Algoritmos y Estructuras de Datos I

Practico 3

(Parar antes)

Damián Barsotti

Fa.M.A.F., Universidad Nacional de Córdoba, Argentina

Todos mayores que cero

Programa derivado

Const $N : Int; a : array[0, N) of\ Int;$

Var $r : Bool; n : Int;$

$\{P : N \geq 0\}$

$r, n := True, 0;$

$\{I : r = \langle \forall i : 0 \leq i < n : a.i > 0 \rangle \wedge 0 \leq n \leq N\}$

do $n \neq N \rightarrow$

$\{I \wedge B\}$

$r, n := r \wedge a.n > 0, n + 1$

$\{I\}$

od

$\{I \wedge \neg B\}$

$\{Q : r = \langle \forall i : 0 \leq i < N : a.i > 0 \rangle\}$

Todos mayores que cero

Programa derivado

Const $N : Int; a : array[0, N) of\ Int;$
Var $r : Bool; n : Int;$
$\{P : N \geq 0\}$
$r, n := True, 0;$
$\{I : r = \langle \forall i : 0 \leq i < n : a.i > 0 \rangle \wedge 0 \leq n \leq N\}$
do $n \neq N \rightarrow$
$\{I \wedge B\}$
$r, n := r \wedge a.n > 0, n + 1$
$\{I\}$
od
$\{I \wedge \neg B\}$
$\{Q : r = \langle \forall i : 0 \leq i < N : a.i > 0 \rangle\}$

Poco eficiente

Todos mayores que cero

Programa derivado

Const $N : Int; a : array[0, N) of\ Int;$
Var $r : Bool; n : Int;$
$\{P : N \geq 0\}$
$r, n := True, 0;$
$\{I : r = \langle \forall i\ : 0 \leq i < n :\ a.i > 0 \rangle \wedge 0 \leq n \leq N\}$
do $n \neq N \rightarrow$
$\{I \wedge B\}$
$r, n := r \wedge a.n > 0, n + 1$
$\{I\}$
od
$\{I \wedge \neg B\}$
$\{Q : r = \langle \forall i\ : 0 \leq i < N :\ a.i > 0 \rangle\}$

Poco eficiente

- Si arreglo grande

Todos mayores que cero

Programa derivado

Const $N : Int; a : array[0, N) of\ Int;$

Var $r : Bool; n : Int;$

$\{P : N \geq 0\}$

$r, n := True, 0;$

$\{I : r = \langle \forall i\ : 0 \leq i < n :\ a.i > 0\, \rangle \wedge 0 \leq n \leq N\}$

do $n \neq N \rightarrow$

$\{I \wedge B\}$

$r, n := r \wedge a.n > 0, n + 1$

$\{I\}$

od

$\{I \wedge \neg B\}$

$\{Q : r = \langle \forall i\ : 0 \leq i < N :\ a.i > 0\, \rangle\}$

Poco eficiente

- Si arreglo grande
- $a.1 = -1$

Todos mayores que cero

Programa derivado

Const $N : Int; a : array[0, N) of\ Int;$

Var $r : Bool; n : Int;$

$\{P : N \geq 0\}$

$r, n := True, 0;$

$\{I : r = \langle \forall i : 0 \leq i < n : a.i > 0 \rangle \wedge 0 \leq n \leq N\}$

do $n \neq N \rightarrow$

$\{I \wedge B\}$

$r, n := r \wedge a.n > 0, n + 1$

$\{I\}$

od

$\{I \wedge \neg B\}$

$\{Q : r = \langle \forall i : 0 \leq i < N : a.i > 0 \rangle\}$

Poco eficiente

- Si arreglo grande
- $a.1 = -1$
- recorre todo el arreglo sin necesidad

Mejora

Programa

Const $N : Int; a : array[0, N) of\ Int;$

Var $r : Bool; n : Int;$

$\{P : N \geq 0\}$

$r, n := True, 0;$

do $n \neq N \rightarrow$

$r, n := r \wedge a.n > 0, n + 1$

od

$\{Q : r = \langle \forall i : 0 \leq i < N : a.i > 0 \rangle\}$

Mejora

Programa

Const $N : Int; a : array[0, N) of\ Int;$

Var $r : Bool; n : Int;$

$\{P : N \geq 0\}$

$r, n := True, 0;$

do $n \neq N \rightarrow$

$r, n := r \wedge a.n > 0, n + 1$

od

$\{Q : r = \langle \forall i : 0 \leq i < N : a.i > 0 \rangle\}$

- r comienza en *True*.

Mejora

Programa

Const $N : Int; a : array[0, N) of\ Int;$
Var $r : Bool; n : Int;$
$\{P : N \geq 0\}$
$r, n := True, 0;$
do $n \neq N \rightarrow$
$r, n := r \wedge a.n > 0, n + 1$
od
$\{Q : r = \langle \forall i : 0 \leq i < N : a.i > 0 \rangle\}$

- r comienza en *True*.
- Si se hace *False* no hace falta seguir

Mejora

Programa

Const $N : Int; a : array[0, N) of\ Int;$
Var $r : Bool; n : Int;$
$\{P : N \geq 0\}$
$r, n := True, 0;$
do $n \neq N \wedge r \rightarrow$
$r, n := r \wedge a.n > 0, n + 1$
od
$\{Q : r = \langle \forall i : 0 \leq i < N : a.i > 0 \rangle\}$

- r comienza en *True*.
- Si se hace *False* no hace falta seguir
- Fortalezcamos la guarda

$$B' : n \neq N \wedge r$$

Pasos para demostrar el nuevo programa

1. Encontrar invariante candidato I.

Pasos para demostrar el nuevo programa

1. Encontrar invariante candidato I.
2. Inicialización.

Pasos para demostrar el nuevo programa

1. Encontrar invariante candidato I.
2. Inicialización.
3. Finalización $I \wedge \neg B' \Rightarrow Q$.

Pasos para demostrar el nuevo programa

1. Encontrar invariante candidato I.
2. Inicialización.
3. Finalización $I \wedge \neg B' \Rightarrow Q$.
4. Cuerpo del bucle $\{I \wedge B'\}\ S\ \{I\}$.

Pasos para demostrar el nuevo programa

1. Encontrar invariante candidato I.
2. Inicialización.
3. Finalización $I \wedge \neg B' \Rightarrow Q$.
4. Cuerpo del bucle $\{I \wedge B'\}\ S\ \{I\}$.
5. Cota positiva $I \wedge B' \Rightarrow t \geq 0$.

Pasos para demostrar el nuevo programa

1. Encontrar invariante candidato I.
2. Inicialización.
3. Finalización $I \wedge \neg B' \Rightarrow Q$.
4. Cuerpo del bucle $\{I \wedge B'\}\ S\ \{I\}$.
5. Cota positiva $I \wedge B' \Rightarrow t \geq 0$.
6. Cota disminuye $\{I \wedge B' \wedge t = T\}\ S\ \{t < T\}$.

1. Encontrar invariante candidato

- El programa ejecuta menos iteraciones

1. Encontrar invariante candidato

- El programa ejecuta menos iteraciones
- No van a aparecer nuevos estados

1. Encontrar invariante candidato

- El programa ejecuta menos iteraciones
- No van a aparecer nuevos estados

$\Rightarrow$ Probemos el mismo invariante

$$I : r = \langle \forall i \ : 0 \leq i < n : \ a.i > 0 \rangle \wedge 0 \leq n \leq N$$

2. Inicialización

Const $N : Int; a : array[0, N) of\ Int;$
Var $r : Bool; n : Int;$
$\{P : N \geq 0\}$
$r, n := True, 0;$?
$\{I : r = \langle \forall i : 0 \leq i < n : a.i > 0 \rangle \wedge 0 \leq n \leq N\}$
do $n \neq N \wedge r \rightarrow$
$\{I \wedge B'\}$
$r, n := r \wedge a.n > 0, n + 1$?
$\{I\}$
od
$\{I \wedge \neg B'\}$?
$\{Q : r = \langle \forall i : 0 \leq i < N : a.i > 0 \rangle\}$

2. Inicialización

Const $N : Int; a : array[0, N) of\ Int;$
Var $r : Bool; n : Int;$
$\{P : N \geq 0\}$
$r, n := True, 0;$?
$\{I : r = \langle \forall i : 0 \leq i < n : a.i > 0 \rangle \wedge 0 \leq n \leq N\}$
do $n \neq N \wedge r \rightarrow$
$\{I \wedge B'\}$
$r, n := r \wedge a.n > 0, n + 1$?
$\{I\}$
od
$\{I \wedge \neg B'\}$?
$\{Q : r = \langle \forall i : 0 \leq i < N : a.i > 0 \rangle\}$

- Es el programa derivado

2. Inicialización

Const $N : Int; a : array[0, N) of\ Int;$
Var $r : Bool; n : Int;$
$\{P : N \geq 0\}$
$r, n := True, 0;$ $\heartsuit$
$\{I : r = \langle \forall i : 0 \leq i < n : a.i > 0 \rangle \wedge 0 \leq n \leq N\}$
do $n \neq N \wedge r \rightarrow$
$\{I \wedge B'\}$
$r, n := r \wedge a.n > 0, n + 1$?
$\{I\}$
od
$\{I \wedge \neg B'\}$?
$\{Q : r = \langle \forall i : 0 \leq i < N : a.i > 0 \rangle\}$

- Es el programa derivado

$\Rightarrow$ Esta probado

4. Cuerpo del bucle

Dejemos para el final 3. Finalización

4. Cuerpo del bucle

Dejemos para el final 3. Finalización

Const $N : Int; a : array[0, N) of\ Int;$
Var $r : Bool; n : Int;$
$\{P : N \geq 0\}$
$r, n := True, 0;$ ♡
$\{I : r = \langle \forall i \ : 0 \leq i < n : \ a.i > 0 \rangle \wedge 0 \leq n \leq N\}$
do $n \neq N \wedge r \rightarrow$
$\{I \wedge B'\}$
$r, n := r \wedge a.n > 0, n + 1$?
$\{I\}$
od
$\{I \wedge \neg B'\}$?
$\{Q : r = \langle \forall i \ : 0 \leq i < N : \ a.i > 0 \rangle\}$

4. Cuerpo del bucle

Dejemos para el final 3. Finalización

Const $N : Int; a : array[0, N) of\ Int;$
Var $r : Bool; n : Int;$
$\{P : N \geq 0\}$
$r, n := True, 0;$ ♡
$\{I : r = \langle \forall i \ : 0 \leq i < n : \ a.i > 0 \rangle \wedge 0 \leq n \leq N\}$
do $n \neq N \wedge r \rightarrow$
$\{I \wedge B'\}$
$r, n := r \wedge a.n > 0, n + 1$?
$\{I\}$
od
$\{I \wedge \neg B'\}$?
$\{Q : r = \langle \forall i \ : 0 \leq i < N : \ a.i > 0 \rangle\}$

Hay que probar

$$\{I \wedge B'\}$$
$$r, n := r \wedge a.n > 0, n + 1$$
$$\{I\}$$

4. Cuerpo del bucle

Demostración

4. Cuerpo del bucle

Demostración

$$\{I \ \wedge \ B'\} \ r, n := r \ \wedge \ a.n > 0, n + 1 \ \{I\}$$

4. Cuerpo del bucle

Demostración

$$
\begin{array}{ll}
 & \{I \wedge B'\}\ r, n := r \wedge a.n > 0, n+1\ \{I\} \\
\Leftarrow & \{\ \text{Fortalecimiento de precodición con } I \wedge B' \Rightarrow I \wedge B\ \} \\
 & \{I \wedge B\}\ r, n := r \wedge a.n > 0, n+1\ \{I\}
\end{array}
$$

4. Cuerpo del bucle

Demostración

$$
\begin{array}{ll}
& \{I \wedge B'\}\ r, n := r \wedge a.n > 0, n+1\ \{I\} \\
\Leftarrow & \{ \text{ Fortalecimiento de precodición con } I \wedge B' \Rightarrow I \wedge B \ \} \\
& \{I \wedge B\}\ r, n := r \wedge a.n > 0, n+1\ \{I\} \\
\equiv & \{ \text{ Derivación programa original } \} \\
& \mathit{True}
\end{array}
$$

4. Cuerpo del bucle

Demostración

$$
\begin{array}{ll}
 & \{I \wedge B'\}\ r, n := r \wedge a.n > 0, n + 1\ \{I\} \\
\Leftarrow & \{ \text{ Fortalecimiento de precodición con } I \wedge B' \Rightarrow I \wedge B \ \} \\
 & \{I \wedge B\}\ r, n := r \wedge a.n > 0, n + 1\ \{I\} \\
\equiv & \{ \text{ Derivación programa original } \} \\
 & True
\end{array}
$$

Por lo tanto

$$
\begin{array}{c}
\{I \wedge B'\} \\
r, n := r \wedge a.n > 0, n + 1 \\
\{I\}
\end{array}
$$

5. Cota positiva

- Usemos la misma cota candidata $t : N - n$

5. Cota positiva

- Usemos la misma cota candidata $t : N - n$
- Hay que demostrar

$$I \wedge B' \Rightarrow N - n \geq 0$$

5. Cota positiva

- Usemos la misma cota candidata $t : N - n$
- Hay que demostrar

$$I \wedge B' \Rightarrow N - n \geq 0$$

Demostración

5. Cota positiva

- Usemos la misma cota candidata $t : N - n$
- Hay que demostrar

$$I \wedge B' \Rightarrow N - n \geq 0$$

Demostración

$$I \wedge B'$$

5. Cota positiva

- Usemos la misma cota candidata $t : N - n$
- Hay que demostrar

$$I \wedge B' \Rightarrow N - n \geq 0$$

Demostración

$$\begin{array}{ll} & I \wedge B' \\ \Rightarrow & \{ \text{ Monotonía } B' \Rightarrow B \ \} \\ & I \wedge B \end{array}$$

5. Cota positiva

- Usemos la misma cota candidata $t : N - n$
- Hay que demostrar

$$I \wedge B' \Rightarrow N - n \geq 0$$

Demostración

$$
\begin{array}{ll}
 & I \wedge B' \\
\Rightarrow & \{ \text{ Monotonía } B' \Rightarrow B \ \} \\
 & I \wedge B \\
\Rightarrow & \{ \text{ Demostración programa original } \} \\
 & N - n \geq 0
\end{array}
$$

5. Cota positiva

- Usemos la misma cota candidata $t : N - n$
- Hay que demostrar

$$I \wedge B' \Rightarrow N - n \geq 0$$

Demostración

$$\begin{array}{ll} & I \wedge B' \\ \Rightarrow & \{ \text{ Monotonía } B' \Rightarrow B \ \} \\ & I \wedge B \\ \Rightarrow & \{ \text{ Demostración programa original } \} \\ & N - n \geq 0 \end{array}$$

Por transitividad implicación

$$I \wedge B' \Rightarrow N - n \geq 0$$

6. Cota disminuye

Demostración

6. Cota disminuye

Demostración

$$\{I \wedge B' \wedge N - n = T\}\ r, n := r \wedge a.n > 0, n + 1\ \{N - n < T\}$$

6. Cota disminuye

Demostración

$$\begin{array}{ll} & \{I \wedge B' \wedge N - n = T\}\ r, n := r \wedge a.n > 0, n + 1\ \{N - n < T\} \\ \Leftarrow & \{\text{ Fortalecimiento de precodición }\} \\ & \{I \wedge B \wedge N - n = T\}\ r, n := r \wedge a.n > 0, n + 1\ \{N - n < T\} \end{array}$$

6. Cota disminuye

Demostración

$$
\begin{array}{ll}
 & \{I \wedge B' \wedge N - n = T\}\ r, n := r \wedge a.n > 0, n+1\ \{N - n < T\} \\
\Leftarrow & \{ \text{ Fortalecimiento de precodición } \} \\
 & \{I \wedge B \wedge N - n = T\}\ r, n := r \wedge a.n > 0, n+1\ \{N - n < T\} \\
\equiv & \{ \text{ Derivación programa original } \} \\
 & \mathit{True}
\end{array}
$$

6. Cota disminuye

Demostración

$$
\begin{array}{ll}
 & \{I \wedge B' \wedge N - n = T\}\ r, n := r \wedge a.n > 0, n+1\ \{N - n < T\} \\
\Leftarrow & \{\text{ Fortalecimiento de precodición }\} \\
 & \{I \wedge B \wedge N - n = T\}\ r, n := r \wedge a.n > 0, n+1\ \{N - n < T\} \\
\equiv & \{\text{ Derivación programa original }\} \\
 & \mathit{True}
\end{array}
$$

Por lo tanto

$$
\begin{array}{c}
\{I \wedge B' \wedge N - n = T\} \\
r, n := r \wedge a.n > 0, n+1 \\
\{N - n < T\}
\end{array}
$$

6. Cota disminuye

Demostración

$$\begin{array}{ll}
 & \{I \wedge B' \wedge N - n = T\}\ r, n := r \wedge a.n > 0, n+1\ \{N - n < T\} \\
\Leftarrow & \{ \text{ Fortalecimiento de precodición } \} \\
 & \{I \wedge B \wedge N - n = T\}\ r, n := r \wedge a.n > 0, n+1\ \{N - n < T\} \\
\equiv & \{ \text{ Derivación programa original } \} \\
 & True
\end{array}$$

Por lo tanto

$$\begin{array}{c}
\{I \wedge B' \wedge N - n = T\} \\
r, n := r \wedge a.n > 0, n+1 \\
\{N - n < T\}
\end{array}$$

Falta 3. Finalización

3. Finalización

Const $N : Int; a : array[0, N) of\ Int;$
Var $r : Bool; n : Int;$
$\{P : N \geq 0\}$
$r, n := True, 0;$ $\heartsuit$
$\{I : r = \langle \forall i\ : 0 \leq i < n :\ a.i > 0 \rangle \wedge 0 \leq n \leq N\}$
do $n \neq N \wedge r \rightarrow$
$\{I \wedge B'\}$
$r, n := r \wedge a.n > 0, n + 1$ $\heartsuit$
$\{I\}$
od
$\{I \wedge \neg B'\}$?
$\{Q : r = \langle \forall i\ : 0 \leq i < N :\ a.i > 0 \rangle\}$

3. Finalización

Const $N : Int; a : array[0, N) of\ Int;$
Var $r : Bool; n : Int;$
$\{P : N \geq 0\}$
$r, n := True, 0;$ ♡
$\{I : r = \langle \forall i \ : 0 \leq i < n : \ a.i > 0 \rangle \wedge 0 \leq n \leq N\}$
do $n \neq N \wedge r \rightarrow$
$\{I \wedge B'\}$
$r, n := r \wedge a.n > 0, n + 1$ ♡
$\{I\}$
od
$\{I \wedge \neg B'\}$?
$\{Q : r = \langle \forall i \ : 0 \leq i < N : \ a.i > 0 \rangle\}$

Hay que probar $I \wedge \neg B' \Rightarrow Q$

3. Finalización

Const $N : Int; a : array[0, N) of\ Int;$
Var $r : Bool; n : Int;$
$\{P : N \geq 0\}$
$r, n := True, 0;$ $\heartsuit$
$\{I : r = \langle \forall i \ : 0 \leq i < n : \ a.i > 0 \rangle \wedge 0 \leq n \leq N\}$
do $n \neq N \wedge r \rightarrow$
$\{I \wedge B'\}$
$r, n := r \wedge a.n > 0, n + 1$ $\heartsuit$
$\{I\}$
od
$\{I \wedge \neg B'\}$?
$\{Q : r = \langle \forall i \ : 0 \leq i < N : \ a.i > 0 \rangle\}$

Hay que probar $I \wedge \neg B' \Rightarrow Q$

No sale con transitividad (prueba anterior)

$$I \wedge \neg B' \Leftarrow I \wedge \neg B \Rightarrow Q \qquad \text{(ya que } \neg B \Rightarrow \neg B'\text{)}$$

3. Finalización (demo)

Demostración

3. Finalización (demo)

Demostración

$I \wedge \neg B' \Rightarrow Q$

3. Finalización (demo)

Demostración

$$
\begin{aligned}
& I \wedge \neg B' \Rightarrow Q \\
\equiv\ & \{ \text{ Def } B', \text{ morgan } \} \\
& I \wedge (n = N \vee \neg r) \Rightarrow Q
\end{aligned}
$$

3. Finalización (demo)

Demostración

$$
\begin{array}{ll}
 & I \wedge \neg B' \Rightarrow Q \\
\equiv & \{ \text{ Def } B', \text{ morgan } \} \\
 & I \wedge (n = N \vee \neg r) \Rightarrow Q \\
\equiv & \{ \text{ Distr. } \wedge \vee \} \\
 & (I \wedge n = N) \vee (I \wedge \neg r) \Rightarrow Q
\end{array}
$$

3. Finalización (demo)

Demostración

$$
\begin{array}{ll}
 & I \wedge \neg B' \Rightarrow Q \\
\equiv & \{ \text{ Def } B', \text{ morgan } \} \\
 & I \wedge (n = N \vee \neg r) \Rightarrow Q \\
\equiv & \{ \text{ Distr. } \wedge \ \vee \ \} \\
 & (I \wedge n = N) \vee (I \wedge \neg r) \Rightarrow Q \\
\equiv & \{ \text{ Implicacion de la disyunción } \} \\
 & (\underline{I \wedge n = N \Rightarrow Q}) \wedge (I \wedge \neg r \Rightarrow Q)
\end{array}
$$

3. Finalización (demo)

Demostración

$$
\begin{array}{ll}
 & I \wedge \neg B' \Rightarrow Q \\
\equiv & \{ \text{ Def } B', \text{ morgan } \} \\
 & I \wedge (n = N \vee \neg r) \Rightarrow Q \\
\equiv & \{ \text{ Distr. } \wedge \ \vee \ \} \\
 & (I \wedge n = N) \vee (I \wedge \neg r) \Rightarrow Q \\
\equiv & \{ \text{ Implicacion de la disyunción } \} \\
 & (\underline{I \wedge n = N \Rightarrow Q}) \wedge (I \wedge \neg r \Rightarrow Q) \\
\equiv & \{ \text{ Derivación programa original. Neutro } \wedge \ \} \\
 & I \wedge \neg r \Rightarrow Q
\end{array}
$$

3. Finalización (demo)

Demostración

$$
\begin{aligned}
& I \wedge \neg B' \Rightarrow Q \\
\equiv\ & \{ \text{ Def } B' \text{, morgan } \} \\
& I \wedge (n = N \vee \neg r) \Rightarrow Q \\
\equiv\ & \{ \text{ Distr. } \wedge\ \vee \ \} \\
& (I \wedge n = N) \vee (I \wedge \neg r) \Rightarrow Q \\
\equiv\ & \{ \text{ Implicacion de la disyunción } \} \\
& (\underline{I \wedge n = N \Rightarrow Q}) \wedge (I \wedge \neg r \Rightarrow Q) \\
\equiv\ & \{ \text{ Derivación programa original. Neutro } \wedge \ \} \\
& I \wedge \neg r \Rightarrow Q
\end{aligned}
$$

Hay que demostrar solo

$$
\begin{gathered}
r = \langle \forall i \ : 0 \leq i < n : \ a.i > 0 \rangle \wedge 0 \leq n \leq N \wedge \neg r \\
\Rightarrow r = \langle \forall i \ : 0 \leq i < N : \ a.i > 0 \rangle
\end{gathered}
$$

3. Finalización (demo)

Demostración

$$
\begin{array}{ll}
 & I \wedge \neg B' \Rightarrow Q \\
\equiv & \{ \text{ Def } B', \text{ morgan } \} \\
 & I \wedge (n = N \vee \neg r) \Rightarrow Q \\
\equiv & \{ \text{ Distr. } \wedge \vee \ \} \\
 & (I \wedge n = N) \vee (I \wedge \neg r) \Rightarrow Q \\
\equiv & \{ \text{ Implicacion de la disyunción } \} \\
 & (\underline{I \wedge n = N \Rightarrow Q}) \wedge (I \wedge \neg r \Rightarrow Q) \\
\equiv & \{ \text{ Derivación programa original. Neutro } \wedge \ \} \\
 & I \wedge \neg r \Rightarrow Q
\end{array}
$$

Hay que demostrar solo

$$
r = \langle \forall i \ : 0 \leq i < n : \ a.i > 0 \rangle \wedge 0 \leq n \leq N \wedge \neg r \\
\Rightarrow r = \langle \forall i \ : 0 \leq i < N : \ a.i > 0 \rangle
$$

Supongo el antecedente y hago *True* el consecuente

3. Finalización (demo)

Sup $r = \langle \forall i \ : 0 \leq i < n : \ a.i > 0 \rangle \wedge 0 \leq n \leq N \wedge \neg r$

$$r = \left\langle \forall i \ : \underline{0 \leq i < N} : \ a.i > 0 \right\rangle$$

3. Finalización (demo)

Sup $r = \langle \forall i \ : 0 \leq i < n : \ a.i > 0 \rangle \wedge 0 \leq n \leq N \wedge \neg r$

$$
\begin{array}{ll}
 & r = \Big\langle \forall i \ : \underline{0 \leq i < N} : \ a.i > 0 \Big\rangle \\
\equiv & \{ \text{ Arit., suposición } 0 \leq n \leq N \ \} \\
 & r = \langle \forall i \ : 0 \leq i < n \vee n \leq i < N : \ a.i > 0 \rangle
\end{array}
$$

3. Finalización (demo)

Sup $r = \langle \forall i : 0 \leq i < n : a.i > 0 \rangle \wedge 0 \leq n \leq N \wedge \neg r$

$$
\begin{array}{ll}
 & r = \Big\langle \forall i : \underline{0 \leq i < N} : a.i > 0 \Big\rangle \\
\equiv & \{ \text{ Arit., suposición } 0 \leq n \leq N \ \} \\
 & r = \langle \forall i : 0 \leq i < n \vee n \leq i < N : a.i > 0 \rangle \\
\equiv & \{ \text{ Partición de Rango } \} \\
 & r = \underline{\langle \forall i : 0 \leq i < n : a.i > 0 \rangle} \wedge \langle \forall i : n \leq i < N : a.i > 0 \rangle
\end{array}
$$

3. Finalización (demo)

Sup $r = \langle \forall i \ : 0 \leq i < n : \ a.i > 0 \rangle \wedge 0 \leq n \leq N \wedge \neg r$

$$
\begin{array}{ll}
 & r = \left\langle \forall i \ : \underline{0 \leq i < N} : \ a.i > 0 \right\rangle \\
\equiv & \{ \text{ Arit., suposición } 0 \leq n \leq N \ \} \\
 & r = \langle \forall i \ : 0 \leq i < n \vee n \leq i < N : \ a.i > 0 \rangle \\
\equiv & \{ \text{ Partición de Rango } \} \\
 & r = \underline{\langle \forall i \ : 0 \leq i < n : \ a.i > 0 \rangle} \wedge \langle \forall i \ : n \leq i < N : \ a.i > 0 \rangle \\
\equiv & \{ \text{ Sup } \} \\
 & \underline{r} = \underline{r} \wedge \langle \forall i \ : n \leq i < N : \ a.i > 0 \rangle
\end{array}
$$

3. Finalización (demo)

Sup $r = \langle \forall i : 0 \leq i < n : a.i > 0 \rangle \wedge 0 \leq n \leq N \wedge \neg r$

$$
\begin{aligned}
& r = \langle \forall i : \underline{0 \leq i < N} : a.i > 0 \rangle \\
\equiv & \ \{ \text{ Arit., suposición } 0 \leq n \leq N \ \} \\
& r = \langle \forall i : 0 \leq i < n \vee n \leq i < N : a.i > 0 \rangle \\
\equiv & \ \{ \text{ Partición de Rango } \} \\
& r = \underline{\langle \forall i : 0 \leq i < n : a.i > 0 \rangle} \wedge \langle \forall i : n \leq i < N : a.i > 0 \rangle \\
\equiv & \ \{ \text{ Sup } \} \\
& \underline{r} = \underline{r} \wedge \langle \forall i : n \leq i < N : a.i > 0 \rangle \\
\equiv & \ \{ \text{ Doble negación } \} \\
& \neg(\underline{\neg r}) = \neg(\underline{\neg r}) \wedge \langle \forall i : n \leq i < N : a.i > 0 \rangle
\end{aligned}
$$

3. Finalización (demo)

Sup $r = \langle \forall i \ : 0 \leq i < n : \ a.i > 0 \rangle \wedge 0 \leq n \leq N \wedge \neg r$

$$
\begin{array}{ll}
 & r = \langle \forall i \ : \underline{0 \leq i < N} : \ a.i > 0 \rangle \\
\equiv & \{ \text{ Arit., suposición } 0 \leq n \leq N \ \} \\
 & r = \langle \forall i \ : 0 \leq i < n \ \vee \ n \leq i < N : \ a.i > 0 \rangle \\
\equiv & \{ \text{ Partición de Rango } \} \\
 & r = \underline{\langle \forall i \ : 0 \leq i < n : \ a.i > 0 \rangle} \wedge \langle \forall i \ : n \leq i < N : \ a.i > 0 \rangle \\
\equiv & \{ \text{ Sup } \} \\
 & \underline{r} = \underline{r} \wedge \langle \forall i \ : n \leq i < N : \ a.i > 0 \rangle \\
\equiv & \{ \text{ Doble negación } \} \\
 & \neg(\underline{\neg r}) = \neg(\underline{\neg r}) \wedge \langle \forall i \ : n \leq i < N : \ a.i > 0 \rangle \\
\equiv & \{ \text{ Sup } \neg r \ \} \\
 & \neg \mathit{True} = \neg \mathit{True} \wedge \langle \forall i \ : n \leq i < N : \ a.i > 0 \rangle
\end{array}
$$

3. Finalización (demo)

Sup $r = \langle \forall i \ : 0 \leq i < n : \ a.i > 0 \rangle \wedge 0 \leq n \leq N \wedge \neg r$

$$
\begin{array}{ll}
 & r = \langle \forall i \ : \underline{0 \leq i < N} : \ a.i > 0 \rangle \\
\equiv & \{ \text{ Arit., suposición } 0 \leq n \leq N \ \} \\
 & r = \langle \forall i \ : 0 \leq i < n \vee n \leq i < N : \ a.i > 0 \rangle \\
\equiv & \{ \text{ Partición de Rango } \} \\
 & r = \underline{\langle \forall i \ : 0 \leq i < n : \ a.i > 0 \rangle} \wedge \langle \forall i \ : n \leq i < N : \ a.i > 0 \rangle \\
\equiv & \{ \text{ Sup } \} \\
 & \underline{r} = \underline{r} \wedge \langle \forall i \ : n \leq i < N : \ a.i > 0 \rangle \\
\equiv & \{ \text{ Doble negación } \} \\
 & \neg(\underline{\neg r}) = \neg(\underline{\neg r}) \wedge \langle \forall i \ : n \leq i < N : \ a.i > 0 \rangle \\
\equiv & \{ \text{ Sup } \neg r \ \} \\
 & \neg \mathit{True} = \neg \mathit{True} \wedge \langle \forall i \ : n \leq i < N : \ a.i > 0 \rangle \\
\equiv & \{ \text{ Def } \mathit{False} \ \} \\
 & \mathit{False} = \mathit{False} \wedge \langle \forall i \ : n \leq i < N : \ a.i > 0 \rangle
\end{array}
$$

3. Finalización (demo)

Sup $r = \langle \forall i \ : 0 \leq i < n : \ a.i > 0 \rangle \wedge 0 \leq n \leq N \wedge \neg r$

$$
\begin{array}{cl}
 & r = \langle \forall i \ : \underline{0 \leq i < N} : \ a.i > 0 \rangle \\
\equiv & \{ \text{ Arit., suposición } 0 \leq n \leq N \ \} \\
 & r = \langle \forall i \ : 0 \leq i < n \vee n \leq i < N : \ a.i > 0 \rangle \\
\equiv & \{ \text{ Partición de Rango } \} \\
 & r = \underline{\langle \forall i \ : 0 \leq i < n : \ a.i > 0 \rangle} \wedge \langle \forall i \ : n \leq i < N : \ a.i > 0 \rangle \\
\equiv & \{ \text{ Sup } \} \\
 & \underline{r} = \underline{r} \wedge \langle \forall i \ : n \leq i < N : \ a.i > 0 \rangle \\
\equiv & \{ \text{ Doble negación } \} \\
 & \neg(\underline{\neg r}) = \neg(\underline{\neg r}) \wedge \langle \forall i \ : n \leq i < N : \ a.i > 0 \rangle \\
\equiv & \{ \text{ Sup } \neg r \ \} \\
 & \neg \mathit{True} = \neg \mathit{True} \wedge \langle \forall i \ : n \leq i < N : \ a.i > 0 \rangle \\
\equiv & \{ \text{ Def } \mathit{False} \ \} \\
 & \mathit{False} = \mathit{False} \wedge \langle \forall i \ : n \leq i < N : \ a.i > 0 \rangle \\
\equiv & \{ \text{ Absorbente } \wedge \ \} \\
 & \mathit{False} = \mathit{False}
\end{array}
$$

3. Finalización (demo)

Sup $r = \langle \forall i : 0 \leq i < n : a.i > 0 \rangle \wedge 0 \leq n \leq N \wedge \neg r$

$$
\begin{array}{ll}
 & r = \langle \forall i : \underline{0 \leq i < N} : a.i > 0 \rangle \\
\equiv & \{ \text{ Arit., suposición } 0 \leq n \leq N \ \} \\
 & r = \langle \forall i : 0 \leq i < n \vee n \leq i < N : a.i > 0 \rangle \\
\equiv & \{ \text{ Partición de Rango } \} \\
 & r = \underline{\langle \forall i : 0 \leq i < n : a.i > 0 \rangle} \wedge \langle \forall i : n \leq i < N : a.i > 0 \rangle \\
\equiv & \{ \text{ Sup } \} \\
 & \underline{r} = \underline{r} \wedge \langle \forall i : n \leq i < N : a.i > 0 \rangle \\
\equiv & \{ \text{ Doble negación } \} \\
 & \neg(\underline{\neg r}) = \neg(\underline{\neg r}) \wedge \langle \forall i : n \leq i < N : a.i > 0 \rangle \\
\equiv & \{ \text{ Sup } \neg r \ \} \\
 & \neg \mathit{True} = \neg \mathit{True} \wedge \langle \forall i : n \leq i < N : a.i > 0 \rangle \\
\equiv & \{ \text{ Def } \mathit{False} \ \} \\
 & \mathit{False} = \mathit{False} \wedge \langle \forall i : n \leq i < N : a.i > 0 \rangle \\
\equiv & \{ \text{ Absorbente } \wedge \ \} \\
 & \mathit{False} = \mathit{False} \\
\equiv & \{ \text{ Equivalencia } \} \\
 & \mathit{True}
\end{array}
$$

3. Finalización demostrada

Const $N : Int; a : array[0, N) of\ Int;$
Var $r : Bool; n : Int;$
$\{P : N \geq 0\}$
$r, n := True, 0;$ $\heartsuit$
$\{I : r = \langle \forall i\ : 0 \leq i < n :\ a.i > 0 \rangle \wedge 0 \leq n \leq N\}$
do $n \neq N \wedge r \rightarrow$
$\{I \wedge B'\}$
$r, n := r \wedge a.n > 0, n + 1$ $\heartsuit$
$\{I\}$
od
$\{I \wedge \neg B'\}$ $\heartsuit$
$\{Q : r = \langle \forall i\ : 0 \leq i < N :\ a.i > 0 \rangle\}$

Programa final

Const $N : Int; a : array[0, N) of\ Int;$
Var $r : Bool; n : Int;$
$\{P : N \geq 0\}$
$r, n := True, 0;$
do $n \neq N \wedge r \rightarrow$
$r, n := r \wedge a.n > 0, n + 1$
od
$\{Q : r = \langle \forall i : 0 \leq i < N : a.i > 0 \rangle\}$