1. Proof the correctness of program based on unit testing

1.1 Assumptions

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

1.2 Proof that a block of P is correct on a test

1.2.1 Declaration of a program block

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

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

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

1.2.2 Declaration of all program blocks

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

1.2.3 Declaration of a test

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

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

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

1.2.4 Declaration of all test sets for P

$Let\ TS_{pb_i}\ =\ \{t_j \mid t_j\ is\ a\ test\}$

$Let\ TS_P\ =\ \bigcup_{pb_i \in Blocks(P)} TS_{pb_i}$

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

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

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

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

1.2.6 Proof that the whole program is correct

$\mathrm{correctBlock}(TS_{pb_i}, pb_i)\ =\ \forall \mathrm{t_j} \in\ TS_{pb_i} \mid \mathrm{valid}(\mathrm{t_j}, pb_i)$

$correctP(TS_P, Blocks(P))\ =\ \forall pb_i \in\ Blocks(P), \forall TS_{pb_i} \in TS_P \mid \mathrm{correctBlock}(TS_{pb_i}, pb_i)$

2. Proof correctness of program from requirements point of view.

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

2.2 Declaration of requirements

Let $pre - condition\ =\ \{v \mid v\ is\ a\ value\}$

Let $\mathrm{post} - condition\ =\ \{v \mid v\ is\ a\ value\}$

$Let\ r_i =\ (pre - condition, post - condition)$

$Let\ RS_{pb_i} = \{r_i \mid r_i\ is\ a\ requirement\}$

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

2.3 Test satisfaction of a requirement

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

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

$$\text{conditionInput}(t_j, pb_i, r_i) = \begin{cases} 1, \text{if inputsEquality}(t_j, pb_i) \text{ and } \forall \text{v} \in [pb_i]_{\text{inputs}_{fpb}} \mid \text{v} \in [r_i]_{\text{pre-condition}} \\ 0, \text{otherwise} \end{cases}$$

$$\text{conditionOutput } (t_j, pb_i, r_i) = \begin{cases} 1, \text{if outputsEquality}(t_j, pb_i) \text{ and } \forall \text{v} \in [pb_i]_{\text{output}_{fpb}} \mid \text{v} \in [r_i]_{\text{post-condition}} \\ 0, \text{otherwise} \end{cases}$$

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

2.4 A specific requirement is satisfied for a program block

Let $\text{satisfactionR}(TS_{pb_i}, pb_i, r_i) =$

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

2.5 All requirements are satisfied for a program block

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

2.6 All requirements are satisfied for a program

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