**Verification Conditions**

The implementation of our wlp-functions:

|  |  |  |
| --- | --- | --- |
| **Hoare Logic** | **(without side effects)** | **Side effects** |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

Some explanations for our new definitions:

Reminder:

* wlp⟦c⟧Q Maps a command (c) and a postcondition (Q, an assertion) to the weakest precondition (W) that will make {W} c {Q} hold.
* ⊨{P} c {Q} iff ⊨ P →wlp⟦c⟧Q

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **command** | **Hoare** | |  | **VC** |
| assume |  |  | iff ⊨ P →(¬b ∨ Q) | ⊨ P→ wlp⟦assume⟧Q |
| assert |  |  | iff ⊨ P → b ∧ Q | ⊨ P→ wlp⟦assert⟧Q |

For the **while command** we used its verification conditions:

|  |  |  |
| --- | --- | --- |
| **command** | **Hoare** | **VC** |
| while |  | 1. ⊨ P → Inv 2. ⊨ Inv ∧ ¬b →Q   (3) VC ({Inv ∧ b} c ({Inv} ) |

Verification conditions (2),(3) are indepedent of P (the precondition) and can be verified independently. Condition (1) is similiar to the wlp-conditions (⊨ P→ wlp⟦c⟧Q) so in our implementation of the wlp function we handle the inv as if it’s the wlp of the while command, with the addition of “side effects” (the independent conditions 2,3).

Formally: if (2)+(3) are valid, then iff {P} (while {inv} b do c) {Q}. Otherwise,