S3 Compiler Project

Copy your `S2.java` to `S3.java`. Then replace every occurrence of "S2" in `S3.java` with "S3".

Be sure your name in `S3.java` on line 10.

Modify `S3.java` so that it meets the specifications for S3 described below.

Compile your S3 compiler with

javac S3.java

Compile `S3.s` (which is in the J1 Software Package) with your S3 compiler with

java S3 S3

Assemble the output file `S3.a` created by your S3 compiler with

a S3.a

Finally run the executable program in `S3.e` created by the assembler with

e S3 /c

Submit S3.java, S3.a and S3.<*family name*>.log (the log file that the e program creates when it runs S3.e) in the form requested. If files requested, do not submit a ZIP FILE.

SPECIFICATIONS FOR S3

```
    S3 is the S2 compiler with the following extensions:

1) The println allows zero arguments.

2) Both println and print can take a string argument, as well as an
   expression.

3) Unary plus and minus are fully supported.  In S2, unary plus and minus are
   allowed only on constants.  In S3, they are also allowed on variables and
   parenthesized expressions.

4) Cascaded assignments are allowed.  For example, in S3, the following
   statement is legal:

       x = y = z = 7;

    Support this statement according recursive productions that require a
    look ahead as described in Section 14.3.

5) The readint statement is supported.  It reads in a single integer from the
   keyboard, converts it to binary, and places the binary value in the
   variable
   specified in the readint statement.  For example,
```

```
     readint(x);
```

 reads an integer into x.

6) The token trace is optional. S3 (the hand-written version) should
   generate a token trace only if the -trace_tokens argument is specified
   after the input file name on the command line when the S3 compiler is
   invoked. The S3j compiler(the JavaCC version of S3) should generate a
   token trace only if the S3j compiler is created with a javacc command in
   which you specify the -common_token_action argument as the first argument.

7) The assembly language file generated by the compiler should contain the
   source code as comments.