

**CS520—Spring 2013—Homework 5**  
**Wednesday February 27**

**Question 1**

Why does the maTe assembler have two passes? What is done on the first pass? What is done on the second pass?

**Question 2**

Assemble the follow sequence of maTe instructions. Show the result as a series of 32-bit words in hex. Assume the first instruction is at address 540 (decimal).

```
    aconst_null
    astore 0
    aconst_null
    astore 1
    newint 0
    astore 0
    newint 1
    astore 1
W.0:
    aload 0
    newint 13
    invokevirtual 14 2
    ifeq $W.1
    aload 0
    newint 1
    invokevirtual 4 2
    astore 0
    aload 0
    aload 1
    invokevirtual 8 2
    astore 1
    goto $W.0
W.1:
    newint 0
    areturn
```

**Question 3**

Why do you think the Java Virtual Machine has a *istore* instruction as well as *istore\_0*, *istore\_1*, *istore\_2* and *istore\_3* instructions? What do these instructions do?

**Question 4**

Disassemble the following bytecode for a Java method:

```
1A 05 68 04 60 AC
```

**Question 5**

What does the method of Question 4 compute?