Simplest-Possible Hoare Logic While-Loop Example in JAPE

Program

```
m := 0;
while m < n
  do
  m := m + 1
  od
```

Program With Assertions

{n ≥ 0}	Assumption
m := 0;	
{m ≤ n}	Loop Invariant
while m < n	
do	
m := m + 1	
od	
{m = n}	Expectation

What we enter

```
WHERE DISTINCT m, n IS {n ≥ 0} (m := 0) {m ≤ n} while m < n do m := m+1 od {m = n}
```

As it appears initially in JAPE

```
   ...
1: {n≥0}(m:=0){m≤n}while m<n do m:=m+1 od{m=n}
```

Use the Ntuple Rule

- Ntuple creates two separate triples: one for the initialization and one for the while loop.

```
   ...
1: {n≥0}(m:=0){m≤n}
   ...
2: {m≤n}while m<n do m:=m+1 od{m=n}
3: {n≥0}(m:=0){m≤n}while m<n do m:=m+1 od{m=n} Ntuple 1,2
```

Proving the Initialization

- Because the derived pre-condition is not identical to the assumption, a use of the consequence(L) rule is automatically introduced, along with a companion logical implication.

…

1:	$n\geq 0 \rightarrow 0\leq n$	
2:	$\{0\leq n\}(m:=0)\{m\leq n\}$	variable-assignment
3:	$\{n\geq 0\}(m:=0)\{m\leq n\}$	consequence(L) 1,2

Proving the Implication

- Click on the upper portion to prove the implication as a consequence (rather than using it as a hypothesis).

…

1:	$n\geq 0 \rightarrow 0\leq n$	
2:	$\{0\leq n\}(m:=0)\{m\leq n\}$	variable-assignment
3:	$\{n\geq 0\}(m:=0)\{m\leq n\}$	consequence(L) 1,2
	…	
4:	$\{m\leq n\}$while $m<n$ do $m:=m+1$ od$\{m=n\}$	
5:	$\{n\geq 0\}(m:=0)\{m\leq n\}$while $m<n$ do $m:=m+1$ od$\{m=n\}$	Ntuple 3,4

Resolve the gap using a Rule from the Comparison menu $A\leq B \triangleq B\geq A$

1:	$n\geq 0$	assumption
	…	
2:	$0\leq n$	
3:	$n\geq 0 \rightarrow 0\leq n$	$\rightarrow$ intro 1-2
4:	$\{0\leq n\}(m:=0)\{m\leq n\}$	variable-assignment
5:	$\{n\geq 0\}(m:=0)\{m\leq n\}$	consequence(L) 3,4

1:	$n\geq 0$	assumption
2:	$0\leq n$	$A\leq B \triangleq B\geq A$ 1
3:	$n\geq 0 \rightarrow 0\leq n$	$\rightarrow$ intro 1-2
4:	$\{0\leq n\}(m:=0)\{m\leq n\}$	variable-assignment
5:	$\{n\geq 0\}(m:=0)\{m\leq n\}$	consequence(L) 3,4

Now Expand the while

…

6:	$\{m\leq n\}$while $m<n$ do $m:=m+1$ od$\{m=n\}$	
7:	$\{n\geq 0\}(m:=0)\{m\leq n\}$while $m<n$ do $m:=m+1$ od$\{m=n\}$	Ntuple 5,6

This generates several unfinished sub-proofs:

6:	$\{m\leq n\wedge m<n\}(m:=m+1)\{m\leq n\}$	Partial correctness of loop body
	…	
7:	$m\leq n\wedge m<n\rightarrow\_M>0$	Continuation condition for loop body
8:	integer Km	assumption
	…	Termination condition for loop body
9:	$\{m\leq n\wedge m<n\wedge\_M=Km\}(m:=m+1)\{\_M<Km\}$	
10:	$\{m\leq n\}$while $m<n$ do $m:=m+1$ od$\{m\leq n\wedge\neg(m<n)\}$	while 6,7,8-9
	…	
11:	$m\leq n\wedge\neg(m<n)\rightarrow m=n$	Exit consequence implication for loop
12:	$\{m\leq n\}$while $m<n$ do $m:=m+1$ od$\{m=n\}$	consequence(R) 10,11
13:	$\{n\geq 0\}(m:=0)\{m\leq n\}$while $m<n$ do $m:=m+1$ od$\{m=n\}$	Ntuple 5,12

Partial correctness of loop body

…

6: $\{m\leq n\wedge m<n\}(m:=m+1)\{m\leq n\}$

Use the assignment rule, with an implied consequence(L). This generates a logic implication for the consequence(L) rule.

…

6:	$m\leq n\wedge m<n\rightarrow m+1\leq n$	
7:	$\{m+1\leq n\}(m:=m+1)\{m\leq n\}$	variable-assignment
8:	$\{m\leq n\wedge m<n\}(m:=m+1)\{m\leq n\}$	consequence(L) 6,7

The logic implication remains to be proved.

Proof of Upper Implication

6:	$m\leq n\wedge m<n$	assumption
7:	$m\leq n$	$\wedge$ elim 6
8:	$m<n$	$\wedge$ elim 6
	…	
9:	$m+1\leq n$	

Close using another Comparison rule on line 9.

$A+1\leq B \triangleq A<B$

6:	$m\leq n\wedge m<n$	assumption
7:	$m<n$	$\wedge$ elim 6
8:	$m+1\leq n$	$A+1\leq B \triangleq A<B$ 7

Proof of Lower Implication

...

15: m≤n∧¬(m<n)→m=n

Use a Lemma we've introduced:

16:	m≤n∧¬(m<n)	assumption
17:	m≤n	∧ elim 16
18:	¬(m<n)	∧ elim 16
19:	m=n	x≤n, ¬(x<n) ⊢ x=n 17,18

What Remains

- We need a *variant* that unifies with _M
- The variant n-m should work.

...

12:	m≤n∧m<n→_M>0	
13:	integer Km	assumption
	...	
14:	{m≤n∧m<n∧_M=Km}(m:=m+1){_M<Km}	
15:	{m≤n}while m<n do m:=m+1 od{m≤n∧¬(m<n)}	while 11,12,13–14

...

12:	m≤n∧m<n→n−m>0	
13:	integer Km	assumption
	...	
14:	{m≤n∧m<n∧n−m=Km}(m:=m+1){n−m<Km}	

Prove the Upper Implication

...

12: m≤n∧m<n→n−m>0

Apply another "obvious" lemma

12:	m≤n∧m<n	assumption
13:	m<n	∧ elim 12
14:	n−m>0	m<n ⊢ n−m>0 13
15:	m≤n∧m<n→n−m>0	→ intro 12–14

Prove the while termination body

16:	integer Km
	...
17:	{m≤n∧m<n∧n−m=Km}(m:=m+1){n−m<Km}

Another consequence(L) is introduced:

16:	integer Km	assumption
	...	
17:	m≤n∧m<n∧n−m=Km→n−(m+1)<Km	
18:	{n−(m+1)<Km}(m:=m+1){n−m<Km}	variable-assignment
19:	{m≤n∧m<n∧n−m=Km}(m:=m+1){n−m<Km}	consequence(L) 17,18

One more "obvious" lemma completes the proof.

16:	integer Km	assumption
17:	m≤n∧m<n∧n−m=Km	assumption
18:	n−m=Km	∧ elim 17
19:	n−(m+1)<Km	n−m=X ⊢ n−(m+1)<X 18
20:	m≤n∧m<n∧n−m=Km→n−(m+1)<Km	→ intro 17–19
21:	{n−(m+1)<Km}(m:=m+1){n−m<Km}	variable-assignment
22:	{m≤n∧m<n∧n−m=Km}(m:=m+1){n−m<Km}	consequence(L) 20,21

1:	n≥0	assumption
2:	0≤n	A≤B≜B≥A 1
3:	n≥0→0≤n	→ intro 1–2
4:	{0≤n}(m:=0){m≤n}	variable-assignment
5:	{n≥0}(m:=0){m≤n}	consequence(L) 3,4
6:	m≤n∧m<n	assumption
7:	m<n	∧ elim 6
8:	m+1≤n	A+1≤B≜A<B 7
9:	m≤n∧m<n→m+1≤n	→ intro 6–8
10:	{m+1≤n}(m:=m+1){m≤n}	variable-assignment
11:	{m≤n∧m<n}(m:=m+1){m≤n}	consequence(L) 9,10
12:	m≤n∧m<n	assumption
13:	m<n	∧ elim 12
14:	n−m>0	m<n ⊢ n−m>0 13
15:	m≤n∧m<n→n−m>0	→ intro 12–14
16:	integer Km	assumption
17:	m≤n∧m<n∧n−m=Km	assumption
18:	n−m=Km	∧ elim 17
19:	n−(m+1)<Km	n−m=X ⊢ n−(m+1)<X 18
20:	m≤n∧m<n∧n−m=Km→n−(m+1)<Km	→ intro 17–19
21:	{n−(m+1)<Km}(m:=m+1){n−m<Km}	variable-assignment
22:	{m≤n∧m<n∧n−m=Km}(m:=m+1){n−m<Km}	consequence(L) 20,21

Provided:
DISTINCT m, n