

Algoritmos e Complexidade

LEI/LCC (2º ano)

2ª Ficha Prática

Ano Lectivo de 2009/10

O objectivo desta ficha é treinar o aluno na utilização das regras da lógica de Hoare que não envolvem ciclos, bem como na escrita de invariantes.

Primeiras provas de correcção com Lógica de Hoare

1. Apresente uma prova que justifique cada um dos seguintes triplos de Hoare:

- (a) $\{I > J\} J := I + 1; I := J + 1 \{I > J\}$
- (b) $\{I \neq J\} \text{IF } I > J \text{ THEN } M := I - J \text{ ELSE } M := J - I \{M > 0\}$
- (c) $\{A > B\} M := 1; N := A - B \{M * N > 0\}$
- (d) $\{S = 2^I\} I := I + 1; S := S * 2 \{S = 2^I\}$
- (e) $\{\text{True}\} \text{IF } I < J \text{ THEN } \text{MIN} := I \text{ ELSE } \text{MIN} := J \{\text{MIN} \leq I \wedge \text{MIN} \leq J\}$
- (f) $\{I > 0 \wedge J > 0\} \text{IF } I < J \text{ THEN } \text{MIN} := I \text{ ELSE } \text{MIN} := J \{\text{MIN} > 0\}$

Introdução aos invariantes de ciclo

1. Encontre um invariante para o ciclo no programa seguinte e prove a sua preservação. O invariante deverá implicar que, no caso de terminação do ciclo, a pós-condição $S = 2^I$ é satisfeita.

```
WHILE I < N DO
  BEGIN I:=I+1; S:=S*2; END
```

2. Encontre um invariante para o ciclo no programa seguinte e prove a sua preservação. O invariante deverá implicar que, no caso de terminação do ciclo, a pós-condição $R < Y \wedge X = R + (Y * Q)$ é satisfeita.

```
R:=X;
Q:=0;
WHILE Y <= R DO
  BEGIN R:=R-Y; Q:=Q+1; END
```

3. Considere o seguinte algoritmo de multiplicação de dois números inteiros.

```
RES := 0;
WHILE (Y>0) DO
  BEGIN
    RES := RES + X;
    Y = Y-1
  END
```

- (a) Escreva predicados que descrevam a pré e pós condição deste algoritmo. Procure que a pré-condição garanta a terminação do algoritmo.
 - (b) Apresente um invariante de ciclo que, no caso da terminação do ciclo, implique a pós-condição que identificou na alínea anterior.
 - (c) Prove a preservação do invariante de ciclo.
4. Repita as duas últimas alíneas do exercício anterior agora para o seguinte algoritmo que toma partido de que a divisão e multiplicação por 2 são operações muito eficientes (trata-se, em representação binária, de *shifts*).

```

RES := 0;
WHILE (Y>0) DO
BEGIN
  IF (Y % 2 != 0) THEN BEGIN Y := Y - 1; RES := RES + X; END
  X := X*2;
  Y := Y/2;
END

```

5. Considere o seguinte programa, em que L1 e L2 são variáveis booleanas, e n1 e n2 são variáveis inteiras.

```

L1 := false; L2 := false;
WHILE (N1 + N2 > 0) DO
BEGIN
  IF (L2 = false) THEN
  BEGIN
    IF (N1 > 0)
    THEN BEGIN L1 := true; N1 := N1 - 1; END
    ELSE L1 := false;
  END
  IF (L1 = false) THEN
  BEGIN
    IF (n2 > 0)
    THEN BEGIN L2 := true; N2 := N2 - 1; END
    ELSE L2 := false;
  END
  L2 := ! L2;
  L1 := ! L1;
END

```

Encontre um invariante para o ciclo WHILE e prove a sua preservação. O invariante deverá implicar que, no caso de terminação do ciclo, a pós-condição $L1 = \text{false} \vee L2 = \text{false}$ é satisfeita.

Algumas Regras em Lógica de Hoare

Fortalecimento da pré-condição

$$\frac{R \Rightarrow P \quad \{P\} S \{Q\}}{\{R\} S \{Q\}} \text{ (Fort)}$$

Enfraquecimento da pós-condição

$$\frac{\{P\} S \{Q\} \quad Q \Rightarrow R}{\{P\} S \{R\}} \text{ (Enfracq)}$$

Atribuição-1

$$\frac{}{\{P[x \setminus E]\} x = E \{P\}} \text{ (Atrib1)}$$

Atribuição-2

$$\frac{P \Rightarrow (Q[x \setminus E])}{\{P\} x = E \{Q\}} \text{ (Atrib2)}$$

Sequência

$$\frac{\{P\} S_1 \{R\} \quad \{R\} S_2 \{Q\}}{\{P\} S_1; S_2 \{Q\}} \text{ (;)}$$

Condiciona

$$\frac{\{P \wedge c\} S_1 \{Q\} \quad \{P \wedge \neg c\} S_2 \{Q\}}{\{P\} \text{ if } c S_1 \text{ else } S_2 \{Q\}} \text{ (if)}$$

Ciclo-1

$$\frac{\{I \wedge c\} S \{I\}}{\{I\} \text{ while } c S \{I \wedge \neg c\}} \text{ (while-1)}$$

Ciclo-2

$$\frac{P \Rightarrow I \quad \{I \wedge c\} S \{I\} \quad (I \wedge \neg c) \Rightarrow Q}{\{P\} \text{ while } c S \{Q\}} \text{ (while-2)}$$