

Clayton School of Information Technology
FIT3013 Assignment 2 (ver 2)
Money, Money, Money!
Refined Refined Specification

Due Date: 12noon, 24 Sep 2008

This assignment is about developing the simple specification of a sum of money, started in Assignment 1. You are to use the RODIN tool to build the specification.

Recall that a sum of money is made up of a number of coins and notes. In Australian currency, there are 6 coins in circulation: 5 cents, 10 cents, 20 cents, 50 cents, 1 dollar, and 2 dollars. We now add notes to the model, whose denominations are 5, 10, 20, 50 and 100 dollars.

We wish to build a specification that does not have the same restrictions as assignment 1, that is:

- Any number of notes and/or coins are to be accepted in a transaction.
- There are two main events, *credit*, which adds a sum of money to the purse, and *debit* which removes a sum of money from the purse.

The assignment is to be approached as an exercise in refinement. To this end, consider the initial specification to deal simply with an arbitrary amount of money *amount*, with no details of the actual coins/note involved.

1. **Specify a context *CURRENCY* that defines the fact that Australian currency in circulation handles only amounts of money in multiples of 5 cents, and a machine *PURSE* that has two events, *credit* and *debit*, that add or remove a sum of money *amount* to/from the purse.**
2. **Refine the context *CURRENCY* to a new context *CURRENCY_R* that specifies a set *DENOMINATION* that defines the coins and notes in circulation. Refine machine *PURSE* to *PURSE_R* by refining *credit* and *debit* to handle handle an arbitrary *sum* of money given in coins and notes. *sum* should be specified as a total function from *DENOMINATION* to natural numbers.**

credit should add to the amount of money held in the *PURSE_R*, while *debit* should subtract from the amount of money in *PURSE_R*. Assume (guard) that the amount of money removed is already held in the *PURSE_R*. Elements of *DENOMINATION* cannot be interchanged - if two 5 cent coins are to be removed, it cannot be done as one 10 cent coin. Discharge the proof obligations.

3. **Develop a further refinement, *PURSE_R2*, that avoids the restriction in *PURSE_R* event *debit*, by removing coins or notes greater than the required amount, and setting a variable *changeRequired* to the amount (in cents only) that represents the difference. Define a new event *giveChange* that accepts an amount *change* in response to the *debit* of *PURSE_R2* that reduces the *changeRequired* to zero. What should happen if this condition is not met?**

Submit your solution as a fully commented Rodin export file through the Moodle submission mechanism. Include L^AT_EX generated pdf files of your project components, as described in lectures.