CS412 Exercise sheet 7

Refinements

1. The following machine keeps track of passengers booking and cancelling for a coach tour. Provide a refinement which stores the bookings as a partial function with domain $1..50$ and whose range is *pass*. The refinement should provide deterministic operations.

```
MACHINE         Passengers
SETS            PID
VARIABLES       pass
INVARIANT       pass <: PID & card(pass) <= 50
INITIALISATION  pass := {}
OPERATIONS
     book(pp) = PRE   pp:PID & pp /: pass & card(pass) < 50
                THEN  pass := pass \/ {pp}
                END;

     cancel(pp) =  PRE   pp:PID & pp : pass
                   THEN  pass := pass - {pp}
                   END;

     oo <-- query(pp) = PRE   pp:PID
                        THEN  oo := bool(pp : pass)
                        END;

     oo <-- spaces = oo := 50 - card(pass)
END
```

2. Complete any of the following that we didn't have time for in lectures. Suggest linking invariants for the following cases.

   (a) **Abstract** A library has a supply of registered books (*stock*). Some of these are for library use only (*notforloan*). Of the stock that can be lent out, *onloan* is the set currently on loan. Ie:

   $$stock \subseteq BOOK \wedge notforloan \subseteq stock \wedge onloan \subseteq stock$$

   Concrete It is decided to introduce an enumerated type:

   $$STATUS = \{neverloan, readytoloan, outonloan\}$$

   and use the concrete variable $bookarr \subseteq stock \rightarrow STATUS$.

(b) **Abstract** The variable $marked \subseteq ASSIGNMENT$ records the assignments that have been marked so far.

Concrete List of assignments still to mark: $tomark \in \text{iseq}(ASIGNMENT)$.

(c) **Abstract** Info about club members is kept as $minfo \in MID \nrightarrow (NAME \times ADDR)$

Concrete uses $mname \in MID \nrightarrow NAME$ and $maddr \in MID \nrightarrow ADDR$.

3. Suppose the CS department keeps a record of up to 30 students who are willing to act as guides at open days. For each open day, 6 student guides are required. Suppose student identifiers are represented using *SID*.

   (a) Write an abstract machine for this situation which uses the set variables *volunteer* for the (up to) 30 possible guides and *chosen* for the 6 currently selected (although a selection cannot be made until at least 6 students have volunteered). It should include a suitable invariant and initialisation and the following operations:

   - $newvolunteer(vv)$ - to add vv as a volunteer if max not reached;
   - $swap(v1, v2)$ - when the 30 max volunteers has been reached, this replaces one of the current 30 volunteers ($v1$) with a new volunteer ($v2$);
   - *newchoice* - to nondeterministically choose either the initial or a new selection of 6 who aren't currently selected;
   - *query* - to output the current value of "chosen".

   (b) Suppose that a refinement is proposed in which the volunteers and the chosen set are to be represented as sequences of *SID*s. Write a suitable linking invariant for this refinement machine. Consider how instances of concrete and abstract states match up to check you understand the relationship given by your linking invariant.

   (c) Write a refinement machine which uses this linking invariant. Suppose the decision is made that, for the *newchoice* operation, the initial selection is to be made deterministic at this stage but subsequent choices are to be left nondeterministic.

   (d) Try different data refinements for the original machine and see how the linking invariant, initialisation and operations work out for your different approaches. Try some of these things out in the B tool.