1 Explain rigorously why each of these triples hold:

(a) $\{\{x == y\}\}\ z := x - y\ \{\{z == 0\}\}$
We assume that x equals y. If x equals y, then x subtracted by y will equal 0. As z equals x subtracted by y, z equals 0 - which is the postcondition

(b) $\{\{\text{true}\}\}\ x := 100\ \{\{x == 100\}\}$
As the term true is always true we can ignore the precondition. If x equals 100, then x will equal 100 - which is the postcondition

(c) $\{\{0 \leq x < 100\}\}$ x := x + 1 $\{\{0 \leq x \leq 100\}\}$
As x $\in [0, 100)$, x + 1 $\in [0 + 1, 100 + 1)$ which is equivalent with x + 1 $\in [1, 101)$ which is equivalent with x + 1 $\in [0, 100]$ - which is the postcondition

2 For each of the following triples, find initial values for x and y that demonstrate that the triple does not hold.

(a) $\{\{\text{true}\}\}\ x := 2 * y\ \{\{y \leq x\}\}$
When y equals -1, x equals -2. As -2 is less than -1, $y > x$ - which contradicts the postcondition

(b) $\{\{0 \leq x\}\}\ x := x - 1\ \{\{0 \leq x\}\}$
When $x = 0$, $x - 1 = -1$ and as $-1 < 0$ the postcondition doesn't stand

3 For each of the following triples, come up with some predicate to replace the question mark to make it a Hoare triple that holds. Make your conditions as precise as possible.

(a) $\{\{0 \leq x < 100\}\}\ x := 2 * x\ \{\{?\}\}$
As the program states that x will become $2*x$, then the inequalities from the precondition will become $0 \leq x < 200$ which can become the postcondition

(b) $\{\{0 \leq x < N\}\}\ x := x + 1\ \{\{?\}\}$
As the program states that x will become $x+1$, then the inequalities from the precondition will become $1 \leq x < N+1$ which can become the postcondition

4 For each of the following triples, come up with some predicate to replace the question mark to make it a Hoare triple that holds. Make your conditions as precise as possible.

(a) $\{\{?\}\}\ x := 400\ \{\{x == 400\}\}$
As we need x to be 400, and the program states that x will be 400, the precondition can be anything

(b) $\{\{?\}\}\ x := 65\ \{\{y \leq x\}\}$
If $x = 65$ then $y \leq 65$, which should be the precondition

5 Write the program which computes the sum of first n natural numbers.

For the following Hoare Logic formula we have:
P: $n \in \mathbb{N}$
I: sum $== \frac{i(i+1)}{2}$
b: $i \leq n$
c: $\text{sum} = \text{sum} + 1,\ i = i + 1$
Q: $\text{sum} = \frac{n(n+1)}{2}$
t: $n - 1$

$$\frac{P \Longrightarrow I \quad I \Longrightarrow t \geq 0 \quad \{\{I \land b \land t\}\}\ c\ \{\{I \land t < N\}\} \quad (I \land \neg b) \Longrightarrow Q}{\{\{P\}\}\ \text{while } b \text{ do } c\ \{\{Q\}\}}$$

```
method sum(n: nat) returns (sum: nat)
  ensures sum == n * (n + 1) / 2
{
  var i := 0;
  sum := 0;
  while i <= n
    invariant 0 <= i <= n + 1
    invariant sum == i * (i - 1) / 2
    decreases n - i
  {
    sum := sum + i;
    i := i + 1;
  }
}
```

6 Write the program which computes the product of first n natural numbers. Prove its total correctness.

```
function factorial(n: nat) : nat
  decreases n
{
  if n == 0 then 1 else n * factorial(n - 1)
}

method product(n: nat) returns (prod: nat)
  ensures prod == factorial(n)
{
  var i := 1;
  prod := 1;
  while i <= n
    invariant 0 <= i <= n + 1
    invariant prod == factorial(i - 1)
    decreases n - i
  {
    prod := prod * i;
    i := i + 1;
  }
}
```