Exercise Sheet 7: Development of a Sequential Program

1 Purpose

The purpose of this exercise is to make you developing a program called "the Celebrity Problem"

2 Description of the "Celebrity Problem"

Among a set P of persons there is a celebrity c. We are given a binary relation $knows$ between two *different* persons. More precisely, if the pair $p \mapsto q$ belongs to $knows$ it means that person p knows person q. The characteristic property of the celebrity c is that everyone knows c whereas c does not know anybody. We would like to search for the celebrity by asking who knows or does not know who.

3 Your Task

We distribute to you a Rodin development named "07_celebrity". It only contains a context c0 with three constants: the finite set of persons P (a subset of the set of Natural Numbers $\mathbb{N}$), the binary relation $knows$ (a binary relation from P to P) and the celebrity c (a member of P).

Together with the development "07_celebrity", we distribute to you a tactic profile called "WS_profile2". You must use this tactic profile to perform proofs of your development.

You are asked to develop a program by following the steps described in subsequent sections.

3.1 Making the context c0 more precise

Enlarge the context c0 with more axioms taking account of the following properties:

1. The relation $knows$ is a relation between two different persons of the set P.
2. Every person in P (except c itself) knows c.
3. The celebrity c does not know anybody.

3.2 Initial Machine

Define an initial machine m0. Do not forget to have m0 "sees" c0. Machine m0 has a single event find (besides the initialisation event) setting a result variable r to the celebrity c. Initially, the variable r is assigned to any person of P.

3.3 First Refinement

Refine the previous machine m0 by a machine m1. For this, introduce a new variable Q which is supposed to be a subset of P containing the celebrity c. Initially Q is equal to P.

The purpose of this machine is to introduce two new events, remove1 and remove2, gradually removing elements from the set Q. The event remove1 removes a person p from Q if p knows another person q of Q. The event remove2 removes a person q from Q if q is not known from another person p of Q. Give comments explaining the justification for these removals. This refinement must contain the following events:

1. Initialisation
2. find
3. remove1 with parameter p (defined in the ANY clause)
4. remove2 with parameter q (defined in the ANY clause)

3.4 Second Refinement

Refine the previous machine m1 by a machine m2. For this, introduce two variables R and b. Variable R is a subset of the set P and variable b is a person of P which is not in R. Variable Q disappears but it is related to variables R and b by the following "gluing invariant": $Q = R \cup \{b\}$.

Initially, the variable b is set to a constant $b0$, which is a member of P. Extend context c0 by a context c1 where $b0$ is defined. Do not forget to have machine m1 "sees" context c1.

Refine the events as follows by adding an event remove3:

1. Event remove1 refines abstract event remove1. The abstract parameter p is replaced by b. For this, fill in the clause WITH as follows: $p \quad p = b$. Modify b and R.

2. Event remove2 refines abstract event remove2. The abstract parameter q is replaced by b. For this, fill in the clause WITH as follows: $q \quad q = b$. Modify b and R.

3. Event remove3 refines abstract event remove1. Keep the abstract parameter p and suppose $p \mapsto b \in k$. Do not modify b. Remove p from R.

3.5 Third Refinement

Refine the previous machine m2 by a machine m3.

We suppose now that the set P is exactly the interval $0 \,..\, n$ for some positive number n. For this, extend context c1 by a context c2 defining a positive natural constant n. Add the axiom $P = 0 \,..\, n$. Do not forget to have machine m3 "sees" context c2.

We introduce a new variable a which is a number in the interval $1 \,..\, n + 1$. Initially a is equal to 1 and b is equal to 0 (add the axiom $b0 = 0$ in context c2). The set R disappears. It is related to a and n by the following gluing invariant: $R = a \,..\, n$. Refine the events of previous machine.

3.6 Fourth Refinement

Refine the previous machine m3 by a machine m4.

This machine contains two events: remove12 and remove3. Event remove12 merges events remove1 and remove2 of previous machine. Event remove3 refines event remove3 of previous machine without any modification.

Perform an animation with AnimB. For this, give a small value to n, to c, and to the relation k in various contexts (clause AnimB VALUES).

3.7 Constructing the final program

Apply manually the merging rules presented in the class in order to construct your final program. You should be able to obtain the following program:

```
b, a := 0, 1;
while a ≤ n do
   if b ↦ a ∈ k  ∨  a ↦ b ∉ k  then
      a, b := a + 1, a
   else
      a := a + 1
   end
end ;
r := b
```

3.8 Optional step

Add another refinement by encoding the relation $knows$ in a boolean matrix of size $0\,..\,n \times 0\,..\,n$. Refine machine m4. Construct the new program. Translate it to C and run it.