CS350 Lab 3: Attributes

Change the definition of `which_cs350_lab` in `tiger.cc` so that it has the value 3.

In this lab, you will check for various errors that you didn't have to identify with your grammar. You should answer the following by creating boolean synthesized attributes. For each part, add a comment describing the attribute you define and explaining how it works, right before the `%union` in `tiger-grammar.y`, and then declare the attribute in the `%union` and add appropriate actions to your productions. If your grammar already detects the given error, just give a comment before the `%union` explaining this.

1. Detect programs in which a `break` statement is not enclosed in a `for` or `while`.

2. Detect illegal lvalues (i.e., values that are used on the left side of an assignment but should not be, such as `3 + 4 := x`).

3. Detect cases in which expressions other than identifiers are used where a type name is needed (such as `(3 + 4) [8] of 17`). You do not have to distinguish type names from other identifiers until Lab 5 (for example, you do not have to decide, during this lab, whether or not to give an error for `x[8] of 17` — this depends on whether `x` is, for example an integer variable (in which case this is illegal) or a type name (in which case it is legal)).

Your tiger compiler should now give error messages for programs with illegal lvalues, expressions used where type names are needed, or illegal breaks.

Record important design decisions for this lab in the file `Lab3-Attributes` in the `Design_Documents` subdirectory

For 75% partial credit, don't bother with anything related to arrays, records, or type declarations.