

Métodos Formais em Engenharia de Software

MIEIC, 4º ano, 1º semestre

Exame – 24 de janeiro de 2012 – Sem consulta – Duração 10 minutos (Alloy)

Nº:	Nome:
-----	-------

1. [8 valores] Para cada pergunta de escolha múltipla, escolha a resposta correcta. Cada resposta certa vale 2 pontos e cada resposta errada desconta 0.5 valores.

a) Considere as relações $A=\{2 \rightarrow 2, 2 \rightarrow 3\}$ e $B=\{1 \rightarrow 3, 3 \rightarrow 2\}$. Qual das seguintes afirmações é correcta?

- $\# \wedge (B[3] \prec A) = 3$
- $(\sim A) \cdot \{(2)\} = (\{(1), (3)\} \prec B) \cdot \{(1), (3)\}$
- $A \cdot B \neq B \cdot \sim A$
- Todas as alíneas anteriores estão correctas
- Nenhuma alínea está correcta

b) Considere as relações $A=\{3 \rightarrow 2 \rightarrow 3, 2 \rightarrow 2 \rightarrow 3\}$ e $B=\{3, 2\}$. Qual das seguintes afirmações é correcta?

- $A \cdot B = \{(2)\} \rightarrow B$
- $A \cdot B = B \rightarrow B = B \cdot A$
- $\{(2)\} \cdot (\sim (A \cdot B)) = B$
- Todas as alíneas anteriores estão correctas
- Nenhuma alínea está correcta

c) Qual das seguintes relações descreve uma relação injetiva?

- $r: A \rightarrow \text{some } B$
- $r: A \text{ one } \rightarrow \text{one } B$
- $r: A \text{ some } \rightarrow B$
- As duas alíneas anteriores estão correctas
- Nenhuma alínea está correcta

d) Qual das seguintes relações torna a expressão " $r: A \rightarrow B \mid r[A] \cdot \sim r = A$ " verdadeira?

- $(\#r \cdot B = \#A \cdot r)$ and (no disj x, y in $A \mid x \cdot r = y \cdot r$)
- $(A \cdot r = B)$ and (no disj x, y in $B \mid r \cdot x = r \cdot y$)
- $(r \cdot B = A)$ and (no disj x, y in $A \mid x \cdot r = r[y]$)
- Duas das alíneas anteriores estão correctas
- As três primeiras alíneas estão correctas
- Nenhuma alínea está correcta

Métodos Formais em Engenharia de Software

MIEIC, 4º ano, 1º semestre

Exame – 24 de janeiro de 2012 – Com consulta – Duração 50 minutos (Alloy)

Nº:	Nome:
-----	-------

2. [12 valores] Considere a seguinte representação de uma árvore rubro-negra em Alloy. Uma árvore rubro-negra é um tipo árvore binária balanceada, usada em ciência da computação para organizar dados que possam ser comparáveis. Neste tipo de árvores, os nós das folhas (nós com **NIL** na figura 1) não são relevantes e não contêm dados (**assuma** que **NIL** é representado pelo valor **0** em **id**).

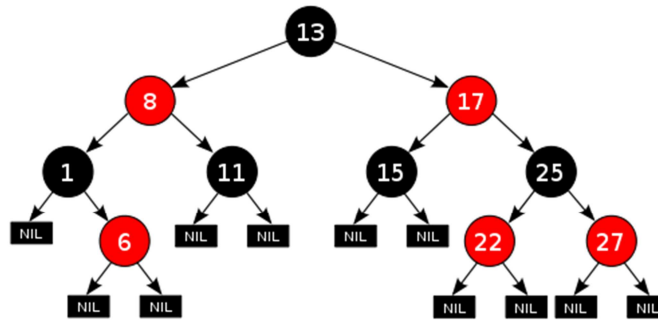


Figura 1: exemplo de árvore rubro-negra

```

abstract sig color {}

sig RED, BLACK extends color {}

sig edge {
  cor : one color
  id : Int
  left : lone edge
  right : lone edge
}

one sig root extends edge{}

-- fact1: The root is black
-- fact2: All leaves are the same color as the root
-- fact3: Both children of every red node are black
    
```

- 2.1. Formalize os factos *fact1* (a raiz é um nó preto) e *fact2* (todas as folhas têm a mesma cor da raiz).

- 2.2. Escreva uma operação que retorne o elemento (*edge*) com *id* máximo de uma árvore. Por exemplo, na figura, o *edge* com *id* máximo tem o valor 27.

-
- 2.3. Escreva uma operação, *sameBlackNodes*, que verifique se todos os caminhos a partir da raiz até às folhas contém o mesmo número de nós pretos. A operação retorna *true* (se se verificar a condição) ou *false*. Assuma que existe uma função *blacks*[*n1*: *edge*, *n2*: *edge*] que retorna o número de nós pretos entre dois nós.

- 2.4. Formalize o *fact3* (os filhos de nós vermelhos são pretos).

Boa sorte!