# Loop proving example 1

Let $x_0, y_0$ = values at start of an iteration, $x, y$ = values at end of that iteration

$x = n \wedge y = 1$ {

```
while (x!=0) {y = y*x;  x = x-1;}
```

} $y = n!$

- ***Invariant I:*** $y = (x+1) \cdot \ldots \cdot n$
- ***I is an invariant:*** $y_0 = (x_0+1) \cdot \ldots \cdot n \;\&\; y = y_0 x_0 \;\&\; x = x_0 - 1 \Rightarrow y = (x+1) \cdot \ldots \cdot n$
- ***I holds at the start:*** $x = n$, and $(n+1) \cdot \ldots \cdot n = 1$ (i.e. product of empty set)
- ***Q holds at the end:*** $y = (x+1) \cdot \ldots \cdot n \wedge x = 0 \Rightarrow y = 1 \cdot 2 \cdot \ldots \cdot n$
- $T(x, y, n) =$ $x$
- $T(x, y, n) \geq 0$: Loop stops when $x$ gets to zero (should add "$n \geq 0$" to pre-condition)
- $T(x, y, n) < T(x_0, y_0, n)$: Clearly, $x$ decreases on every iteration

# Loop proving example 2

$a = lis \wedge b = 0$ {

```
while (a != []) { b = b + hd(a); a = tl(a); }
```

} $b = \Sigma lis$

- ***Invariant I:*** $b = \Sigma lis - \Sigma a$
- ***I is an invariant:*** $b_0 = \Sigma lis - \Sigma a_0 \wedge b = b_0 + hd\, a_0 \;\&\; a = tl\, a_0 \Rightarrow b = \Sigma lis - \Sigma a$
- ***I holds at the start:*** $b = 0 = \Sigma lis - \Sigma lis$
- ***Q holds at the end:*** $a = [] \Rightarrow \Sigma lis - \Sigma a = \Sigma lis$
- $T(a, b, lis) = |a|$
- $T(a, b, lis) \geqslant 0$: Length of a list always $\geq 0$
- $T(a, b, lis) < T(a_0, b_0, lis)$: Size of a decreases in every iteration

# Loop proving example 3

$a > 0 \wedge b > 0 \wedge a = x \wedge b = y$

```
{ while (a != b) if  (a > b) a = a - b;
                     else b = b - a; }
```
$a = gcd(\cancel{a_0}\,x, \cancel{b_0}\,y)$

- ***Invariant I:*** $gcd(a,b) = gcd(x,y)$
- ***I is an invariant:*** Theorem: $n > m \Rightarrow gcd(n,m) = gcd(n-m, m)$
- ***I holds at the start:*** $a = x \wedge b = y \Rightarrow gcd(a,b) = gcd(x,y)$
- ***Q holds at the end:*** $a = b \Rightarrow a = gcd(a,b)$
- $T(a,b,x,y) = a + b$
- $T(a,b,x,y) \geq 0$: a and b are initially positive, and we always subtract the smaller from the larger
- $T(a,b,x,y) < T(a_0,b_0,x,y)$: Either a or b is decreased in each iteration, while the other is unchanged.

# Loop proving example 4

$x = 0 \wedge y = 0$ {

```
while (y < n) { y = y + 1; x:= x + y; }
```

} $x = 1 + \cdots + n$

- ***Invariant I:*** $x = \sum_{i=1}^{y} i$
- ***I is an invariant:*** $x_0 = \sum_{i=1}^{y_0} i \wedge y = y_0 + 1 \wedge x = x_0 + y \Rightarrow x = \sum_{i=1}^{y} i$
- ***I holds at the start:*** $\Sigma$ over empty set $= 0$
- ***Q holds at the end:*** $y = n \Rightarrow x = \sum_{i=1}^{n} i$
- $T(x, y, n) = n - y$
- $T(x, y, n) \geqslant 0$: Loop ends when $y = n$
- $T(x, y, n) < T(x_0, y_0, n)$: Clearly, $y$ increases in every iteration

# Loop proving example 5

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

```
while (z != n) { y = x + y; x = y - x; z = z + 1; }
```

} $y = fib(n)$

- ***Invariant I:*** $y = fib(z) \wedge x = fib(z-1)$
- ***I is an invariant:*** $y = fib(z_0) + fib(z_0 - 1) = fib(z_0 + 1) = fib(z)$
  $x = y - x_0 = fib(z_0) + fib(z_0 - 1) - fib(z_0 - 1) = fib(z_0) = fib(z-1)$
- ***I holds at the start:*** $fib(1) = 1$, $fib(0) = 0$
- ***Q holds at the end:*** Immediate
- $T(x, y, z, n) = n - z$
- $T(x, y, z, n) \geqslant 0$: Loop terminates when $z = n$
- $T(x, y, z, n) < T(x_0, y_0, z_0, n)$: clear

# Loop proving example 6

(Notation: $|l| = length(l)$ )

$x = lst \wedge y = 0$ {

```
while (x != []) { x = tl x; y = y + 1; }
```

} $y = length(lst)$

- ***Invariant I:*** $y = |lst| - |x|$
- ***I is an invariant:*** $y_0 = |lst| - |x_0| \Rightarrow y_0 + 1 = |lst| - |tl(x_0)|$
- ***I holds at the start:*** $0 = |lst| - |lst|$
- ***Q holds at the end:*** $x = [] \Rightarrow |lst| - |x| = |lst|$
- $T(x, y, lst) = |x|$
- $T(x, y, lst) \geq 0$: Length of lists always $\geq 0$
- $T(x, y, lst) < T(x_0, y_0, lst)$: Clear

# Loop proving example 7

$x = lst \wedge y = [] \{$

```
while (x != []) { y = hd x :: y; x = tl x; }
```

$\} \; y = reverse(lst)$

- ***Invariant I:*** reverse (y) @ x = lst
- ***I is an invariant:*** $rev(y_0) @ (x_0^h :: x_0^t) = rev(x_0^h :: y_0) @ x_0^t$
- ***I holds at the start:*** $y = [] \Rightarrow rev(y) = [] \Rightarrow rev(y) @ x = x$
- ***Q holds at the end:*** $x = [] \Rightarrow rev(y) @ x = rev(y)$ $\wedge$ $rev(y) = lst \Rightarrow y = rev(lst)$
- $T(x, y, lst) \; = \; |x|$
- $T(x, y, lst) \geq 0$: as above
- $T(x, y, lst) < T(x_0, y_0, lst)$: clear

# Hoare logic

- **C.A.R. Hoare presented a logic — axioms and rules of inference, similar to SOS rules — for proving Hoare triples.**

(Assignment) $P[e/x] \{ x = e \} P$

(While) $P \{ \text{while } (b)\ S \} Q$
$I \wedge b \{ S \} I$
(if $P \wedge b \supset I$ and $P \wedge \neg b \supset Q$)

(Sequence) $P \{ S_1;\ S_2 \} Q$
$P \{ S_1 \} R$
$R \{ S_2 \} Q$

(If) $P \{ \text{if } (b)\ S_1 \text{ else } S_2 \} Q$
$P \wedge b \{ S_1 \} Q$
$P \wedge \neg b \{ S_2 \} Q$

(Consequence) $P \{ S \} Q$
$P' \{ S \} Q'$
(if $P \supset P'$ and $Q' \supset Q$)