

# 课后阅读：反向赋值语句推理规则

## 1 变量赋值语句规则（反向）与最弱前条件

后条件  $Q$  与程序  $c$  的最弱前条件  $P$  是指满足下面条件的断言  $P$ 。

- $\{P\}c\{Q\}$  成立；
- 如果  $\{P'\}c\{Q\}$  成立，那么  $P'$  能推出  $P$ ；

例如，后条件 `temp == m && y == n` 与程序 `temp = x` 的一个最弱前条件是： `x == m && y == n`。

变量赋值规则（反向）：

那么  $\{P[x \mapsto e]\} x = e \{P\}$

## 2 符号执行（反向）

反向符号执行

```
//@ require true
//@ ensure x == 10
//@ [generated] 0 <= 10
x = 0;
//@ inv x <= 10
while (x < 10) do
{
    //@ [target] x <= 10 && x < 10
    //@ [generated] x + 1 <= 10
    x = x + 1
    //@ [generated] x <= 10
}
//@ [target] x <= 10 && !(x < 10)
```

生成的验证条件

```
true |-- 0 <= 10
```

```
x <= 10 && x < 10 |-- x + 1 <= 10
```

```
x <= 10 && !(x < 10) |-- x == 10
```