50 points possible. Open notes, text, calculator. No Internet Access.

NAME: ____________________________________________________________

1. (15 pts) Show the signals that would be necessary to execute all phases (except FETCH) of the following instruction: **STI R2, VAR1.** (There is a diagram of the LC-3 control path and a list of available signals on the last page of the exam.)
2. (35 pts) Consider the following C program fragment:

```c
int main() {
    int a = 12, b = 29, c = -1;
    c = mysub(c, a, b);
    ... etc ...
}

int mysub(int w, int x, int y) {
    int d, e;
    ... more code ...
    d = e + y;
    return d;
}
```

a. (10 pts) Fill in the symbol table that the compiler will generate for the subroutine `mysub`:

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Offset</th>
</tr>
</thead>
<tbody>
<tr>
<td>R6</td>
<td>x6FFD</td>
</tr>
<tr>
<td>R5</td>
<td>x6FFF</td>
</tr>
</tbody>
</table>

b. (10 pts) Above and to the right is an illustration of the stack as it would appear during the execution of `main`. Fill in the rest of the stack as it will appear while executing the region labeled “more code” in `mysub`. Show the new locations of R5 and R6. Label each stack location. **Wherever possible use exact hex values.**

c. (10 pts) Show the assembly language code that will be generated for the line `d = e + y;` in `mysub`. Use R0 and R1 for temporary storage, if necessary.

d. (5 pts) Show the assembly language code that will be generated to store the return value and clean up the stack in `main`, just after returning from `mysub`. 
LC-3 Control signals

SR1, SR2, DR: 3-bit register number

LD.XXX: 1 = load from bus, 0 = do not load

ALUK = ADD = 00, AND = 01, NOT = 10

xxxMUX = inputs are numbered in binary, from left (lowest) to right (highest)

MEM.EN = 0, 1

R.W: R = 0, W = 1

GateXXX: 0 = do nothing, 1 = load data onto the bus