CS712/CS812 Project Phase 1 Spring 2016 Due Sunday February 14

Add features to the initial Tscript system, which is available on agate in *~cs712/public/tscript.tar*:

1. The Boolean type, including Boolean literals. Be sure to integrate the Boolean type with all the operators that are supported by the compiler.

2. The Undefined type. Be sure to integrate the Undefined type with all the operators that are supported by the compiler. Note that there is no Undefined literal. At program start-up, create "undefined" as a variable with the Undefined value as its value.

3. The Null type, including the Null literal. Be sure to integrate the Null type with all the operators that are supported by the compiler.

4. The logical not operator, the equals operator, the less than operator, and the greater than operator.

5. The String type, including basic support for String literals. For string literals, you do not need to support line continuations, and the only escape sequence you need to support is one consisting of the single escape character \n. You only need to support ASCII characters. (Note that there is an incomplete implementation of String literals in the distributed code.) Be sure to integrate the String type with all the operators that are supported by the compiler.

6. The subtraction operator, the division operator, and the unary minus operator. Provide optimized code generation for these operators when possible. To support this optimization, have semantic analysis determine the type of these operators, as is done for the addition and multiplication operators. Display this type in the tree dump.

7. Provide full support for numeric literals.

8. Allow multiple identifiers to be declared in one variable statement. Also support initializers. You will likely need to change how the symbol table operates, since it currently has the Identifier tree nodes point back to the VarStatement that declared the identifier.

You should be sure to complete the first four items in the above list because they will be used to test your Phase 2 and Phase 3 submissions. The remaining items in the list will not be used to test your Phase 2 and Phase 3 submissions.

Each item in the above list will be worth 10 points. However, if you successfully complete at least one item from the above list, then you will be given a bonus of 20 points.

There are some initial test files available on agate in *~cs712/public/tests/phase1*.

To get full credit, your code must be adequately documented and structured. If I can't easily read and understand your code, you may lose points.

You must give me back the system in the same form that I gave it to you. I must be able to install it and run it in the exact same way as when it was delivered to you. I would also like you to keep the source code organized in the same directory hierarchy. If you fail to do this, a significant deduction will be made to your grade.

Note that there is a script, *distribute.sh*, in *tscript/bin*, which will generate a tar file for you. This tar file is what I require you to submit to me for grading.

To turn in this assignment, type (on agate.cs.unh.edu):
~cs712/bin/submit phase1 tscript.tar

Submissions can be checked by typing (on agate.cs.unh.edu):
~cs712/bin/scheck phase1

The assignment is due on Sunday February 14. There is a grace period to 8am on Monday February 15 when no late penalty will be assigned. Submissions between 8am February 15 and 8am February 16 will have a late penalty of 10 points. Submissions between 8am February 16 and 8am February 17 will have a late penalty of 30 points. **No submissions will be accepted after 8am on Wednesday February 17.**

Remember: you are expected to do your own work on this assignment.

Last modified on January 24, 2016.

Comments and questions should be directed to pjh@cs.unh.edu