Formal Verification – Inference Rules

① $$\frac{}{\{Q[E/X]\}\ X=E\ \{Q\}}$$ assignment

② $$\frac{P \to P' \qquad \{P'\}\ C\ \{Q\}}{\{P\}\ C\ \{Q\}}$$ "precondition strengthening" implied

③ $$\frac{\{P\}\ C\ \{Q'\} \qquad Q' \to Q}{\{P\}\ C\ \{Q\}}$$ "post condition weakening" implied

④ $$\frac{\{P\}\ C_1\ \{Q\} \qquad \{Q\}\ C_2\ \{R\}}{\{P\}\ C_1; C_2\ \{R\}}$$ composition

⑤ $$\frac{\{P \wedge B\}\ C_1\ \{Q\} \qquad \{P \wedge \neg B\}\ C_2\ \{Q\}}{\{P\}\ \text{if } (B)\ C_1 \text{ else } C_2\ \{Q\}}$$ if-then-else

⑥ $$\frac{\{P \wedge B\}\ C\ \{Q\} \qquad (P \wedge \neg B) \to Q}{\{P\}\ \text{if } (B)\ C\ \{Q\}}$$ if-then

⑦ $$\frac{\{(I \wedge B)\}\ C\ \{I\}}{\{I\}\ \text{while } (B)\ C\ \{(I \wedge (\neg B))\}}$$ partial-while

Assignments

Complete the following annotations

① ② ③

```
x = 2;          x = 2;          x = 2;
{{x = 2}}       {{x = y}}       {{x = 0}}
```

④ ⑤

```
x = x + 1;      x = y;
{{ x = n+1 }}   {{2 · x = x + y}}
```

Assignments

Prove that the following program satisfies the given triple under partial correctness

① $\{((x = x_0) \wedge (y = y_0))\}$

```
t = x;
x = y;
y = t;
```

$\{((x = y_0) \wedge (y = x_0))\}$

② $\{true\}$

```
z = x;
z = z + y;
u = z;
```

$\{(u = x + y)\}$

Conditional Statements (If-Then)

Prove that the following program satisfies the given triple under partial correctness

$\{ true \}$

```
if (max < x) {
    max = x;
}
```

$\{ (max \geq x) \}$

Conditional Statements (If-Then-Else)

Prove that the following program satisfies the given triple under partial correctness.

{ true }

```
if (x > y) {
    max = x;
} else {
    max = y;
}
```

{ ((x > y) ∧ (max = x)) ∨ ((x ≤ y) ∧ (max = y)) }

While Loops

Prove that the following program satisfies the given triple under partial correctness.

$\{(x \geq 0)\}$

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

$\{(y = x!)\}$

While Loops

Prove that the following program satisfies the given triple under partial correctness

$\{ (n \geq 0) \wedge (a \geq 0) \}$

```
s = 1;
i = 0;
while (i < n) {
    s = s * a;
    i = i + 1;
}
```

$\{ s = a^n \}$