (e.g., iterador de uma TD).

- Métodos auxiliares públicos.
- TD estática (ausência de redispersão ou má implementação).
- TDs criadas com a capacidade por omissão.
- Existência de informação duplicada (excessiva).
- Operações (pesquisas, inserções, remoções, etc.) em EDs com mais passos do que os necessários (mas sem aumento da complexidade).
- [**Peso de 1 a 2**] Criação de objectos desnecessária na inicialização de variáveis.
- [**Peso de 1 a 2**] Repetição de computações (por exemplo, por falta de variáveis auxiliares).
- [**Peso de 1 a 2**] Ausência de métodos auxiliares: código repetido ou métodos de “várias páginas”.
- Código muito confuso.
- Identificadores (de interfaces, classes, métodos, variáveis, parâmetros) incompreensíveis ou pouco expressivos.
- Inexistência de constantes ou não utilização das existentes.
- Ausência de comentários (que fazem mesmo falta).

4 Cálculo da Nota do Trabalho

A nota depende:

- da versão (primeira coluna);
- do número total de erros graves (segunda coluna); e
- do chão da diferença entre a soma dos pesos dos erros médios e a soma dos pesos das bonificações (terceira coluna).

Por exemplo, um trabalho da versão I, com 0 erros graves, 5, 5 no valor dos erros médios e 1 no valor das bonificações é avaliado com 17.

Versão	Erros Graves	[Erros Médios - Bonif]	Nota do Trabalho
I	0	≤ 0	20
		1	19
		2	18
		3, 4	17
		5, 6	16
		7, 8	15
		9, 10	14
		11, 12	13
		≥ 13	12
	1	0, ..., 4	13
		5, ..., 8	12
		9, ..., 12	11
		≥ 13	10
	2	0, ..., 4	11
		≥ 5	10
≥ 3	≥ 0	10	
J	0	≤ 0	15
		1, ..., 4	14
		5, ..., 8	13
		9, ..., 12	12
		≥ 13	11
	1	0, ..., 4	12
		5, ..., 8	11
		≥ 9	10
	2	0, ..., 4	11
		≥ 5	10
	≥ 3	≥ 0	10