GX1309

OLLSCOIL NA hÉIREANN GAILLIMH

NATIONAL UNIVERSITY OF IRELAND GALWAY

SEMESTER II
SPRING EXAMINATIONS 2000

Third University Examination in Information Technology

FORMAL METHODS (CT306)

Professor D. Bell
Dr. G. Lyons
Dr. M. Mc Gettrick

Time allowed: *two* hours.
Attempt *three* questions.

1. (a) Explain what is meant by each of the following terms:
   (i) Verification
   (ii) Precondition
   (iii) Postcondition
   (iv) Floyd–Hoare Triple
   (v) Code
   (vi) Program

   (b) Explain the distinction between
   (i) Partial and Total Correctness;
   (ii) Free and Bound Variables;
   (iii) Valid and Invalid Substitutions.

2. (a) Explain the terms *Precondition Strengthening* and *Postcondition Weakening*, giving an example of each.

   (b) Write down the free variables (if any) in each of the following assertions:
   (i) $(\forall_x)(A(x,z) \rightarrow (\exists_y)B(y))$
   (ii) $(\forall_z)(P(z) \wedge Q(x)) \vee (\exists_y)Q(y)$
   (iii) $(\exists_x)(\exists_y)(A(x,y) \wedge B(y,z) \rightarrow A(a,z))$

   (c) Determine which of the following are valid substitutions:
   (i) $(\exists_y)(x \leqslant y)[y/x]$
   (ii) $(\exists_y)(x \leqslant y)[z/x]$
   (iii) $(\forall_x)(\exists_y)(P(x,y) \vee Q(x,z))[v/z]$
   (iv) $P(z) \rightarrow (\exists_y)Q(y,z)[x/y]$

pto.

3. (a) Use the Assignment Axiom to determine which (if any) of the following programs are correct:

   (i) $\vdash \{x > y\}$`x:=y; y:=2`$\{x < y\}$

   (ii) $\vdash \{(x < 0) \wedge (y > 0)\}$`x:=x-1; y:=y+1`$\{x < x + y < y\}$

   (iii) $\vdash \{(x > 0) \wedge (y < 0)\}$`x:=x-1; y:=y+1`$\{y < x + y < x\}$

   (b) Verify the following Floyd–Hoare formula which calculates the maximum `m` of three numbers `i, j, k`:

```
if (i<=j) then
  begin
  if (j<k) then m:=k
  else m:=j
  end
else
  begin
  if (i<k) then m:=k
  else m:=i
  end
```

$\{(m \geq i) \wedge (m \geq j) \wedge (m \geq k)\}$

4. (a) This program calculates the square of a number. Verify its correctness, using the Method of Loop Variable Substitution to find the loop invariant.

```
i:=1; j:=1
while (i<>x) do
  begin
  j:=j+2i+1
  i:=i+1
  end
```

$\{j = x^2\}$

   (b) A `repeat C until S` statement (where `C` is a program and `S` a condition) is described by the following behaviour:

   (1) Execute C

   (2) If S is false goto (1)

   (3) Finish

   This can equivalently be written as

   `C; while` $\neg$ `S do C`

   Given that to verify a while program we must prove

   - $\vdash \{A\}C_1;\ C_2;\ \ldots C_n\{P\}$
   - $\vdash \{P \wedge S\}C\{P\}$
   - $\vdash \{P \wedge \neg S\}C_{n+1};\ C_{n+2};\ \ldots C_m\{B\}$

   write down three equivalent steps that must be proven to verify a `repeat C until S` program.

5. (a) Prove

$\vdash \{(A[x] = 3) \wedge (A[y] = 0) \wedge (A[z] = 6)\}$

```
A[x]:=A[y]
A[y]:=A[z]
A[z]:=A[x]
```

$\{A[x] + A[y] + A[z] = 6\}$

(b) For two positive integers x and y, the following program calculates q (the integer division of x by y) and r (the remainder). Verify its correctness.

$\vdash \{(x \geqslant 0) \wedge (y \geqslant 0)\}$

```
q:=0; r:=x
while (r>=y) do
begin r:=r-y; q:=1+q end
```

$\{(x = qy + r) \wedge (0 \leqslant r < y)\}$

Axioms and Rules of Hoare Logic

Assignment axiom: $\vdash \{P[E/I]\}\ I := E\ \{P\}$

Array assignment axiom: $\vdash \{P[A\{E_1 \leftarrow E_2\}/A]\}\ A[E_1] := E_2\ \{P\}$

where

$\vdash A\{N \leftarrow E\}(N) = E$

$\vdash (N \neq M) \supset (A\{N \leftarrow E\}(M) = A(M))$

Precondition strengthening: $$\frac{\vdash P \supset Q \quad \vdash \{Q\}\ C\ \{R\}}{\vdash \{P\}\ C\ \{R\}}$$

Postcondition weakening: $$\frac{\vdash \{P\}\ C\ \{Q\} \quad \vdash Q \supset R}{\vdash \{P\}\ C\ \{R\}}$$

Sequencing: $$\frac{\vdash \{P\}\ C_1\ \{Q\} \quad \vdash \{Q\}\ C_2\ \{R\}}{\vdash \{P\}\ C_1 ; C_2\ \{R\}}$$

Specification conjunction: $$\frac{\vdash \{P_1\}\ C\ \{Q_1\} \quad \vdash \{P_2\}\ C\ \{Q_2\}}{\vdash \{P_1 \wedge P_2\}\ C\ \{Q_1 \wedge Q_2\}}$$

Specification disjunction: $$\frac{\vdash \{P_1\}\ C\ \{Q_1\} \quad \vdash \{P_2\}\ C\ \{Q_2\}}{\vdash \{P_1 \vee P_2\}\ C\ \{Q_1 \vee Q_2\}}$$

Block: $$\frac{\vdash \{P\}\ C\ \{Q\}}{\vdash \{P\}\ \textsf{begin var } I_1; \ldots; \textsf{ var } I_n; C \textsf{ end}\ \{Q\}}$$

where none of $I_1, \ldots, I_n$ occur in P or Q.

One-armed conditional: $$\frac{\vdash \{P \wedge S\}\ C\ \{Q\} \quad \vdash P \wedge \neg S \supset Q}{\vdash \{P\}\ \textsf{if } S \textsf{ then } C\ \{Q\}}$$

Two-armed conditional: $$\frac{\vdash \{P \wedge S\}\ C_1\ \{Q\} \quad \vdash \{P \wedge \neg S\}\ C_2\ \{Q\}}{\vdash \{P\}\ \textsf{if } S \textsf{ then } C_1 \textsf{ else } C_2\ \{Q\}}$$

While: $$\frac{\vdash \{P \wedge S\}\ C\ \{P\}}{\vdash \{P\}\ \textsf{while } S \textsf{ do } C\ \{P \wedge \neg S\}}$$