Non-assessed Exercise – Solutions
Bounded Model Checking

1. Control flow simplification (converting for loops to while loops; removing side-effects):

```
$x := 0$;
$i := 1$;
while ($i \leq 10$) {
    $j := j + 1$;
    if ($i < 4$) {
        $x\ := x + i * i$;
    } else {
        $x := x + i * i * j$;
        $j := j + 1$;
    }
    $i := i + 1$;
}
assert $x < 100$;
```

Loop unwinding (assuming an unrolling depth of 1):

```
$x := 0$;
$i := 1$;
if ($i \leq 10$) {
    $j := j + 1$;
    if ($i < 4$) {
        $x := x + i * i$;
    } else {
        $x := x + i * i * j$;
        $j := j + 1$;
    }
    $i := i + 1$;
}
assert $x < 100$;
```

Conversion to single static assignment form:

```
$x1 := 0$;
$i1 := 1$;
**if** $(i1 \leq 10)$ {
    $j1 := j0 + 1$;
    **if** $(i1 < 4)$ {
        $x2 := x1 + i1 * i1$;
    } **else** {
        $x3 := x1 + i1 * i1 * j1$;
        $j2 := j1 + 1$;
    }
    $x4 := (i1 < 4)\ ?\ x2 : x3$;
    $j3 := (i1 < 4)\ ?\ j1 : j2$;
    $i2 := i1 + 1$;
}
$x5 := (i1 \leq 10)\ ?\ x4 : x1$;
$j4 := (i1 \leq 10)\ ?\ j3 : j0$;
$i3 := (i1 \leq 10)\ ?\ i2 : i1$;
**assert** $x5 < 100$;
```

Conversion to conjunctive normal form (CNF).

$$(x_1 = 0) \wedge (i_1 = 1) \wedge (j_1 = j_0 + 1) \wedge (x_2 = x_1 + i_1 * i_1) \wedge (x_3 = x_1 + i_1 * i_1 * j_1) \wedge \ldots$$

$$\ldots (j_2 = j_1 + 1) \wedge (x_4 = (i_1 < 4)\ ?\ x_2 : x_3) \wedge (j_3 = (i_1 < 4)\ ?\ j_1 : j_2) \wedge (i_2 = i_1 + 1) \wedge \ldots$$

$$\ldots (x_5 = (i_1 \leq 10)\ ?\ x_4 : x_1) \wedge (j_4 = (i_1 \leq 10)\ ?\ j_3 : j_0) \wedge (i_3 = (i_1 \leq 10)\ ?\ i_2 : i_1) \wedge \neg(x_5 < 100)$$

For one unrolling ($k = 1$), as done above, the CNF formula is unsatisfiable, so no violation of the assertion is found. However, we cannot say that the program is correct without considering larger values of k.

In terms of the original program, for the first three iterations of the **for** loop, the 'then' branch of the **if** statement is taken and so x will not exceed 100. This is why no assertion violation is found for $k = 1$.

For larger numbers of unrolling, more precisely for $k > 3$, we can pick an arbitrary value of j which will make the assertion fail. This could be found using bounded model checking using, e.g. $k = 4$ unrollings.