5.1. Demonstration of correctness of a program based on unit testing

5.1.1 Assumptions

Let P be a program to test. Assume that the program is syntactically valid.

5.1.2 Demonstrate that a program P is correct for a set of tests

5.1.2.1 Declaration of a program block

$Let\ inputs_{fpb} \equiv \{v \mid v\ is\ a\ value\}$

$Let\ func_{fpb}: inputs_{fpb} \rightarrow output_{fpb}, where\ func_{fpb}\ is\ a\ function$

$Let\ pb_i \equiv (func_{fpb}, inputs_{fpb}, output_{fpb})$

5.1.2.2 Declaration of all program blocks

$Let\ Blocks(P) \equiv \{pb_i \mid pb_i\ is\ a\ program\ block\ of\ P\}$

5.1.2.3 Declaration of a test

$Let\ inputs_{ft} \equiv \{v \mid v\ is\ a\ value\}$

$Let\ func_{ft}: inputs_{ft} \rightarrow output_{ft}, where\ func_{ft}\ is\ a\ function$

$Let\ t_j \equiv (func_{ft}, inputs_{ft}, output_{ft})$

5.1.2.4 Declaration of all test sets for P

$Let\ TA_{pb_i} \equiv [t_j \mid t_j\ is\ a\ test]$

$Let\ TS_P \equiv \bigcup_{pb_i \in Blocks(P)} TA_{pb_i}$

5.1.2.5 Show the validity and success of a test on a program block

$$success(t_j, pb_i) \equiv \begin{cases} 1, if\ [t_j]_{output_{ft}} = [pb_i]_{output_{fpb}} \\ 0,\ otherwise \end{cases}$$

$$valid(t_j, pb_i) \equiv \begin{cases} 1, if\ [t_j]_{func_{ft}} = [pb_i]_{func_{fpb}}\ and\ [t_j]_{inputs_{ft}} = [pb_i]_{inputs_{fpb}} \\ 0,\ otherwise \end{cases}$$

$$valid(t_j, pb_i) \rightarrow success(t_j, pb_i) = 1$$

5.1.2.6 Demonstrate that the whole program is correct

$correctBlock(TA_{pb_i}, pb_i) \equiv \forall t_j \in TA_{pb_i} \mid valid(t_j, pb_i)$

$correctP\big(TS_P, Blocks(P)\big) \equiv$

$\forall pb_i \in Blocks(P), \forall TA_{pb_i} \in TS_P \mid correctBlock(TA_{pb_i}, pb_i)$

5.2. Demonstration of correctness of a program from requirements point of view.

5.2.1 Considerate the previous parts 1.2.1 to 1.2.4 include as the first part of this proof.

5.2.2 Declaration of requirements

$Let\ condition\ \equiv\ [v \mid v\ is\ a\ value]$

$Let\ conditionsPre \equiv \{condition \mid condition\ is\ a\ set\ of\ values\}$

$Let\ conditionsPost \equiv \{condition \mid condition\ is\ a\ set\ of\ values\}$

$$Let\ pre-condition([t_j]_{inputs_{ft}})\ \equiv \begin{cases} 1, \forall input \in [t_j]_{inputs_{ft}}, \exists\ condition\ \in\ conditionsPre \mid input\ \in\ condition \\ 0, otherwise \end{cases}$$

$$Let\ post-condition([t_j]_{output_{ft}})\ \equiv \begin{cases} 1, \forall output \in [t_j]_{output_{ft}}, \exists\ condition\ \in\ conditionsPost \mid output\ \in\ condition \\ 0, otherwise \end{cases}$$

$$Let\ r_i \equiv\ (pre-condition([t_j]_{inputs_{ft}}), post-condition([t_j]_{output_{ft}}))$$

$$Let\ RA_{pb_i}\ \equiv\ [r_i \mid r_i\ is\ a\ requirement]$$

$$Let\ RS_P\ \equiv\ \bigcup_{pb_i \in Blocks(P)} RS_{pb_i}$$

5.2.3 Test satisfaction of a requirement

$$inputsEquality(t_j, pb_i)\ \equiv \begin{cases} 1, if\ [t_j]_{inputs_{ft}} = [pb_i]_{inputs_{fpb}} \\ 0, otherwise \end{cases}$$

$$outputsEquality(t_j, pb_i)\ \equiv \begin{cases} 1, if\ [t_j]_{output_{ft}} = [pb_i]_{output_{fpb}} \\ 0, otherwise \end{cases}$$

$$conditionInput(t_j, pb_i, r_i)\ \equiv \begin{cases} 1, if\ inputsEquality(t_j, pb_i) | [r_i]_{pre-condition([pb_i]_{inputs_{fpb}})} \\ 0, otherwise \end{cases}$$

$$conditionOutput\ (t_j, pb_i, r_i) \equiv \begin{cases} 1, if\ outputsEquality(t_j, pb_i)\ | [r_i]_{post-condition([pb_i]_{output_{fpb}})} \\ 0, otherwise \end{cases}$$

$$satsify(t_j, pb_i, r_i)\ \equiv \begin{cases} 1, if\ conditionInput(t_j, pb_i, r_i)\ and\ conditionOutput\ (t_j, pb_i, r_i)\ and\ valid(t_j, pb_i) \\ 0, otherwise \end{cases}$$

5.2.4 A specific requirement is satisfied for a program block

$$Let\ satisfactionR(TS_{pb_i}, pb_i, r_i)\ \equiv \begin{cases} 1, if\ \forall t_j \in\ TS_{pb_i} |\ satsify(t_j, pb_i, r_i)\ \rightarrow\ (\exists t_j \in\ TS_{pb_i} | satsify(t_j, pb_i, r_i)\ =\ 1) \\ 0, otherwise \end{cases}$$

5.2.5 All requirements are satisfied for a program block

$$Let\ satisfactionSetR(TS_{pb_i}, pb_i, RS_{pb_i})\ \equiv \begin{cases} 1, if\ \forall t_j \in\ TS_{pb_i}, \forall r_i \in RS_{pb_i} |\ satisfactionR(TS_{pb_i}, pb_i, r_i)\ =\ 1 \\ 0, otherwise \end{cases}$$

5.2.6 All requirements are satisfied for a program

$$Let\ satisfactionALL\ \equiv \forall pb_i \in\ Blocks(P), \forall TS_{pb_i} \in TS_P, RS_{pb_i} \in RS_P \mid satisfactionSetR(TS_{pb_i}, pb_i, RS_{pb_i})\ =\ 1$$