Soot Exercise 1
CS6235

The running example available here implements `DummyAnalysis` - an analysis that iterates over the `Unit`s of a method (i.e., its Jimple statements) in a naive order.

Recall the four kinds of definition statements presented as part of the Points-To Analysis module:

1. Allocation : `x = new ..()`
2. Copy : `x = y`
3. Load : `x = y.f`
4. Store : `y.f = x`

Observe that `DummyAnalysis` is currently equipped to recognise the **store** statements and output the variable being stored, the receiver of the dereference, and the field being accessed.

As an exercise, enhance `DummyAnalysis` to identify the other three definition statements, and for each kind of statement – output the following information to `STDIO`:

- Allocation : the LHS, and the type of object being allocated.
- Copy : the LHS and RHS
- Load : the LHS, the receiver and the field being read

Bonus

As an advanced exercise, print the control-flow successors and predecessors of each statement to `STDIO`. This exercise will help you think about program statements in a control-flow order which is essential for many of the analyses we deal with. Eventually, we will update `DummyAnalysis` to iterate over the statements of a method in control-flow order instead of a naive order.

Hint: refer to Section 5.7 in the Soot Survivor's Guide. Try to explore the classes it introduces and experiment before we visit it in our upcoming sessions.