EventB Exercises 2
A Simple Bank Machine

The objective of this tutorial exercise is to develop EventB models. In all cases the resulting machines should be introduced into the Rodin Toolkit, analyzed, proof obligations generated, the autoprover run and any remaining undischarged proof obligations inspected carefully. In many cases it would be a good idea to animate the machine.

1. **A simple bank** Produce a model, consisting of SimpleBank_ctx and SimpleBank machines, of a very simple bank with the following requirements. Follow the English very carefully.

   - **accounts** the bank customers are represented by accounts. Having obtained an account a customer may use the other operations supported by the bank.
   - **balance** the bank needs to maintain a balance for all accounts.
   - **NewAccount** an operation by which a customer obtains an account identifier. Account identifiers are allocated from a pool (set) of identifiers maintained by the bank.
   - **Deposit** an operation to add an *amount* to an *account* balance.
   - **WithDraw** an operation withdraw an *amount* from an *account*. Customers cannot withdraw more than the balance in their account.
   - **Transfer** an operation that transfers an *amount* of money from one bank account to another.

   **Note:** the balance and all other money amounts can be represented by natural numbers.
Relations and Functions

Given sets $X$ and $Y$, $X \leftrightarrow Y$ is the set of all possible relations between elements of $X$ and $Y$. A relation is a set of pairs relating—in this case—elements of $X$ to elements in $Y$. Notice that relations are directional: from $X$ to $Y$; even though the symbol $\leftrightarrow$ might suggest bidirectional mappings. Notice that $X \leftrightarrow Y$ is identical to $\mathcal{P}(X \times Y)$.

Relations are, in general, many-to-many expressing the possibility of one element of $X$ mapping to many elements of $Y$, and many elements $X$ mapping to the same element of $Y$.

An example of a relation is $factors \in \mathbb{N}_1 \leftrightarrow \mathbb{N}_1$ the set of all mappings from non-zero natural numbers to their factors.

$$factors = \{ 1 \mapsto 1, 2 \mapsto 1, 2 \mapsto 2, 3 \mapsto 1, 3 \mapsto 3, 4 \mapsto 1, 4 \mapsto 2, 4 \mapsto 4, \ldots \}$$

If $r \in X \leftrightarrow Y$ then $r^{-1}$ is the inverse of $r$—written $r-$ in ASCII—in which all the maplets are reversed.

In many situations we want a relation that is at most many-to-one, that is a relation where elements of the domain map to at most one element in the range. This is called a functional mapping.

$X \mapsto Y$ is the set of all partial functions mapping from $X$ to $Y$. A function is partial if not all elements of the domain set, $X$, are required to have mappings.

$X \rightarrow Y$ is the set of all total functions mapping from $X$ to $Y$. A total function must map every element of $X$ to some element of $Y$.

$X \mapsto Y$ is the set of all partial injective functions from $X$ to $Y$. An injective function is one-to-one in which at most one element of $X$ maps to any particular element of $Y$.

Notice that, if $f \in X \mapsto Y$ then, in general, $f^{-1}$ is not a function, but if $f \in X \rightarrow Y$ then $f^{-1}$ is a function.

For functions we have the notation of function application, $f(x)$. If $f \in X \rightarrow Y$ then $\exists (x, y).(x \in X \wedge y \in Y \wedge x \mapsto y \in f \Rightarrow f(x) = y)$. What do we have for relations?

Since general relations are many-to-many, function application cannot work. Instead we have relational image. If $r \in X \leftrightarrow Y$ and $s \subseteq X$, then $r[s]$, the image of $s$ under $r$, is the subset of $Y$ to which the elements of $s$ are mapped under $r$.

$$factors[\{6\}] = \{1, 2, 3, 6\}$$

Note: $r[s] = \text{ran}(s \circ r)$