Dokazovanje pravilnosti

Specifikacija:

Predpis/navodilo/naročilo kaj naj bi program delal

Implementacija

Program, ki zadošča specifikaciji S je implementacija/realizacija S.

```
s := 0 ;
i := 0 ;
while i < 100 do
  i := i + 1 ;
  s := s + i
done
```

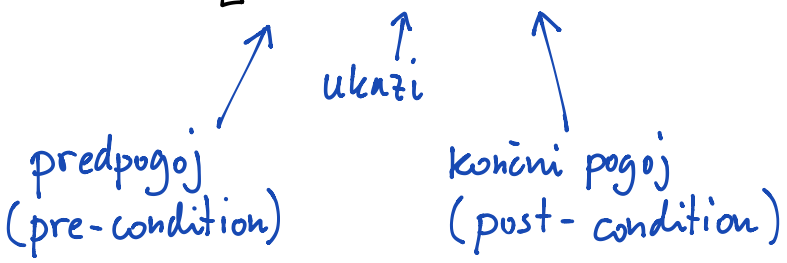
CompCert
(proof assistant)

Hoarova logika

Hoarova trojica

$\{P\}\ c\ \{Q\}$ delna pravilnost

$[P]\ c\ [Q]$ popolna pravilnost



$\{P\}\ c\ \{Q\}$ — če velja P (1) in če se c ustavi (2), potem bo veljal Q

2 predpostavki

1 sklep

$[P]\ c\ [Q]$ — če velja P, potem se c ustavi in bo veljal Q

1 predpostavka

2 sklepa

Primeri

$$\{x = m \wedge y = n\} \quad c \quad \{x = n \wedge y = m\}$$

"c zamenja vrednost spremenljivk x in y"

Pišemo navpično:

$$\begin{array}{l} \{P\} \\ C \\ \{Q\} \end{array} \qquad \begin{array}{l} \{P_1\} \\ C_1 \\ \{P_2\} \\ C_2 \\ \{P_3\} \end{array}$$

Prepletamo spec. in impl.

$$\begin{array}{l} \{x = m \wedge y = n\} \\ \quad x := y; \\ \quad y := x \\ \{x = n \wedge y = m\} \end{array}$$

$$\begin{array}{l} \{x = m \wedge y = n\} \\ \quad t := x; \\ \quad x := y; \\ \quad y := t \\ \{x = n \wedge y = m\} \end{array}$$

$$\begin{array}{l} \{x = m \wedge y = n\} \\ \quad x := x + y; \\ \quad y := x - y; \\ \quad x := x - y \\ \{x = n \wedge y = m\} \end{array}$$

$$\begin{array}{l} \{x = m \wedge y = n\} \\ \quad \text{while true do} \\ \quad\quad \text{skip} \\ \quad \text{done} \\ \{x = n \wedge y = m\} \end{array}$$

$\{ x = m \wedge y = n \}$

$t := m;$

$m := n;$

$n := t$

$\{ x = n \wedge y = m \}$

✓

$\{ x = m \wedge y = n \}$

C

$\{ x = n \wedge y = m \}$

"m in n naj se ne spreminjata"

(sta duhova, ghost variables)

C ne sme omeniti m in n

$\{ x = m \wedge y = n \}$

$x := 42;$

$n := 42;$

$y := m$

$\{ x = n \wedge y = m \}$

✓

Primer: $\{ true \}\ C\ \{ x \leq y \}$

Če se C ustavi, bo veljalo $x \leq y$

$\{ false \}\ C\ \{ Q \}$

↑

vedno velja, ker iz false sledi karkoli

$\{ true \}$

$x := 0;$

$y := 13$

$\{ x \leq y \}$

✓

Primer: $\{ x = m \wedge y = n \}\ C\ \{ x = min(m,n) \wedge y = max(m,n) \}$

(m,n duhova)

↓

```
if x > y then
    t := x; x := y; y := t
else
    skip
end
```

Pravila sklepanja

predpostavke: p_1 p_2 . . . p_n

sklep: q

$$\frac{P' \Rightarrow P \qquad \{ P \}\ c\ \{ Q \} \qquad Q \Rightarrow Q'}{\{ P' \}\ c\ \{ Q' \}}$$

Splošna logična pravila.

$$\frac{\{ P_1 \}\ c\ \{ Q_1 \} \qquad \{ P_2 \}\ c\ \{ Q_2 \}}{\{ P_1 \wedge P_2 \}\ c\ \{ Q_1 \wedge Q_2 \}}$$

Pišemo: $\{P'\} \Downarrow$ (sklepamo)
$\{P\}$
c
$\{Q\} \Downarrow$
$\{Q'\}$

Definicija:

FV(c) = vse spremenljivke, ki se pojavijo v c (free variables)

FA(c) = vse spremenljivke, ki jih c naslavlja na levi strani ":=" (tiste, ki jih bi lahko spremenil) (assigned variables)

FV(if $x \le y$ then $x := y+3$ else $x := z$) = $\{x, y, z\}$

FA(if $x \le y$ then $x := y+3$ else $x := z$) = $\{x\}$

$$\frac{FV(P) \cap FA(c) = \emptyset}{\{ P \}\ c\ \{ P \}} *$$

Če c ne spreminja spremenljivk iz P, potem ne spreminja veljavnosti P

$\{x = m \wedge y = n\}$
$x := x+y$ *
$\{ \ldots ? \ldots \wedge y = n \}$

$$\frac{}{\{ P \}\ \texttt{skip}\ \{ P \}}$$

$$\frac{\{ P \}\ c_1\ \{ Q \} \qquad \{ Q \}\ c_2\ \{ R \}}{\{ P \}\ c_1 ; c_2\ \{ R \}}$$

$\{P\}$
c_1
$\{Q\}$
c_2
$\{R\}$

$$\frac{\{ P \wedge b \}\ c_1\ \{ Q \} \qquad \{ P \wedge \neg b \}\ c_2\ \{ Q \}}{\{ P \}\ \texttt{if b then } c_1 \texttt{ else } c_2 \texttt{ end}\ \{ Q \}}$$

$\{P\}$
if b then
  $\{P \wedge b\}$
  c_1
  $\{Q\}$
else
  $\{P \wedge \neg b\}$
  c_2
  $\{Q\}$
end
$\{Q\}$

$$\frac{\{ P \wedge b \}\ c\ \{ P \}}{\{ P \}\ \texttt{while b do c done}\ \{ \neg\ b \wedge P \}}$$

P invarianta zanke

V praksi

$\{P'\}$ $\Downarrow$ → metoda brittnosti
$\{P\}$
while b do
  $\{P \wedge b\}$
  c
  $\{P\}$
done
$\{P\}$ $\Downarrow$ → običajno precej lahek sklep
$\{Q\}$

$$\frac{}{\{ P[x \mapsto e] \}\ x := e\ \{ P \}}$$

$P[x \mapsto e]$ v P zamenjaj x z e (substitucija)

Popolna pravilnost

Vsa pravila razen spodaj nastetih lahko iz $\{\dots\}\dots\{\dots\}$ spremenimo v $[\dots]\dots[\dots]$, na primer:

$$\frac{[P \wedge b]\ c_1\ [Q] \qquad [P \wedge \neg b]\ c_2\ [Q]}{[P]\ \texttt{if } b \texttt{ then } c_1 \texttt{ else } c_2 \texttt{ end } [Q]}$$

Izjemi:

$$\frac{FV(P) \cap FA(c) = \varnothing}{\{ P \}\ c\ \{ P \}}$$

ne smemo predelati v $\dfrac{FV(P) \cap FA(c) = \varnothing}{[P]\ c\ [P]}$

Protiprimer: $[x>0]$ while true do skip done $[x>0]$

Pravilno:

$$\frac{FV(P) \cap FA(c) = \varnothing \qquad [R]\ c\ [Q]}{[R \wedge P]\ c\ [Q \wedge P]}$$

→ dodaten razlog, zakaj se c ustavi

Kako dokažemo, da se ustavi zanka?

```
while i < b do
   p := p * a ;
   i := i + 1
done
```

i	p	$p = a^i$, $i \leq b$
0	1	✓
1	a	✓
2	a^2	✓
3	a^3	✓
⋮	⋮	⋮

$b - i$ se zmanjšuje