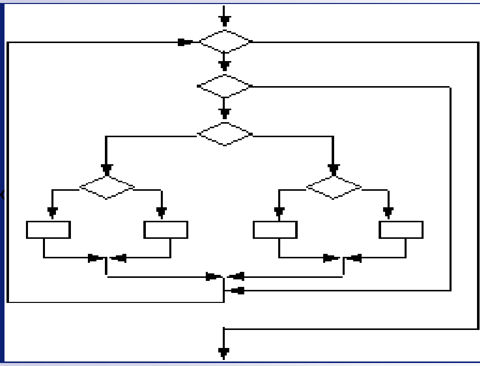
Spring 2008



Program correctness

Proof Outlines

Marcello Bonsangue

Leiden Institute of Advanced Computer Science
Research & Education

Proof outlines

- Formal proofs are long and tedious to follow.
- It is better to organize the proof in small local isolated steps
- We can use the structure of the program to structure our proof!

The idea

- For the program $P = c_1; c_2; c_3; \dots c_n$ we want to show

$$\vdash_{par} \{\phi_0\}\ P\ \{\ \phi_n\}$$

- We can split the problem into smaller ones if we find formulas ϕ_i's such that

$$\vdash_{par} \{\phi_i\}\ c_i \{\phi_{i+1}\}$$

The idea (cont.d)

- Thus we have to find a calculus for presenting a proof $\vdash_{par}\{\phi_0\}\ P\ \{\phi_n\}$ by interleaving formulas with code

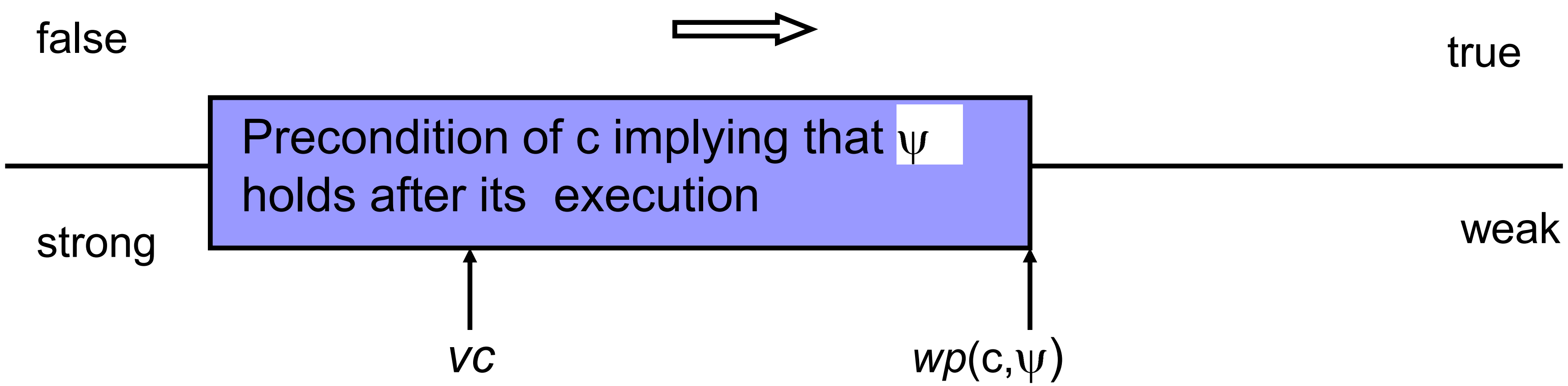
$\{\phi_0\}$
$c_1;$
$\{\phi_1\}$ justification (i.e. skip, ass, if, while, implied)
$c_2;$
$\{\phi_2\}$ justification
$c_3;$
$\vdots$
$\{\phi_{n-1}\}$ justification
c_n
$\{\phi_n\}$

Composition is implicit !

Verification condition

Problem: How can we find the ϕ_i's ?

Solution: Use Hoare rules and calculate verification conditions, i.e. conditions needed to establish the validity of certain assertions.



Skip, assignment, implied

- $$\frac{}{\{\phi\}\ \text{skip}\ \{\phi\}}$$ skip

- $$\frac{}{\{\phi[a/x]\}\ x := a\ \{\phi\}}$$ assignment

- $$\frac{\vdash \phi \Rightarrow \psi}{\{\phi\}\ \{\psi\}}$$ implied

Example

- To prove $\vdash_{par} \{y = 5\}\ x := y + 1\ \{x = 6\}$

$\{y = 5\}$	
$\{y+1 = 6\}$	implied
$x := y + 1$	
$\{x = 6\}$	assignment

we only need to prove the verification condition $y = 5 \Rightarrow y+1 = 6$

Composition, conditional

- $$\frac{\{\phi\}\ c_1\ \{\psi\} \qquad \{\psi\}\ c_2\ \{\varphi\}}{\{\phi\}\ c_1;\ \{\ \psi\ \}\ c_2\ \{\varphi\}}\quad \text{seq}$$

- $$\frac{\{\phi_1\}\ c_1\ \{\psi\} \qquad \{\phi_2\}\ c_2\ \{\psi\}}{\{b \Rightarrow \phi_1 \wedge \neg b \Rightarrow \phi_2\}\ \underline{\text{if}}\ b\ \underline{\text{then}}\{\phi_1\}c_1\{\psi\}\underline{\text{else}}\{\phi_2\}c_2\{\psi\}\ \text{fi}\{\psi\}}\quad \text{if}$$

Example

- To prove $\vdash_{par}$ {true} z:=x; z:=z+y; u:=z {u = x+y}

{true}	
{ x+y = x+y }	implied
z:=x;	
{ z+y = x+y }	assignment
z:=z+y;	
{ z = x+y }	assignment
u:=z	
{ u = x+y }	assignment

we only need to prove the verification condition

$$\text{true} \Rightarrow x+y = x+y$$

Example

Suppose we want to prove

{true}
a := x+1;
if a = 1 then y := 1 else y := a fi
{y = x+1}

Example

{ true }

$\{x+1=1 \Rightarrow 1=x+1 \wedge x+1 \neq 1 \Rightarrow x+1=x+1\}$ implied

a := x+1;

$\{a=1 \Rightarrow 1=x+1 \wedge a \neq 1 \Rightarrow a=x+1\}$ assignment

<u>if</u> a = 1

<u>then</u> {1 = x+1}

y := 1

{ y = x+1} assignment

<u>else</u>

{a = x+1}

y := a

{ y = x+1 } assignment

<u>fi</u>

{ y = x+1 } if-then-else

While statement

$$\frac{\{I \wedge b\}\ c\ \{I\}}{\{I\}\ \underline{\text{while}}\ b\ \underline{\text{do}}\ \{I \wedge b\}\ c\ \{I\}\ \underline{\text{od}}\ \{I \wedge \neg b\}}\quad \text{while}$$

- We must discover an invariant I
  - I need not hold during the execution of c
  - if I holds before c is executed then it holds if and when c terminates.

Invariant

- For any <u>while</u> b <u>do</u> c <u>od</u> these are invariants
  - true
  - false
  - $\neg b$

  because $\{I \wedge b\}$ c $\{I\}$ is valid. However they are useless to prove

  $$\phi \Rightarrow I \qquad \text{or} \qquad I \wedge \neg b \Rightarrow \psi$$

  when considering the while in a context.
- To find a useful invariant it may help to look at the execution of the while and at the relationships among the variables manipulated by the while-body

Example

- Let W = while $x > 0$ do y := x*y; x := x-1 od
- To prove $\{x = n \wedge n \geq 0 \wedge y=1\}$ W $\{ y = n! \}$

iteration	x	y	$x > 0$?
0	6	1	true
1	5	6	true
2	4	30	true
3	3	120	true
4	2	360	true
5	1	720	true
6	0	720	false

Example I

- Invariant Hypothesis $y*x! = n!$

$\{y*x! = n!\}$	
while $x > 0$ do	
$\{y*x! = n! \wedge x > 0\}$	invariant and guard
$\{x*y*(x-1)! = n!\}$	implied
$y := x*y;$	
$\{y*(x-1)! = n!\}$	assignment
$x := x-1$	
$\{y*x! = n!\}$	assignment
od	
$\{y*x! = n! \wedge \neg x > 0\}$	while

correct !!!

Example II

- Since y*x! = n! is an invariant we have

$\{x = n \wedge n \geq 0 \wedge y=1\}$

$\{y*x! = n!\}$ implied

W

$\{y*x! = n! \wedge \neg x > 0\}$ while

$\{y*x! = n! \wedge x \leq 0\}$ implied

$\{y = n!\}$ implied??

The invariant is too weak!

Example III

- Another invariant hypothesis $y*x! = n! \wedge x \geq 0$

$\{y*x! = n! \wedge x \geq 0\}$
<u>while</u> $x > 0$ <u>do</u>
$\{y*x! = n! \wedge x \geq 0 \wedge x > 0\}$ Inv. Hyp. and guard
$\{x*y*(x-1)! = n! \wedge x \geq 1\}$ implied
$y := x*y;$
$\{y*(x-1)! = n! \wedge x-1 \geq 0\}$ assignment
$x := x-1$
$\{y*x! = n! \wedge x \geq 0\}$ assignment
<u>od</u>
$\{y*x! = n! \wedge x \geq 0 \wedge \neg x > 0\}$ while

correct !!!

Example IV

- With the new invariant we have

$\{x = n \wedge n \geq 0 \wedge y=1 \}$

$\{y*x! = n! \wedge x \geq 0 \}$ implied

W

$\{ y*x! = n! \wedge x \geq 0 \wedge \neg x > 0 \}$ while

$\{ y*x! = n! \wedge x = 0 \}$ implied

$\{ y = n! \}$ implied

Yes!