IN324: Floyd-Hoare logic

Author : Christophe Garion <garion@isae.fr>
Audience : IN
Date :

1. **Inductive definitions**

   1. give an inductive definitions of natural numbers
   2. give an inductive definitions of binary trees

      Prove the following property of binary trees: "the number n of nodes in a binary tree of height h is at least $n = h$ and at most $n = 2^h - 1$ where h is the depth of the tree".

2. **Natural deduction for propositional logic**

   Prove the following theorems using natural deduction for propositional logic:

   (a) $(a \to (b \to c)) \to ((a \to b) \to (a \to c))$

   (b) $((a \lor b) \to c) \to (b \to c)$

(c) $((a \vee b) \wedge (a \to c) \wedge (b \to c)) \to c$

(d) $a \to \neg\neg a$

3. **Maths, again and again**

   Let E be a set. Model the following mathematical notions using a first-order language. Define precisely the signature of your language.

   (a) $=$ define the "classical" equality relation on E

   (b) $\leq$ is a preorder on E

   (c) $(E, .)$ is a monoid

4. **Natural deduction for first-order logic**

   Prove the following theorems using natural deduction for first-order logic:

   (a) $(\forall x\ \varphi \wedge \psi) \rightarrow (\forall x\ \varphi \wedge \forall x\ \psi)$

   (b) $\exists x \forall y\ \varphi \rightarrow \forall y \exists x\ \varphi$

5. **Floyd-Hoare logic to prove programs**

   For each of the following program:

   - determine the invariant in order to prove the partial correctness of the program
   - determine the variant in order to prove the total correctness of the program
   - annotate the program using Floyd-Hoare logic

   (a) consider the function $\mathbb{N} \rightarrow \mathbb{N}$ defined by the following program:

	Invariant	**Variant**

$\{N \geq 0\}$
```
K := 0;
F := 1;
while (K ≠ N) do
      K := K + 1;
      F := F * K
od
```
$\{F = N!\}$

$\{N \geq 0\}$

```
K := 0;

F := 1;

while (K ≠ N) do

      K := K + 1;

      F := F * K

od
```

$\{F = N!\}$

(b) consider the function $\mathbb{N} \to \mathbb{N}$ defined by the following program:

	Invariant	Variant

$\{N \geq 0\}$
`K := N;`
`F := 1;`
while `(K` $\neq$ `0)` **do**
    `F := F * K;`
    `K := K - 1`
od
$\{F = N!\}$		

$\{N \geq 0\}$

```
K := N;
```

```
F := 1;
```

while `(K` $\neq$ `0)` **do**

```
      F := F * K;
```

```
      K := K - 1
```

od

$\{F = N!\}$

(c) consider the function $\mathbb{N}^2 \to \mathbb{N}^2$ defined by the following program:

Invariant **Variant**

$\{X \geq 0 \wedge Y > 0\}$
```
Q := 0;
R := X;
while (Y ≤ R) do
      Q := Q + 1;
      R := R - Y
od
```
$\{X = Q \times Y + R \wedge 0 \leq Q \wedge 0 \leq R < Y\}$

$\{X \geq 0 \wedge Y > 0\}$

```
Q := 0;

R := X;

while (Y ≤ R) do

      Q := Q + 1;

      R := R - Y

od
```

$\{X = Q \times Y + R \wedge 0 \leq Q \wedge 0 \leq R < Y\}$

(d) consider the function $\mathbb{N}^2 \rightarrow \mathbb{N}^2$ defined by the following program:

Invariant **Variant**

$\{A > 0 \wedge B > 0\}$
```
X := A;
Y := B;
while (X ≠ Y) do
      if (X > Y) then
            X := X - Y
      else
            Y := Y - X
      fi
od
```
$\{X = Y \wedge X > 0 \wedge X = gcd(A, B)\}$

where the *gcd* function is defined by:

$$\forall A \in \mathbb{N}^* \; gcd(A, A) = A$$
$$\forall A \in \mathbb{N}^* \forall B \in \mathbb{N}^* \; gcd(A, B) = gcd(B, A)$$
$$\forall A \in \mathbb{N}^* \forall B \in \mathbb{N}^* \; A > B \rightarrow gcd(A, B) = gcd(A - B, B)$$

$\{A > 0 \wedge B > 0\}$

```
X := A;

Y := B;

while (X ≠ Y) do

      if (X > Y) then

            X := X - Y

      else

            Y := Y - X

      fi
```

od

$\{X = Y \land X > 0 \land X = gcd(A, B)\}$