An Event-B Specification of

SetComprehensions

Set comprehension syntax is the most complex part of the Event-B grammar. This project tests all the ways of expressing set comprehension.

It also demonstrates that a machine variable can be overridden with a non-free variable in a set comprehension.

1		HINE Comprehensive
	1.1	aboolean bools coords numbers
	1.2	assignzPF
	1.3	assignFPSpecialForm
	1.4	assignFPSpecialFormPair
	1.5	assignFPSpecialCase
	1.6	assignFPSpecialCaseWithGlobal
	1.7	assignFPSpecialCaseWithGloball

```
1
MACHINE Comprehensive
                                                                                                                    1.1
VARIABLES
 numbers
  coords
  bools
  aboolean
INVARIANTS
  inv1:
           numbers \subseteq \mathbb{N}
            coords \subseteq \mathbb{N} \times \mathbb{N}
 inv2:
 inv3:
            bools \subseteq \mathbb{N} \times BOOL
            aboolean \in BOOL
  inv4:
EVENT INITIALISATION
THEN
 init1:
             numbers := \varnothing
 init2:
             coords := \emptyset
             \mathit{bools} := \varnothing
 init3:
             aboolean := FALSE
 init4:
END
EVENT assignzPF
                                                                                                                    1.2
Create a calculated set using set comprehension. The non-free variables are explicit before the
dot.
THEN
           numbers := \{x \cdot x \in \mathbb{N} \land x < 10 \mid x\}
 act1:
END
                                                                                                                    1.3
EVENT assignFPSpecialForm
Create another calculated set, the non-free variables are implicit the expression before |.
THEN
            numbers := \{x + 2 \mid x \in \mathbb{N} \land x < 10\}
 act1:
END
                                                                                                                    1.4
{\tt EVENT} \ assign FP Special Form Pair
Another set comprehension, the non-free variables are implicit in the expression.
THEN
            coords := \{ x \mapsto y \mid x \in \mathbb{N} \land y \in \mathbb{N} \land y < x \land x < 10 \}
 act1:
END
                                                                                                                    1.5
EVENT assignFPSpecialCase
The single non-free variable case.
THEN
```

 $numbers := \{x \mid x \in \mathbb{N} \land x < 10\}$

act1: END

${\tt EVENT} \ assign FP Special Case With Global$

The variable aboolean is used to assign the right hand side in the pairs.

THEN

```
\mbox{act1:} \quad bools := \{x \mapsto y \mid x \in \mathbb{N} \land aboolean = y\} END
```

${\tt EVENT} \ assign FP Special Case With Globall$

1.7

1.6

Oups, here a boolean becomes a non-free variable! Which is the reason why it can be typed to $\mathbb N$ in this formula.

THEN

```
\mbox{act1:} \quad numbers := \{x + aboolean \mid x \in \mathbb{N} \land aboolean = \mathbf{1}\} END
```

aboolean, 2 assignFPSpecialCase, 2 assignFPSpecialCaseWithGlobal, 3 assignFPSpecialCaseWithGloball, 3 assignFPSpecialForm, 2 assignFPSpecialFormPair, 2 assignzPF, 2

bools, 2

Comprehensive, 2 coords, 2

INITIALISATION, 2

numbers, 2