NFP108 – Introduction à la logique de Hoare

P. Courtieu

21 décembre 2018

Programmes sans effet de bord (purs)

- pas d'effet de bord = pas d'effet sur les variables
- exemple :

```
int max (int x, int y) {
  if (x < y) return y;
  else return x;
}
```

- programme = fonction = objet mathématique bien compris

$$\max : \left| \begin{array}{rcl} \mathbb{N} \times \mathbb{N} & \to & \mathbb{N} \\ (x, y) & \mapsto & \begin{cases} y & \text{si } x < y \\ x & \text{sinon} \end{cases} \end{array} \right.$$

Programmes purs récursifs

- récursif = on peut appeler le programme lui-même
- permet de faire des “boucles” sans effet de bord

```
int somme (int x) {
  if (0 < x) return x + somme(x-1);
  else return 0;
}
```

version impérative :

```
while(x>0){
  res = res + x;
  x = x - 1;
}
```

$$\text{somme} : \left| \begin{array}{ccc} \mathbb{N} & \to & \mathbb{N} \\ x & \mapsto & \begin{cases} x + \text{somme}(x-1) & \text{si } 0 \leq x \\ 0 & \text{sinon} \end{cases} \end{array} \right.$$

- objet mathématique bien compris : suite récurrente

$$U_i \begin{cases} U_0 & = 0 \\ U_{x+1} & = (x+1) + U_x \end{cases}$$

Programmes sans effet de bord (spécif. somme)

- spécification d'un programme pur = propriété d'une fonction
- = relation entre paramètres et résultat

$$\forall x_1 ... \forall x_n, \quad \mathit{Precond}(x_1, ..., x_n) \quad \rightarrow \quad \mathit{Correct}(x_1, ..., x_n, \mathrm{prog}(x_1, ..., x_n))$$

- exemple (`int somme (int x)`) :

$$\forall x, \qquad x \geq 0 \qquad \rightarrow \qquad \mathit{somme}(x) = \sum_{i=0}^{x} i$$

Récurrence

Récurrence

$$\forall P, \left(\begin{array}{l} P(0) \quad \wedge \\ \forall n \in \mathbb{N}, P(n) \rightarrow P(n+1) \end{array} \right) \rightarrow \forall n \in \mathbb{N}, P(n)$$

Récurrence

Récurrence

$$\forall P, \left(\begin{array}{l} P(0) \wedge \\ \forall n \in \mathbb{N}, P(n) \rightarrow P(n+1) \end{array} \right) \rightarrow \forall n \in \mathbb{N}, P(n)$$

cas de base

Récurrence

Récurrence

$$\forall P, \left(\begin{array}{l} P(0) \ \wedge \\ \forall n \in \mathbb{N}, P(n) \rightarrow P(n+1) \end{array} \right) \rightarrow \forall n \in \mathbb{N}, P(n)$$

cas de récurrence

Récurrence

Récurrence

$$\forall P, \left(\begin{array}{l} P(0) \ \wedge \\ \forall n \in \mathbb{N}, P(n) \rightarrow P(n+1) \end{array} \right) \rightarrow \forall n \in \mathbb{N}, P(n)$$

y compris $n = 0$

Récurrence

Récurrence (Variante 1)

$$\forall P \forall k, \left(\begin{array}{l} \forall n \leq k, P(n) \quad \wedge \\ \forall n \geq k, P(n) \rightarrow P(n+1) \end{array} \right) \rightarrow \forall n \in \mathbb{N}, P(n)$$

Récurrence

Récurrence (Variante 1)

$$\forall P \forall k, \left(\begin{array}{l} \text{cas de base} \\ \forall n \leq k, P(n) \wedge \\ \forall n \geq k, P(n) \rightarrow P(n+1) \end{array} \right) \rightarrow \forall n \in \mathbb{N}, P(n)$$

Récurrence

Récurrence (Variante 1)

$$\forall P \forall k, \left(\begin{array}{l} \forall n \leq k, P(n) \ \ \wedge \\ \forall n \geq k, P(n) \rightarrow P(n+1) \end{array} \right) \rightarrow \forall n \in \mathbb{N}, P(n)$$

cas de récurrence

Récurrence

Récurrence (Variante 1)

$$\forall P \forall k, \left(\begin{array}{l} \forall n \leq k, P(n) \quad \wedge \\ \forall n \geq k, P(n) \rightarrow P(n+1) \end{array} \right) \rightarrow \forall n \in \mathbb{N}, P(n)$$

y compris $n = k$

Récurrence

Récurrence (Variante 2)

$$\forall P, \left(\begin{array}{l} P(0) \wedge P(1) \wedge \\ \forall n \in \mathbb{N}, P(n) \rightarrow P(n+2) \end{array} \right) \rightarrow \forall n \in \mathbb{N}, P(n)$$

Récurrence

Récurrence (Variante 2)

$$\forall P, \left(\begin{array}{l} \forall n < 2, P(n) \wedge \\ \forall n \in \mathbb{N}, P(n) \rightarrow P(n+2) \end{array} \right) \rightarrow \forall n \in \mathbb{N}, P(n)$$

Récurrence

Récurrence (Variante 2)

$$\forall P \forall k, \left(\begin{array}{l} \forall n < k, P(n) \wedge \\ \forall n \in \mathbb{N}, P(n) \rightarrow P(n+k) \end{array} \right) \rightarrow \forall n \in \mathbb{N}, P(n)$$

Récurrence

Récurrence (Variante 3 (forte))

$$\forall P, \left(\forall m \in \mathbb{N}, \left(\left(\forall k < m, P(k) \right) \rightarrow P(m) \right) \right) \rightarrow \forall n \in \mathbb{N}, P(n)$$

Récurrence

Récurrence (Variante 3 (forte))

$$\forall P, \left(\forall m \in \mathbb{N}, \left(\left(\forall k < m, P(k) \right) \rightarrow P(m) \right) \right) \rightarrow \forall n \in \mathbb{N}, P(n)$$

Récurrence

Récurrence (Variante 3 (forte))

$$\forall P, \left(\forall m \in \mathbb{N}, \left(\left(\forall k < m, P(k) \right) \rightarrow P(m) \right) \right) \rightarrow \forall n \in \mathbb{N}, P(n)$$

Récurrence

Récurrence (Variante 3 (forte))

$$\forall P, \left(\forall m \in \mathbb{N}, \left(\left(\forall k < m, P(k) \right) \rightarrow P(m) \right) \right) \rightarrow \forall n \in \mathbb{N}, P(n)$$

Récurrence

Récurrence (Variante 3 (forte))

$$\forall P, \left(\forall m \in \mathbb{N}, \left(\left(\forall k < m, P(k) \right) \rightarrow P(m) \right) \right) \rightarrow \forall n \in \mathbb{N}, P(n)$$

- exactement équivalent aux autres variantes
- souvent plus pratique

Démonstration par récurrence de $\forall n \in \mathbb{N}, \texttt{somme}(n) = \sum_{i=0}^{n} i$

Récurrence (Variante 3 (forte))

$$\forall P, \left(\forall m \in \mathbb{N}, \left(\left(\forall k < m, P(k) \right) \rightarrow P(m) \right) \right) \rightarrow \forall n \in \mathbb{N}, P(n)$$

Fonction somme

$$\texttt{somme} : \left| \begin{array}{rcl} x & \mapsto & \left\{ \begin{array}{ll} x + \texttt{somme}(x-1) & \text{si } 0 \leq x \\ 0 & \text{sinon} \end{array} \right. \end{array} \right.$$

Démonstration par récurrence de $\forall n \in \mathbb{N}, \texttt{somme}(n) = \sum_{i=0}^{n} i$

Récurrence (Variante 3 (forte))

$$\forall P, \left(\forall m \in \mathbb{N}, \left(\left(\forall k < m, P(k) \right) \rightarrow P(m) \right) \right) \rightarrow \forall n \in \mathbb{N}, P(n)$$

Fonction somme

$$\texttt{somme} : \left| \begin{array}{ccc} x & \mapsto & \left\{ \begin{array}{ll} x + \texttt{somme}(x-1) & \text{si } 0 \leq x \\ 0 & \text{sinon} \end{array} \right. \end{array} \right.$$

Démonstration par récurrence de $\forall n \in \mathbb{N}, \texttt{somme}(n) = \sum_{i=0}^{n} i$

Récurrence (Variante 3 (forte))

$$\forall P, \left(\forall m \in \mathbb{N}, \left(\left(\forall k < m, P(k) \right) \rightarrow P(m) \right) \right) \rightarrow \forall n \in \mathbb{N}, P(n)$$

Fonction somme

$$\texttt{somme} : \left| \begin{array}{rcl} x & \mapsto & \left\{ \begin{array}{ll} x + \texttt{somme}(x-1) & \text{si } 0 \leq x \\ 0 & \text{sinon} \end{array} \right. \end{array} \right.$$

Soit $m \in \mathbb{N}$.

Démonstration par récurrence de $\forall n \in \mathbb{N}, \texttt{somme}(n) = \sum_{i=0}^{n} i$

Récurrence (Variante 3 (forte))

$$\forall P, \left(\forall m \in \mathbb{N}, \left(\left(\forall k < m, P(k) \right) \rightarrow P(m) \right) \right) \rightarrow \forall n \in \mathbb{N}, P(n)$$

Fonction somme

$$\texttt{somme} : \left| \begin{array}{rcl} x & \mapsto & \left\{ \begin{array}{ll} x + \texttt{somme}(x-1) & \text{si } 0 \leq x \\ 0 & \text{sinon} \end{array} \right. \end{array} \right.$$

Soit $m \in \mathbb{N}$. Supposons $\forall k < m, \texttt{somme}(k) = \sum_{i=0}^{k} i$.

Démonstration par récurrence de $\forall n \in \mathbb{N}, \text{somme}(n) = \sum_{i=0}^{n} i$

Récurrence (Variante 3 (forte))

$$\forall P, \left(\forall m \in \mathbb{N}, \left(\left(\forall k < m, P(k) \right) \rightarrow P(m) \right) \right) \rightarrow \forall n \in \mathbb{N}, P(n)$$

Fonction somme

$$\text{somme} : \left| \begin{array}{rcl} x & \mapsto & \left\{ \begin{array}{ll} x + \text{somme}(x-1) & \text{si } 0 \leq x \\ 0 & \text{sinon} \end{array} \right. \end{array} \right.$$

Soit $m \in \mathbb{N}$. Supposons $\forall k < m, \text{somme}(k) = \sum_{i=0}^{k} i$.

Montrons que $\text{somme}(m) = \sum_{i=0}^{m} i$.

Démonstration par récurrence de $\forall n \in \mathbb{N}, \text{somme}(n) = \sum_{i=0}^{n} i$

Récurrence (Variante 3 (forte))

$$\forall P, \left(\forall m \in \mathbb{N}, \left(\left(\forall k < m, P(k) \right) \rightarrow P(m) \right) \right) \rightarrow \forall n \in \mathbb{N}, P(n)$$

Fonction somme

$$\text{somme} : \left| \begin{array}{rcl} x & \mapsto & \begin{cases} x + \text{somme}(x-1) & \text{si } 0 \leq x \\ 0 & \text{sinon} \end{cases} \end{array} \right.$$

Soit $m \in \mathbb{N}$. Supposons $\forall k < m, \text{somme}(k) = \sum_{i=0}^{k} i$.

Montrons que $\text{somme}(m) = \sum_{i=0}^{m} i$. Déf. de somme, 2 cas :

(1) $m = 0$,

(2) $m > 0$,

Démonstration par récurrence de $\forall n \in \mathbb{N}, \text{somme}(n) = \sum_{i=0}^{n} i$

Récurrence (Variante 3 (forte))

$$\forall P, \left(\forall m \in \mathbb{N}, \left(\left(\forall k < m, P(k) \right) \rightarrow P(m) \right) \right) \rightarrow \forall n \in \mathbb{N}, P(n)$$

Fonction somme

$$\text{somme} : \left| \begin{array}{ccc} x & \mapsto & \left\{ \begin{array}{ll} x + \text{somme}(x-1) & \text{si } 0 \leq x \\ 0 & \text{sinon} \end{array} \right. \end{array} \right.$$

Soit $m \in \mathbb{N}$. Supposons $\forall k < m, \text{somme}(k) = \sum_{i=0}^{k} i$.

Montrons que $\text{somme}(m) = \sum_{i=0}^{m} i$. Déf. de somme, 2 cas :

(1) $m = 0$, alors $\text{somme}(0) = 0 = \sum_{i=0}^{0} i$, OK.

(2) $m > 0$,

Démonstration par récurrence de $\forall n \in \mathbb{N}, \text{somme}(n) = \sum_{i=0}^{n} i$

Récurrence (Variante 3 (forte))

$$\forall P, \left(\forall m \in \mathbb{N}, \left(\left(\forall k < m, P(k) \right) \rightarrow P(m) \right) \right) \rightarrow \forall n \in \mathbb{N}, P(n)$$

Fonction somme

$$\text{somme} : \left| \begin{array}{ccc} x & \mapsto & \left\{ \begin{array}{ll} x + \text{somme}(x-1) & \text{si } 0 \leq x \\ 0 & \text{sinon} \end{array} \right. \end{array} \right.$$

Soit $m \in \mathbb{N}$. Supposons $\forall k < m, \text{somme}(k) = \sum_{i=0}^{k} i$.

Montrons que $\text{somme}(m) = \sum_{i=0}^{m} i$. Déf. de somme, 2 cas :

(1) $m = 0$, alors $\text{somme}(0) = 0 = \sum_{i=0}^{0} i$, OK.

(2) $m > 0$, alors $\text{somme}(m) = m + somme(m-1)$

Démonstration par récurrence de $\forall n \in \mathbb{N}, \text{somme}(n) = \sum_{i=0}^{n} i$

Récurrence (Variante 3 (forte))

$$\forall P, \left(\forall m \in \mathbb{N}, \left(\left(\forall k < m, P(k) \right) \rightarrow P(m) \right) \right) \rightarrow \forall n \in \mathbb{N}, P(n)$$

Fonction somme

$$\text{somme} : \left| \begin{array}{rcl} x & \mapsto & \begin{cases} x + \text{somme}(x-1) & \text{si } 0 \leq x \\ 0 & \text{sinon} \end{cases} \end{array} \right.$$

Soit $m \in \mathbb{N}$. Supposons $\forall k < m, \text{somme}(k) = \sum_{i=0}^{k} i$.

Montrons que $\text{somme}(m) = \sum_{i=0}^{m} i$. Déf. de somme, 2 cas :

(1) $m = 0$, alors $\text{somme}(0) = 0 = \sum_{i=0}^{0} i$, OK.

(2) $m > 0$, alors $\text{somme}(m) = m + somme(m-1) = m + \sum_{i=0}^{m-1} i$

Hyp. de réc.

Démonstration par récurrence de $\forall n \in \mathbb{N}, \text{somme}(n) = \sum_{i=0}^{n} i$

Récurrence (Variante 3 (forte))

$$\forall P, \left(\forall m \in \mathbb{N}, \left(\left(\forall k < m, P(k) \right) \rightarrow P(m) \right) \right) \rightarrow \forall n \in \mathbb{N}, P(n)$$

Fonction somme

$$\text{somme} : \left| \begin{array}{rcl} x & \mapsto & \left\{ \begin{array}{ll} x + \text{somme}(x-1) & \text{si } 0 \leq x \\ 0 & \text{sinon} \end{array} \right. \end{array} \right.$$

Soit $m \in \mathbb{N}$. Supposons $\forall k < m, \text{somme}(k) = \sum_{i=0}^{k} i$.

Montrons que $\text{somme}(m) = \sum_{i=0}^{m} i$. Déf. de somme, 2 cas :

(1) $m = 0$, alors $\text{somme}(0) = 0 = \sum_{i=0}^{0} i$, OK.

(2) $m > 0$, alors $\text{somme}(m) = m + somme(m-1) = m + \sum_{i=0}^{m-1} i = \sum_{i=0}^{m} i$. OK.

Démonstration par récurrence de $\forall n \in \mathbb{N}, \text{somme}(n) = \sum_{i=0}^{n} i$

Récurrence (Variante 3 (forte))

$$\forall P, \left(\forall m \in \mathbb{N}, \left(\left(\forall k < m, P(k) \right) \rightarrow P(m) \right) \right) \rightarrow \forall n \in \mathbb{N}, P(n)$$

Fonction somme

$$\text{somme} : \left| \begin{array}{rcl} x & \mapsto & \begin{cases} x + \text{somme}(x-1) & \text{si } 0 \leq x \\ 0 & \text{sinon} \end{cases} \end{array} \right.$$

Soit $m \in \mathbb{N}$. Supposons $\forall k < m, \text{somme}(k) = \sum_{i=0}^{k} i$.

Montrons que $\text{somme}(m) = \sum_{i=0}^{m} i$. Déf. de somme, 2 cas :

(1) $m = 0$, alors $\text{somme}(0) = 0 = \sum_{i=0}^{0} i$, OK.

(2) $m > 0$, alors $\text{somme}(m) = m + somme(m-1) = m + \sum_{i=0}^{m-1} i = \sum_{i=0}^{m} i$. OK.

CQFD

Programmes récursifs sans effet de bord

Spécification :

- Souvent dans un commentaire
- variables quantifiées implicitement ($\forall x$)

```
/* Précondition: x ≥ 0 */
/* Postcondition: resultat = ∑_{i=0}^{x} i */
int somme (int x) {
  if (0 <= x) return x+somme(x-1);
  else return 0;
}
```

Programmes récursifs sans effet de bord

Spécification :

- Souvent dans un commentaire
- variables quantifiées implicitement ($\forall x$)

```
/* Requires: x ≥ 0 */
/* Ensures: resultat = ∑_{i=0}^{x} i */
int somme (int x) {
  if (0 <= x) return x+somme(x-1);
  else return 0;
}
```

Justification :

Programmes récursifs sans effet de bord

Spécification :

- Souvent dans un commentaire
- variables quantifiées implicitement ($\forall x$)

```
/* Précondition: x ≥ 0 */
/* Postcondition: resultat = ∑_{i=0}^{x} i */
int somme (int x) {
  if (0 <= x) return x+somme(x-1);
  else return 0;
}
```

Correct

Justification :

- case de base OK

Programmes récursifs sans effet de bord

Spécification :

- Souvent dans un commentaire
- variables quantifiées implicitement ($\forall x$)

```
/* Précondition: x ≥ 0 */
/* Postcondition: resultat = ∑_{i=0}^{x} i */
int somme (int x) {
  if (0 <= x) return x+somme(x-1);
  else return 0;
}
```

Correct si `somme(x-1)` correct

Justification :

- case de base OK
- autres cas OK si appels récursifs OK (récurrence).

Programmes récursifs sans effet de bord

Spécification :

- Souvent dans un commentaire
- variables quantifiées implicitement ($\forall x$)

```
/* Précondition: x ≥ 0 */
/* Postcondition: resultat = ∑_{i=0}^{x} i */
int somme (int x) {
  if (0 <= x) return x+somme(x-1);
  else return 0;
}
```

Justification :

- case de base OK
- autres cas OK si appels récursifs OK (récurrence).
- par récurrence OK.

Programmes récursifs sans effet de bord

Spécification :

- Souvent dans un commentaire
- variables quantifiées implicitement ($\forall x$)

```
/* Précondition: x ≥ 0 */
/* Postcondition: resultat = ∑_{i=0}^{x} i */
int somme (int x) {
  if (0 <= x) return x+somme(x-1);
  else return 0;
}
```

Justification :

- case de base OK
- autres cas OK si appels récursifs OK (récurrence).
- par récurrence OK.
- seulement quand ça termine !

Programmes récursifs sans effet de bord

- autre exemple

```
int div (int x, int y) {
  if (x < y) return 0;
  else return 1 + div (x-y) y;
}
```

- série récursive

$$\text{div} : \left| \begin{array}{rcl} \mathbb{N} \times \mathbb{N} & \rightarrow & \mathbb{N} \\ (x, y) & \mapsto & \begin{cases} 0 & \text{si } x < y \\ 1 + div(x - y, y) & \text{sinon} \end{cases} \end{array} \right.$$

Programmes sans effet de bord (spécif.div)

```
int div (int x, int y) {
  if (x < y) return 0;
  else return 1 + div (x-y) y;
}
```

$$\forall x_1 ... \forall x_n, \quad \mathit{Precond}(x_1, ..., x_n) \quad \rightarrow \quad \mathit{Correct}(x_1, ..., x_n, \mathrm{prog}(x_1, ..., x_n))$$

$$\forall x \forall y, \quad x \geq 0 \wedge y > 0 \quad \rightarrow \quad y \times \mathtt{div}(x, y) \leq x \wedge y \times (1 + \mathtt{div}(x, y))$$

Preuve de $\forall x \forall y, x \geq 0 \wedge y > 0 \rightarrow y \times \mathrm{div}(x,y) \leq x \wedge y \times (1 + \mathrm{div}(x,y)) > x$

$$\mathrm{div} : (x,y) \mapsto \left\{ \begin{array}{l} 0 \text{ si } x < y, \text{ et } 1 + div(x-y,y) \text{ sinon} \end{array} \right.$$

Preuve de $\forall x \forall y, x \geq 0 \wedge y > 0 \rightarrow y \times \text{div}(x,y) \leq x \wedge y \times (1 + \text{div}(x,y)) > x$

$\forall x \forall y,$ $P(x,y)$

$$\text{div} : (x,y) \mapsto \begin{cases} 0 \text{ si } x < y, \text{ et } 1 + div(x-y,y) \text{ sinon} \end{cases}$$

$$\forall P, \left(\forall y \forall m \in \mathbb{N}, \left(\left(\forall k < m, P(k,y) \right) \rightarrow P(m,y) \right) \right) \rightarrow \forall x \forall y \in \mathbb{N}, P(x,y)$$

Soit y et $m \in \mathbb{N}$, supposons $\forall k < m, P(k,y)$, et montrons qu'alors (P(m,y))

Preuve de $\forall x \forall y, x \geq 0 \wedge y > 0 \rightarrow y \times \text{div}(x,y) \leq x \wedge y \times (1 + \text{div}(x,y)) > x$

$$\text{div} : (x,y) \mapsto \begin{cases} 0 \text{ si } x < y, \\ \text{et } 1 + div(x-y,y) \text{ sinon} \end{cases}$$

$$\forall P, \left(\forall y \forall m \in \mathbb{N}, \left(\left(\forall k < m, P(k,y) \right) \rightarrow P(m,y) \right) \right) \rightarrow \forall x \forall y \in \mathbb{N}, P(x,y)$$

Soit y et $m \in \mathbb{N}$, supposons $\forall k < m, P(k,y)$, et montrons qu'alors (P(m,y))

- si $m < y$: cas de base
- sinon $m \geq y$: cas de récurrence
  (supp. appels récursifs corrects)

Preuve de $\forall x \forall y, x \geq 0 \wedge y > 0 \rightarrow y \times \underset{0}{\text{div}}(x,y) \leq x \wedge y \times (1 + \underset{0}{\text{div}}(x,y)) > x$

$$\text{div} : (x,y) \mapsto \begin{cases} 0 \text{ si } x < y, \\ \text{et } 1 + div(x-y,y) \text{ sinon} \end{cases}$$

$$\forall P, \left(\forall y \forall m \in \mathbb{N}, \left(\left(\forall k < m, P(k,y) \right) \rightarrow P(m,y) \right) \right) \rightarrow \forall x \forall y \in \mathbb{N}, P(x,y)$$

Soit y et $m \in \mathbb{N}$, supposons $\forall k < m, P(k,y)$, et montrons qu'alors (P(m,y))

- si $m < y$: alors par définition de div : div(m,y)=0
- sinon $m \geq y$:

Preuve de $\forall x \forall y, x \geq 0 \wedge y > 0 \rightarrow y \times \mathrm{div}(x,y) \leq x \wedge y \times (1 + \mathrm{div}(x,y)) > x$

0 0

$$\mathrm{div} : (x,y) \mapsto \left\{ \begin{array}{l} 0 \text{ si } x < y, \text{ et } 1 + div(x-y,y) \text{ sinon} \end{array} \right.$$

$$\forall P, \left(\forall y \forall m \in \mathbb{N}, \left(\left(\forall k < m, P(k,y) \right) \rightarrow P(m,y) \right) \right) \rightarrow \forall x \forall y \in \mathbb{N}, P(x,y)$$

Soit y et $m \in \mathbb{N}$, supposons $\forall k < m, P(k,y)$, et montrons qu'alors (P(m,y))

- si $m < y$: alors par définition de div : div(m,y)=0
  et par hypothèse : $m \geq 0$,
- sinon $m \geq y$:

Preuve de $\forall x \forall y, x \geq 0 \wedge y > 0 \rightarrow y \times \underset{0}{\text{div}}(x,y) \leq x \wedge y \times (1 + \underset{0}{\text{div}}(x,y)) > x$

$$\text{div} : (x,y) \mapsto \begin{cases} 0 \text{ si } x < y, \text{ et } 1 + div(x-y,y) \text{ sinon} \end{cases}$$

$$\forall P, \left(\forall y \forall m \in \mathbb{N}, \left(\left(\forall k < m, P(k,y) \right) \rightarrow P(m,y) \right) \right) \rightarrow \forall x \forall y \in \mathbb{N}, P(x,y)$$

Soit y et $m \in \mathbb{N}$, supposons $\forall k < m, P(k,y)$, et montrons qu'alors (P(m,y))

- si $m < y$: alors par définition de div : div(m,y)=0
  et par hypothèse : $m \geq 0$, donc : $y \times 0 \leq m$ et $y \times 1 > m$. OK.
- sinon $m \geq y$:

Preuve de $\forall x \forall y, x \geq 0 \wedge y > 0 \rightarrow y \times \text{div}(x,y) \leq x \wedge y \times (1 + \text{div}(x,y)) > x$

$$\text{div} : (x,y) \mapsto \begin{cases} 0 \text{ si } x < y, \text{ et } 1 + div(x-y,y) \text{ sinon} \end{cases}$$

$$\forall P, \left(\forall y \forall m \in \mathbb{N}, \left(\left(\forall k < m, P(k,y) \right) \rightarrow P(m,y) \right) \right) \rightarrow \forall x \forall y \in \mathbb{N}, P(x,y)$$

Soit y et $m \in \mathbb{N}$, supposons $\forall k < m, P(k,y)$, et montrons qu'alors (P(m,y))

- si $m < y$: OK.
- sinon $m \geq y$:

Preuve de $\forall x \forall y, x \geq 0 \wedge y > 0 \rightarrow y \times \text{div}(x,y) \leq x \wedge y \times (1 + \text{div}(x,y)) > x$

$$\text{div} : (x,y) \mapsto \begin{cases} 0 \text{ si } x < y, \text{ et } 1 + div(x-y,y) \text{ sinon} \end{cases}$$

$$\forall P, \left(\forall y \forall m \in \mathbb{N}, \left(\left(\forall k < m, P(k,y) \right) \rightarrow P(m,y) \right) \right) \rightarrow \forall x \forall y \in \mathbb{N}, P(x,y)$$

Soit y et $m \in \mathbb{N}$, supposons $\forall k < m, P(k,y)$, et montrons qu'alors (P(m,y))

- si $m < y$: OK.
- sinon $m \geq y$: alors $div(m,y) = 1 + div(m-y,y)$.

Preuve de $\forall x \forall y, x \geq 0 \wedge y > 0 \rightarrow y \times \text{div}(x,y) \leq x \wedge y \times (1 + \text{div}(x,y)) > x$

$$\text{div} : (x,y) \mapsto \begin{cases} 0 \text{ si } x < y, \text{ et } 1 + div(x-y,y) \text{ sinon} \end{cases}$$

$$\forall P, \left(\forall y \forall m \in \mathbb{N}, \left(\left(\forall k < m, P(k,y) \right) \rightarrow P(m,y) \right) \right) \rightarrow \forall x \forall y \in \mathbb{N}, P(x,y)$$

Soit y et $m \in \mathbb{N}$, supposons $\forall k < m, P(k,y)$, et montrons qu'alors (P(m,y))

- si $m < y$: OK.
- sinon $m \geq y$: alors $div(m,y) = 1 + div(m-y,y)$.
  Or $0 \leq m - y < m$,

Preuve de $\forall x \forall y, x \geq 0 \wedge y > 0 \rightarrow y \times \text{div}(x,y) \leq x \wedge y \times (1 + \text{div}(x,y)) > x$

$$\text{div} : (x,y) \mapsto \begin{cases} 0 \text{ si } x < y, \text{ et } 1 + div(x-y,y) \text{ sinon} \end{cases}$$

$$\forall P, \Big(\forall y \forall m \in \mathbb{N}, \Big(\big(\forall k < m, P(k,y) \big) \rightarrow P(m,y) \Big) \Big) \rightarrow \forall x \forall y \in \mathbb{N}, P(x,y)$$

Soit y et $m \in \mathbb{N}$, supposons $\forall k < m, P(k,y)$, et montrons qu'alors (P(m,y))

- si $m < y$: OK.
- sinon $m \geq y$: alors $div(m,y) = 1 + div(m-y,y)$.
  Or $0 \leq m - y < m$, donc $\texttt{div}(m-y,y)$ vérifie la propriété :

$$y \times \text{div}(m-y,y) \leq m-y \wedge y \times (1 + \text{div}(m-y,y)) > m-y$$

Preuve de $\forall x \forall y, x \geq 0 \wedge y > 0 \rightarrow y \times \text{div}(x,y) \leq x \wedge y \times (1 + \text{div}(x,y)) > x$

$$\text{div} : (x,y) \mapsto \left\{ \begin{array}{l} 0 \text{ si } x < y, \\ \text{et } 1 + div(x-y,y) \text{ sinon} \end{array} \right.$$

$$\forall P, \left(\forall y \forall m \in \mathbb{N}, \left(\left(\forall k < m, P(k,y) \right) \rightarrow P(m,y) \right) \right) \rightarrow \forall x \forall y \in \mathbb{N}, P(x,y)$$

Soit y et $m \in \mathbb{N}$, supposons $\forall k < m, P(k,y)$, et montrons qu'alors (P(m,y))

- si $m < y$: OK.
- sinon $m \geq y$: alors $div(m,y) = 1 + div(m-y,y)$.
  Or $0 \leq m - y < m$, donc $\texttt{div}(m-y,y)$ vérifie la propriété :

$$\begin{array}{rcccl}
y \times \text{div}(m-y,y) & \leq & m-y & \wedge & y \times (1 + \text{div}(m-y,y)) > m-y \\
y \times \text{div}(m-y,y) & \leq & m-y & \wedge & y \times (\text{div}(m,y)) > m-y \\
\quad + y & & + y & & \quad + y \quad + y \\
y \times \text{div}(m,y) & \leq & m & \wedge & y \times (1 + \text{div}(m,y)) > m
\end{array}$$

OK

Quid des programmes impératifs ?

```
int res = 0, i = 0;
while (i<=x) {
  res = res + i;
}
```

- programme avec effet de bord (principalement modif. des variables)
- objet mathématique moins ≪ propre ≫
- fait passer de l'état (des variables) q_1 à l'état q_2
- noté $q_1 \rightsquigarrow_{prog} q_2$
- propriété d'un programme ?

Spécification

- spécification= $Precond + Postcond$
- P : Pré-condition, Q : Post-condition.

Spécification

- spécification= $Precond + Postcond$
- P : Pré-condition, Q : Post-condition.
- Triplet de Hoare :

$$\{P\}programme\{Q\}$$

Exemple de triplet $\{P\}prog\{Q\}$

$\{x \geq 0 \wedge y > 0\}$

```
a := 0;
b := x;
while (b >= y) {
   b := b - y;
   a := a + 1
}
```

$\{a * y \leq x \ \wedge \ (1 + a) * y > x\}$

Spécifications possibles :

- $\{x \geq 0 \ \wedge \ y > 0\}\ P\ \{a * y \leq x \ \wedge \ (1 + a) * y > x\}$
- $\{x >= 0 \ \wedge \ y > 0\}\ P\ \{a * y + b = x \ \wedge \ 0 \leq b < y\}$

Exemple de triplet $\{P\}prog\{Q\}$

$\{x \geq 0 \wedge y > 0\}$

```
a := 0;
b := x;
while (b >= y) {
   b := b - y;
   a := a + 1
}
```

$\{a * y + b = x \ \wedge \ 0 \leq b < y\}$

Spécifications possibles :

- $\{x \geq 0 \ \wedge \ y > 0\}\ P\ \{a * y \leq x \ \wedge \ (1 + a) * y > x\}$
- $\{x >= 0 \ \wedge \ y > 0\}\ P\ \{a * y + b = x \ \wedge \ 0 \leq b < y\}$

Sémantique : 2 versions

On note $q_1 \rightsquigarrow_{prog} q_2 \equiv$ programme $prog$ transforme l'état q_1 en q_2

- triplet $\{P\}prog\{Q\}$ (partiellement) correct ssi :

$$\forall q_1 \forall q_2, P(q_1) \rightarrow (q_1 \rightsquigarrow_{prog} q_2) \rightarrow Q(q_2)$$

- correction totale :

$$\forall q_1, P(q_1) \rightarrow \exists q_2, (q_1 \rightsquigarrow_{prog} q_2) \rightarrow Q(q_2)$$

Comparaison avec la logique

- Rappel de logique :

  formule logique / sémantique / système de déduction (ex : DN)

$$\frac{\textit{formule} \ \ldots \ \textit{formule}}{\textit{formule}}$$

- De la même façon :

  triplet de hoare / sémantique / système de déduction

$$\frac{\textit{triplet} \ \ldots \ \textit{triplet}}{\textit{triplet}}$$

Cond(`if`) et Seq (;)

$$\textsc{Seq}\ \frac{\{P\}C_1\{Q\} \qquad \{Q\}C_2\{R\}}{\{P\}C_1\ ;C_2\{R\}}$$

$$\textsc{Cond}\ \frac{\{P \wedge B\}\ I_1\ \{Q\} \qquad \{P \wedge \neg B\}\ I_2\ \{Q\}}{\{P\}\ \mathit{if}\ B\ \mathit{then}\ I_1\ \mathit{else}\ I_2\ \{Q\}}$$

AFF(affectation)

$$E[x \leftarrow e] \quad P[x \leftarrow e]$$
$$\equiv$$

« expression E dans laquelle la variable x est substituée par l'expression e »
« formule E dans laquelle la variable x est substituée par l'expression e »

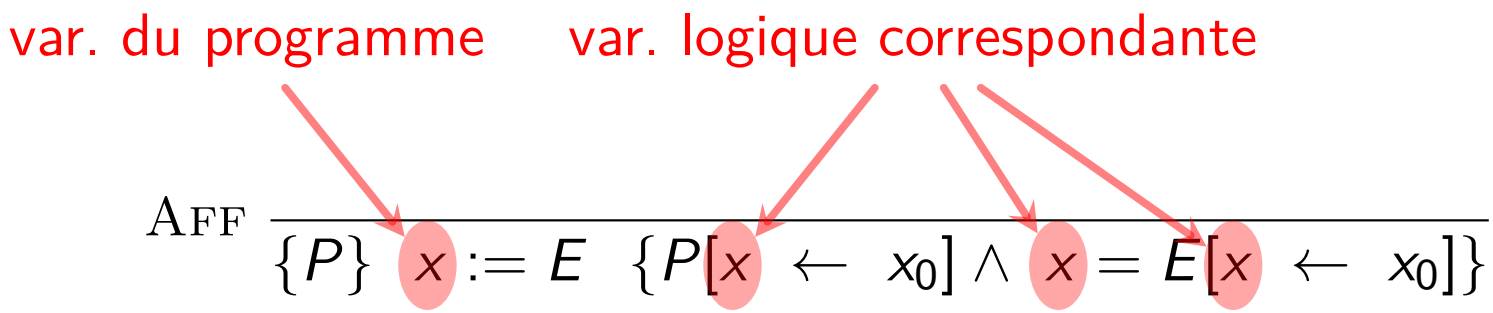
Axiome $$\text{AFF}\ \frac{}{\{P\}\ \ x := E\ \ \{P[x\ \leftarrow\ x_0] \wedge\ x = E[x\ \leftarrow\ x_0]\}}$$

Version moins intuitive $$\text{AFF1}\ \frac{}{\{P[x \leftarrow E]\}\ x := E\ \{P\}}$$.

AFF(affectation)

$$E[x \leftarrow e] \quad P[x \leftarrow e]$$
$$\equiv$$

« expression E dans laquelle la variable x est substituée par l'expression e »
« formule E dans laquelle la variable x est substituée par l'expression e »



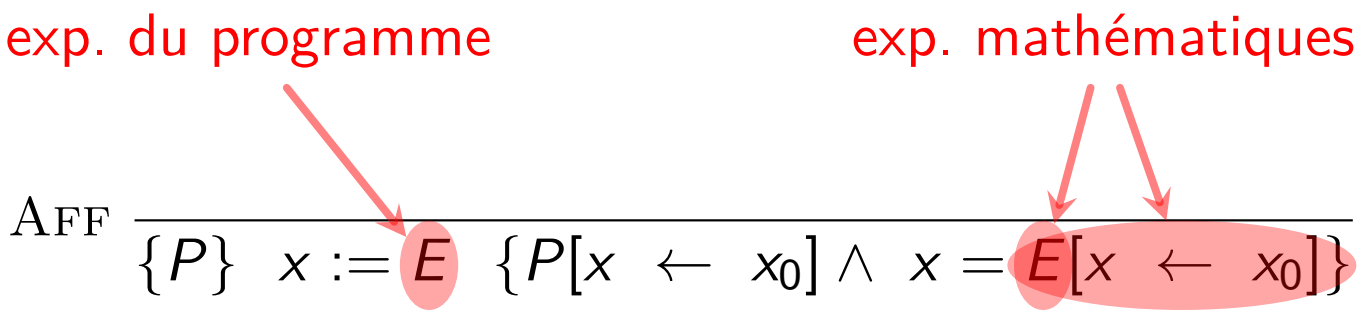
Axiome $\textsc{Aff}\ \dfrac{}{\{P\}\ x := E\ \{P[x \leftarrow x_0] \wedge x = E[x \leftarrow x_0]\}}$

Version moins intuitive $\textsc{Aff1}\ \dfrac{}{\{P[x \leftarrow E]\}\ x := E\ \{P\}}$.

AFF(affectation)

$$E[x \leftarrow e] \quad P[x \leftarrow e]$$
$$\equiv$$

« expression E dans laquelle la variable x est substituée par l'expression e »
« formule E dans laquelle la variable x est substituée par l'expression e »



Axiome $\textsc{Aff}\ \dfrac{}{\{P\}\ x := E\ \{P[x \leftarrow x_0] \wedge x = E[x \leftarrow x_0]\}}$

Version moins intuitive $\textsc{Aff1}\ \dfrac{}{\{P[x \leftarrow E]\}\ x := E\ \{P\}}$.

AFF(affectation)

$$E[x \leftarrow e] \quad P[x \leftarrow e]$$
$$\equiv$$

« expression E dans laquelle la variable x est substituée par l'expression e »
« formule E dans laquelle la variable x est substituée par l'expression e »

formules logiques (pré/postconditions)

Axiome

$$\textsc{Aff}\ \frac{}{\{P\}\ x := E\ \{P[x \leftarrow x_0] \wedge x = E[x \leftarrow x_0]\}}$$

Version moins intuitive

$$\textsc{Aff1}\ \frac{}{\{P[x \leftarrow E]\}\ x := E\ \{P\}}$$

.

AFF(affectation)

$$E[x \leftarrow e] \quad P[x \leftarrow e]$$
$$\equiv$$

« expression E dans laquelle la variable x est substituée par l'expression e »
« formule E dans laquelle la variable x est substituée par l'expression e »

variable logique « fraîche » (ancienne valeur de x)

Axiome

$$\textsc{Aff}\ \frac{}{\{P\}\ \ x := E\ \ \{P[x \leftarrow x_0] \wedge\ x = E[x \leftarrow x_0]\}}$$

Version moins intuitive

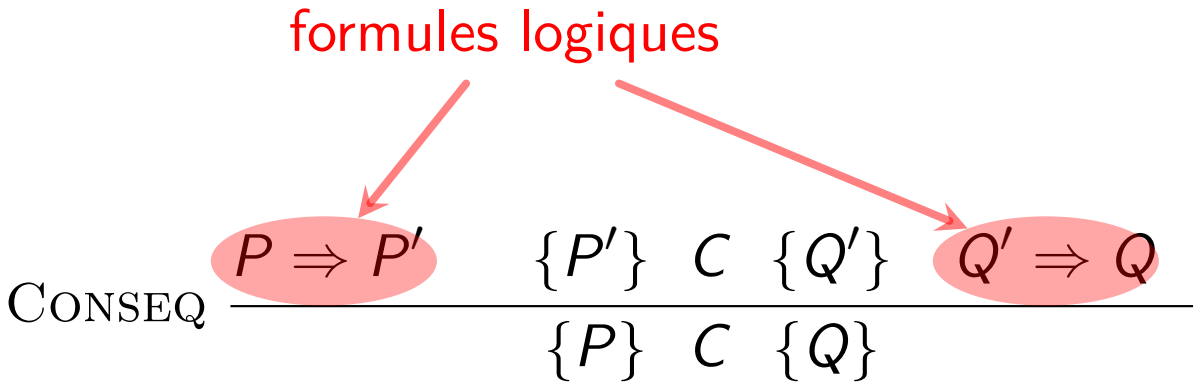
$$\textsc{Aff1}\ \frac{}{\{P[x \leftarrow E]\}\ x := E\ \{P\}}$$

.

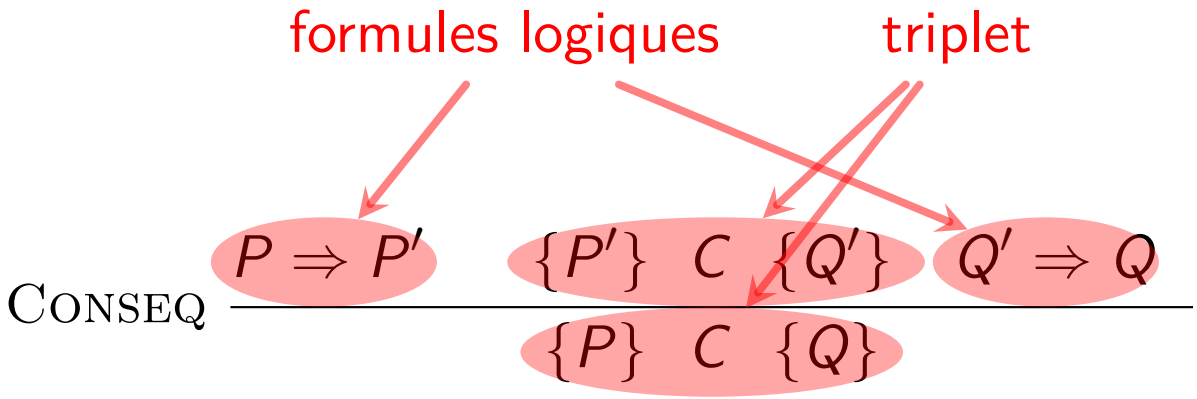
CONSEQ : permet la déduction logique

$$\textsc{Conseq}\ \frac{P \Rightarrow P' \qquad \{P'\}\ \ C\ \ \{Q'\} \qquad Q' \Rightarrow Q}{\{P\}\ \ C\ \ \{Q\}}$$

CONSEQ : permet la déduction logique



CONSEQ : permet la déduction logique



CONSEQ : permet la déduction logique

$$\textsc{Conseq}\ \frac{\dfrac{\vdots}{P \Rightarrow P'} \quad \{P'\}\ C\ \{Q'\} \quad \dfrac{\vdots}{Q' \Rightarrow Q}}{\{P\}\ C\ \{Q\}}$$

- Les formules se prouvent dans un système de déuction logique

CONSEQ : permet la déduction logique

$$\textsc{Conseq}\ \frac{\dfrac{\vdots}{P \Rightarrow P'} \qquad \dfrac{\vdots}{\{P'\}\ \ C\ \ \{Q'\}} \qquad \dfrac{\vdots}{Q' \Rightarrow Q}}{\{P\}\ \ C\ \ \{Q\}}$$

- Les formules se prouvent dans un système de déuction logique
- Les triplets se prouvent récursivement avec les mêmes règles

Règles du `while`

$$\textsc{While}\ \frac{\{I \wedge B\}\ C\ \{I\}}{\{I\}\ \mathit{while}\ \ B\ \mathit{do}\ \ C\ \mathit{done}\ \{I \wedge \neg B\}}$$

$$\textsc{WhileT}\ \frac{\langle P\ \wedge\ B\ \wedge\ (E = n)\rangle\ C\ \langle\ P\ \wedge\ E\ <\ n\ \wedge\ E\ \geq\ 0\ \rangle}{\langle P\rangle\ \mathit{while}\ B\ \mathit{do}\ C\ \mathit{done}\ \langle P \wedge \neg B\rangle}$$

Invariant

$$\textsc{While}\ \frac{\{I \wedge B\}\ C\ \{I\}}{\{I\}\ \mathit{while}\ \ B\ \mathit{do}\ \ C\ \mathit{done}\ \{I \wedge \neg B\}}$$

```
{x≥ 0∧ y>0}
a := 0;
b := x;

while (b >= y) {

  b := b - y;
  a := a + 1

}


{a×y+b=x∧0≤b<y}
```

Invariant

$$\textsc{While}\ \frac{\{I \wedge B\}\ C\ \{I\}}{\{I\}\ \mathit{while}\ \ B\ \mathit{do}\ \ C\ \mathit{done}\ \{I \wedge \neg B\}}$$

```
{x≥ 0∧ y>0}
a := 0;
b := x;

while (b >= y) {

  b := b - y;
  a := a + 1

}


{a×y+b=x∧0≤b<y}
```

Invariant

$$\textsc{While}\ \frac{\{I \wedge B\}\ C\ \{I\}}{\{I\}\ \mathit{while}\ \ B\ \mathit{do}\ \ C\ \mathit{done}\ \{I \wedge \neg B\}}$$

- invariant `I` ≪ sorti du chapeau ≫

```
{x≥ 0∧ y>0}
a := 0;
b := x;

while (b >= y) {
{I∧b≥y}
  b := b - y;
  a := a + 1
{I}
}


{a×y+b=x∧0≤b<y}
```

Invariant

$$\textsc{While}\ \frac{\{I \wedge B\}\ C\ \{I\}}{\{I\}\ \mathit{while}\ \ B\ \mathit{do}\ \ C\ \mathit{done}\ \{I \wedge \neg B\}}$$

- invariant `I` ≪ sorti du chapeau ≫
- si corps de la boucle préserve `I`

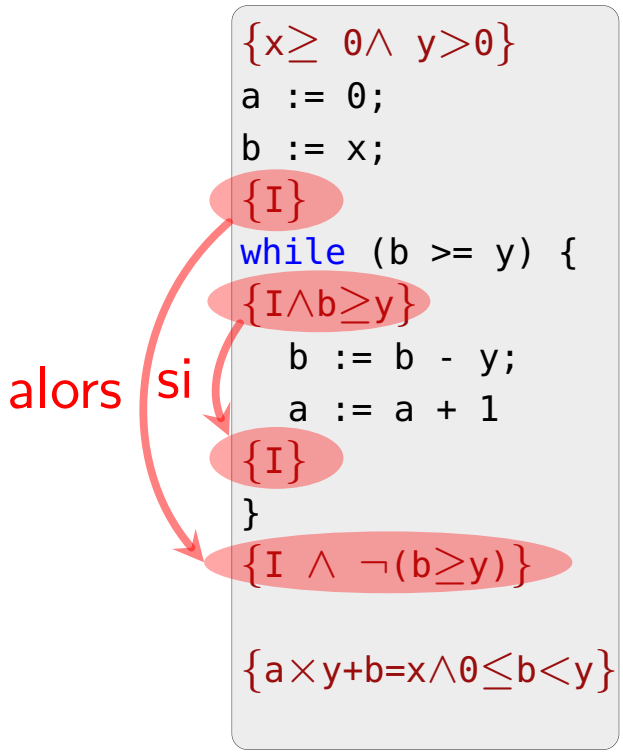
```
{x≥ 0∧ y>0}
a := 0;
b := x;

while (b >= y) {
{I∧b≥y}
  b := b - y;
  a := a + 1
{I}
}


{a×y+b=x∧0≤b<y}
```

si

Invariant

$$\textsc{While}\ \frac{\{I \wedge B\}\ C\ \{I\}}{\{I\}\ \mathit{while}\ \ B\ \mathit{do}\ \ C\ \mathit{done}\ \{I \wedge \neg B\}}$$

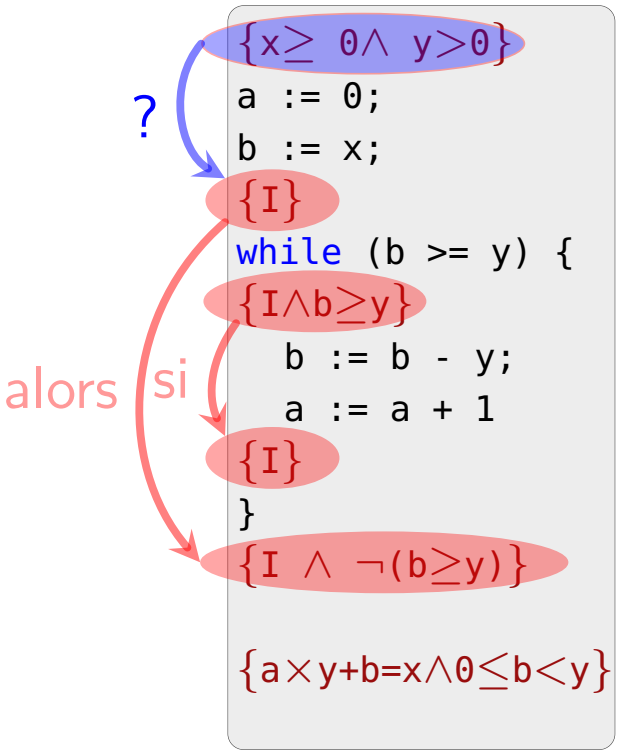
- invariant `I` ≪ sorti du chapeau ≫
- si corps de la boucle préserve `I`
- alors boucle préserve `I`



Invariant

$$\textsc{While}\ \frac{\{I \wedge B\}\ C\ \{I\}}{\{I\}\ \mathit{while}\ \ B\ \mathit{do}\ \ C\ \mathit{done}\ \{I \wedge \neg B\}}$$

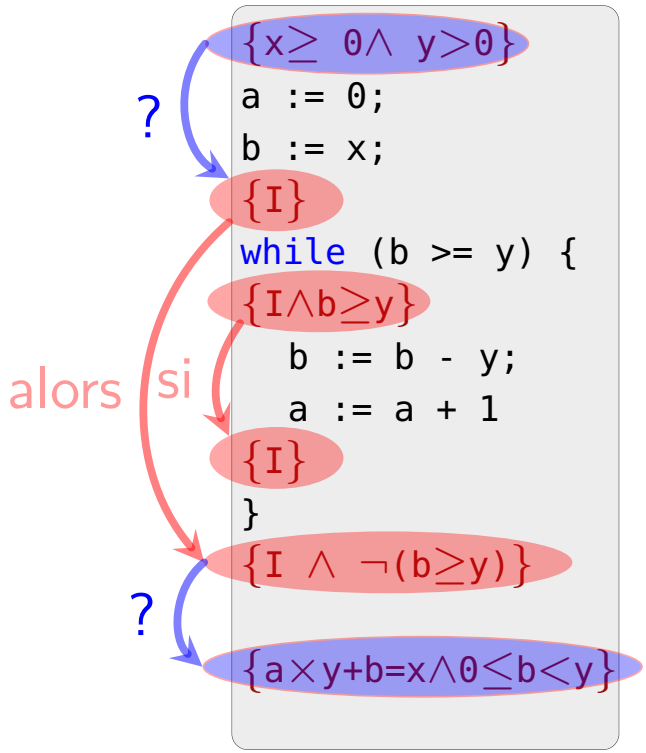
- invariant `I` ≪ sorti du chapeau ≫
- si corps de la boucle préserve `I`
- alors boucle préserve `I`
- reste à montrer que `I` vrai avant la boucle



Invariant

$$\textsc{While}\ \frac{\{I \wedge B\}\ C\ \{I\}}{\{I\}\ \mathit{while}\ \ B\ \mathit{do}\ \ C\ \mathit{done}\ \{I \wedge \neg B\}}$$

- invariant `I` ≪ sorti du chapeau ≫
- si corps de la boucle préserve `I`
- alors boucle préserve `I`
- reste à montrer que `I` vrai avant la boucle
- et que `I` $\wedge \neg B$ suffisant pour la suite



Invariant

Construction de la déduction totale :

$$\dfrac{\dfrac{\dots}{\{I \wedge b \geq y\}\ \text{corps}\ \{I\}}}{\{I\}\ \texttt{while (b>=y)}\{\text{corps}\}\ \{I \wedge \neg(b \geq y)\}}$$

```
{x≥ 0∧ y>0}
a := 0;
b := x;
{I}
while (b >= y) {
{I∧b≥y}
  b := b - y;
  a := a + 1
{I}
}
{I ∧ ¬(b≥y)}

{a×y+b=x∧0≤b<y}
```

Invariant

Construction de la déduction totale :

$$\dfrac{\dfrac{\dfrac{\dots}{\{\mathtt{I} \wedge \mathtt{b} \geq \mathtt{y}\}\ \mathtt{corps}\ \{\mathtt{I}\}}}{\{\mathtt{I}\}\ \mathtt{while\ (b>=y)\{corps\}}\ \{\mathtt{I} \wedge \neg(\mathtt{b} \geq \mathtt{y})\}}}{\text{CONSEQ}\ \dfrac{\{\mathtt{P}\}\ \mathtt{a:=0;b:=y;\ while\ (b>=y)\{corps\}}\ \{\mathtt{I} \wedge \neg(\mathtt{b} \geq \mathtt{y})\}}{\{\mathtt{P}\}\ \mathtt{a:=0;b:=y;\ while\ (b>=y)\{corps\}}\ \{\mathtt{Q}\}}}$$

```
{x≥ 0∧ y>0}
a := 0;
b := x;
{I}
while (b >= y) {
{I∧b≥y}
  b := b - y;
  a := a + 1
{I}
}
{I ∧ ¬(b≥y)}

{a×y+b=x∧0≤b<y}
```

Invariant

Construction de la déduction totale :

$$\text{CONSEQ}\ \frac{\dfrac{\dots}{\{P\}\ a:=0;b:=y\ \{\phi_1\}}\qquad \dfrac{\dots}{\{I \wedge b \geq y\}\ corps\ \{I\}}}{\{P\}\ \ a:=0;b:=y\ \ \{I\}\qquad \{I\}\ \ \texttt{while}\ (b>=y)\{corps\}\ \ \{I \wedge \neg(b \geq y)\}}$$

$$\text{CONSEQ}\ \frac{\{P\}\ \ a:=0;b:=y;\ \texttt{while}\ (b>=y)\{corps\}\ \ \{I \wedge \neg(b \geq y)\}}{\{P\}\ \ a:=0;b:=y;\ \texttt{while}\ (b>=y)\{corps\}\ \ \{Q\}}$$

```
{x≥ 0∧ y>0}
a := 0;
b := x;
{I}
while (b >= y) {
{I∧b≥y}
  b := b - y;
  a := a + 1
{I}
}
{I ∧ ¬(b≥y)}

{a×y+b=x∧0≤b<y}
```

Invariant

Construction de la déduction totale :

$$\dfrac{\dfrac{\dfrac{\dots}{\{P\}\ \texttt{a:=0;b:=y}\ \{\phi_1\}}\ \text{CONSEQ}\qquad \dfrac{\dots}{\{I \wedge b \geq y\}\ \text{corps}\ \{I\}}}{\{P\}\ \texttt{a:=0;b:=y}\ \{I\}\quad \{I\}\ \texttt{while (b>=y)}\{\text{corps}\}\ \{I \wedge \neg(b \geq y)\}}}{\dfrac{\{P\}\ \texttt{a:=0;b:=y; while (b>=y)}\{\text{corps}\}\ \{I \wedge \neg(b \geq y)\}}{\{P\}\ \texttt{a:=0;b:=y; while (b>=y)}\{\text{corps}\}\ \{Q\}}\ \text{CONSEQ}}$$

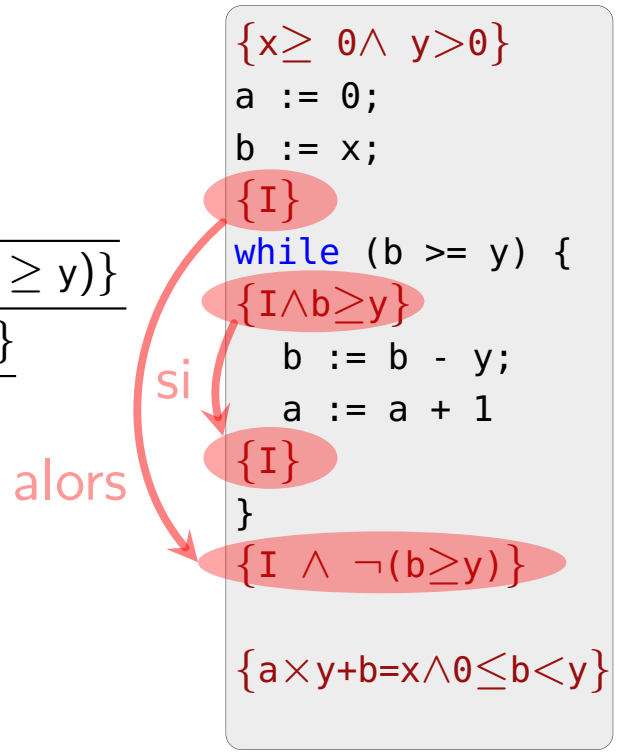
```
{x≥ 0∧ y>0}
a := 0;
b := x;
{I}
while (b >= y) {
{I∧b≥y}
  b := b - y;
  a := a + 1
{I}
}
{I ∧ ¬(b≥y)}

{a×y+b=x∧0≤b<y}
```

si

Invariant

Construction de la déduction totale :

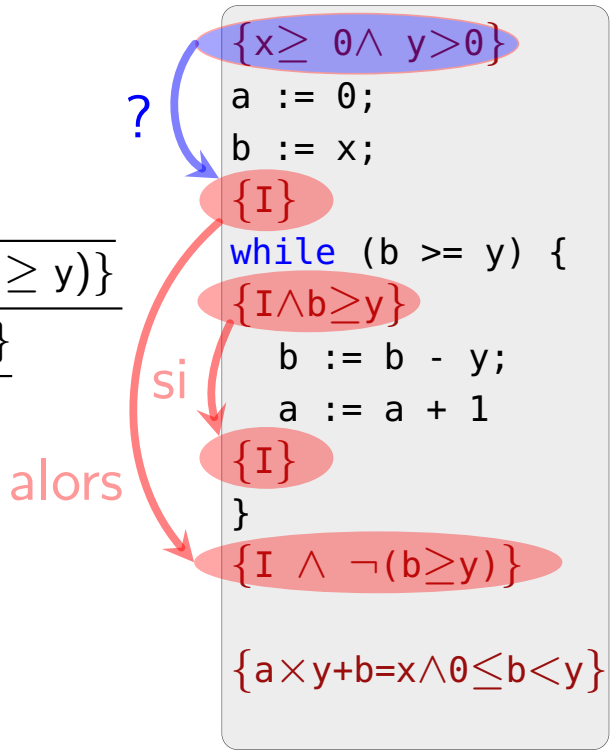
$$\dfrac{\dfrac{\dfrac{\dots}{\{P\}\ a:=0;b:=y\ \{\phi_1\}}}{\{P\}\ \ a:=0;b:=y\ \ \{I\}}\ \text{CONSEQ} \qquad \dfrac{\dfrac{\dots}{\{I \wedge b \geq y\}\ \text{corps}\ \{I\}}}{\{I\}\ \ \texttt{while (b>=y)}\{\text{corps}\}\ \ \{I \wedge \neg(b \geq y)\}}}{\dfrac{\{P\}\ \ a:=0;b:=y;\ \texttt{while (b>=y)}\{\text{corps}\}\ \ \{I \wedge \neg(b \geq y)\}}{\{P\}\ \ a:=0;b:=y;\ \texttt{while (b>=y)}\{\text{corps}\}\ \ \{Q\}}\ \text{CONSEQ}}$$



Invariant

Construction de la déduction totale :

$$\textsc{conseq}\ \dfrac{\textsc{conseq}\ \dfrac{\dfrac{\ldots}{\{P\}\ a:=0;b:=y\ \{\phi_1\}}\ ?\quad \dfrac{\ldots}{\{I \wedge b \geq y\}\ corps\ \{I\}}}{\{P\}\ \ a:=0;b:=y\ \ \{I\}\ \ \{I\}\ \texttt{while}\ (b>=y)\{corps\}\ \ \{I \wedge \neg(b \geq y)\}}}{\{P\}\ \ a:=0;b:=y;\ \texttt{while}\ (b>=y)\{corps\}\ \ \{I \wedge \neg(b \geq y)\}}\ \textsc{conseq}\ \dfrac{\{P\}\ \ a:=0;b:=y;\ \texttt{while}\ (b>=y)\{corps\}\ \ \{I \wedge \neg(b \geq y)\}}{\{P\}\ \ a:=0;b:=y;\ \texttt{while}\ (b>=y)\{corps\}\ \ \{Q\}}$$

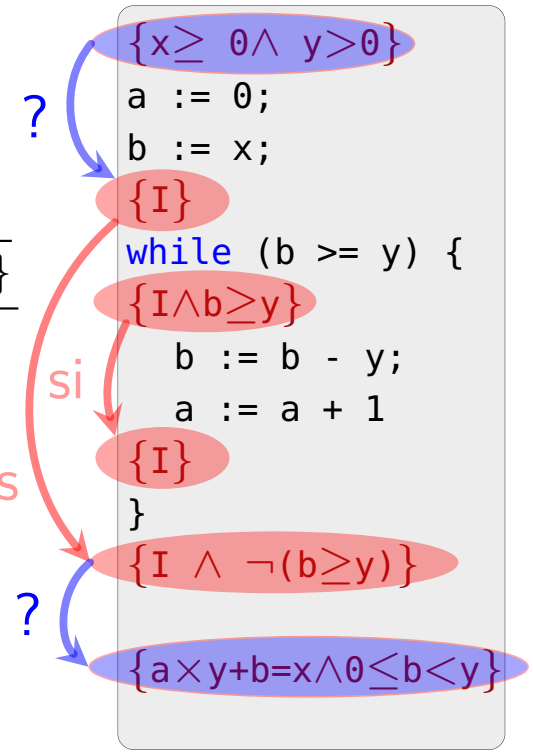


Invariant

Construction de la déduction totale :

$$\dfrac{\dfrac{\dfrac{\dots}{\{P\}\ a:=0;b:=y\ \{\phi_1\}}\ ?}{\{P\}\ \ a:=0;b:=y\ \ \{I\}}\ \textsc{conseq} \qquad \dfrac{\dots}{\{I \wedge b \geq y\}\ corps\ \{I\}}}{\{I\}\ \texttt{while}\ (b>=y)\{corps\}\ \{I \wedge \neg(b \geq y)\}}$$

$$\dfrac{\{P\}\ \ a:=0;b:=y;\ \texttt{while}\ (b>=y)\{corps\}\ \ \{I \wedge \neg(b \geq y)\}}{\{P\}\ \ a:=0;b:=y;\ \texttt{while}\ (b>=y)\{corps\}\ \ \{Q\}}\ ?\ \textsc{conseq}$$

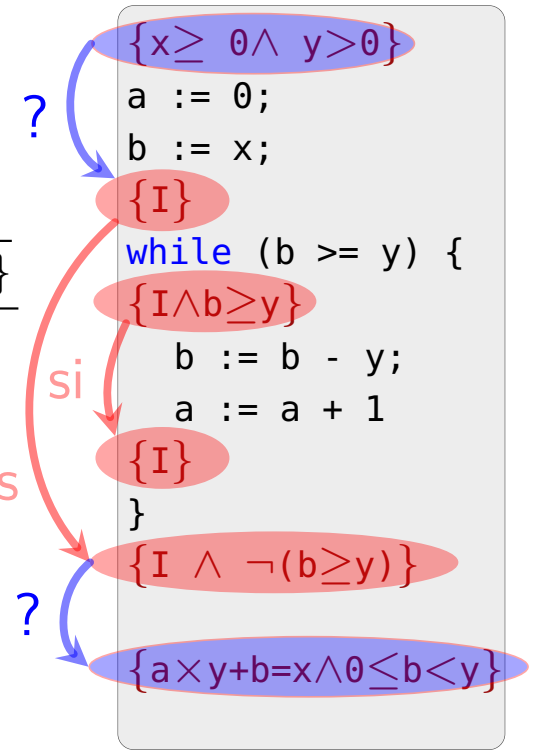


Invariant

Construction de la déduction totale :

$$\dfrac{\dfrac{\dfrac{\dots}{\{P\}\ a:=0;b:=y\ \{\phi_1\}}\ ?}{\{P\}\ \ a:=0;b:=y\ \ \{I\}}\ \text{CONSEQ} \qquad \dfrac{\dots}{\{I \wedge b \geq y\}\ \text{corps}\ \{I\}}}{\{I\}\ \texttt{while (b>=y)}\{\text{corps}\}\ \{I \wedge \neg(b \geq y)\}}$$

$$\dfrac{\{P\}\ \ a:=0;b:=y;\ \texttt{while (b>=y)}\{\text{corps}\}\ \ \{I \wedge \neg(b \geq y)\}}{\{P\}\ \ a:=0;b:=y;\ \texttt{while (b>=y)}\{\text{corps}\}\ \ \{Q\}}\ \text{CONSEQ}\ ?$$



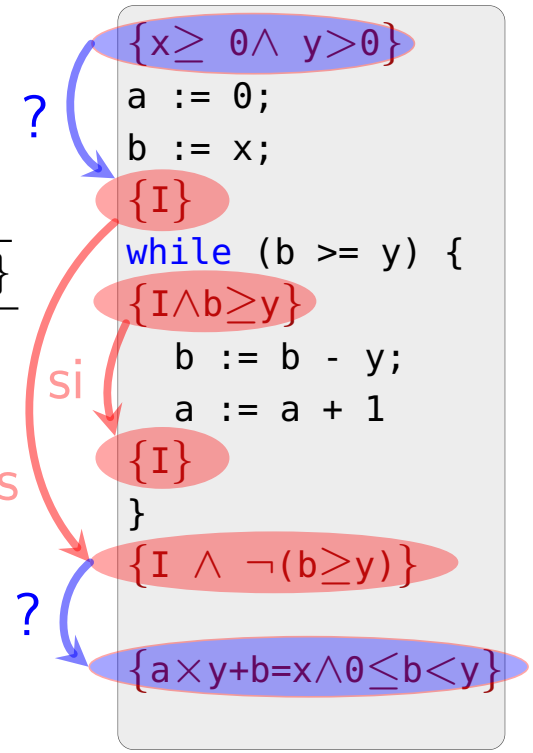
- invariant `I` ≪ sorti du chapeau ≫

Invariant

Construction de la déduction totale :

$$\dfrac{\dfrac{\dfrac{\dots}{\{P\}\ a:=0;b:=y\ \{\phi_1\}}\ ?\quad \dfrac{\dots}{\{I \wedge b \geq y\}\ corps\ \{I\}}}{\{P\}\ \ a:=0;b:=y\ \ \{I\}\quad \{I\}\ \texttt{while}\ (b>=y)\{corps\}\ \{I \wedge \neg(b \geq y)\}}\ \text{CONSEQ}}{\{P\}\ \ a:=0;b:=y;\ \texttt{while}\ (b>=y)\{corps\}\ \ \{I \wedge \neg(b \geq y)\}}\ \text{CONSEQ}$$

$$\dfrac{\{P\}\ \ a:=0;b:=y;\ \texttt{while}\ (b>=y)\{corps\}\ \ \{I \wedge \neg(b \geq y)\}}{\{P\}\ \ a:=0;b:=y;\ \texttt{while}\ (b>=y)\{corps\}\ \ \{Q\}}\ ?$$

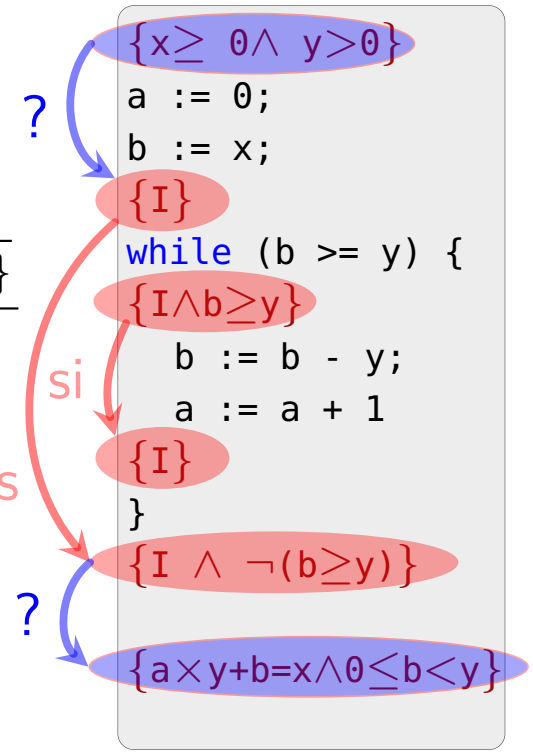


- invariant `I` ≪ sorti du chapeau ≫
- force l'application de CONSEQ avant et après

Invariant

Construction de la déduction totale :

$$\textsc{conseq}\ \frac{\textsc{conseq}\ \dfrac{\dfrac{\ldots}{\{P\}\ \texttt{a:=0;b:=y}\ \{\phi_1\}}\ ?\quad \dfrac{\ldots}{\{I \wedge b \geq y\}\ \text{corps}\ \{I\}}}{\{P\}\ \texttt{a:=0;b:=y}\ \{I\}\quad \{I\}\ \texttt{while (b>=y)}\{\text{corps}\}\ \{I \wedge \neg(b \geq y)\}}}{\{P\}\ \texttt{a:=0;b:=y; while (b>=y)}\{\text{corps}\}\ \{I \wedge \neg(b \geq y)\}}\quad \textsc{conseq}\ \frac{\{P\}\ \texttt{a:=0;b:=y; while (b>=y)}\{\text{corps}\}\ \{I \wedge \neg(b \geq y)\}}{\{P\}\ \texttt{a:=0;b:=y; while (b>=y)}\{\text{corps}\}\ \{Q\}}\ ?$$



- invariant `I` « sorti du chapeau »
- force l'application de CONSEQ avant et après
- avant = `I` vrai avant la boucle ?
- après = `I` suffisant pour la suite ?