

Semântica das Linguagens de Programação

1º Teste (20 de Abril de 2016) – Duração: 90 min

Questão 1 Sejam $C, P \in \mathbf{Stm}$ os seguintes programas:

$$\begin{array}{l|l} C \equiv \begin{array}{l} \text{if } (y > 0) \text{ then } z := 2 \\ \text{else if } (x > 0) \text{ then } z := 1 \\ \text{else } z := 0 \end{array} & P \equiv \{y := 7; x := y + 2\}; C \end{array}$$

1. Recorrendo à *semântica de transição (small-step)* simule a execução do programa P a partir do estado inicial s onde todas as variáveis têm o valor 0. Para cada transição apresente a árvore de prova que a justifica.
2. Diga, justificando, se o programa P é semanticamente equivalente ao seguinte programa:
 $z := 2; y := 7; x := z + y.$
3. Queremos agora estender a linguagem **While** com uma estrutura de controlo condicional com 3 ramos. Um exemplo da sua utilização (atente à sintaxe utilizada) poderá ser:
$$\begin{array}{l} \text{if } (y > 0) \text{ then } z := 2 \\ \text{elsif } (x > 0) \text{ then } z := 1 \\ \text{else } z := 0 \end{array}$$
 - (a) Apresente as regras da *semântica natural (big-step)* para este novo comando.
 - (b) Apresente uma *interpretação denotacional* para este novo comando.
 - (c) Proponha uma regra da lógica de Hoare para este novo comando, e prove a sua correcção face a uma das semânticas que definiu nas alíneas anteriores (escolha a que quiser).

Questão 2 Pretende-se estender a linguagem **While** acrescentando-lhe um novo comando **assert**(b) com a seguinte semântica: “Se o valor da expressão booleana b for verdadeiro, o comando comporta-se como o *skip*. Caso contrário, o programa bloqueia.”

1. Estenda as regras da *semântica de transições (small-step)* para lidar com este novo comando.
2. Será o comando $\{\text{assert}(\text{false}); S\}$ semanticamente equivalente a $\{\text{while true do skip}\}$? Justifique a sua resposta.
3. Relativamente à máquina abstracta **AM**, o que deverá fazer para poder lidar com esta versão da linguagem **While** estendida com o comando **assert**(b) ?

Questão 3 Sejam $b_1, b_2 \in \mathbf{Bexp}$ tal que $b_1 \Rightarrow b_2$, e $A, B, P_1, P_2 \in \mathbf{Stm}$ tal que

$$\begin{array}{lcl} P_1 & \equiv & \text{while } b_1 \text{ do } \{\text{if } b_2 \text{ then } A \text{ else } B\} \\ P_2 & \equiv & \text{while } b_1 \text{ do } A \end{array}$$

Com base na semântica natural (big-step), indique o significado formal da afirmação:

“ P_1 e P_2 são programas semanticamente equivalentes”

e demonstre uma das implicações (à sua escolha) envolvidas na prova dessa equivalência.