C.3 State machines

C.3.1 Background

No exercises here.

C.3.2 In-lab section

1.
```
>> d = {'a' 'b' 'a' 'a' 'b'};
>> n = 0;
>> for e = d
      if strcmp(e, 'a') n = n+1;
      end
   end
>> n

n =

     3
```

Note how the for loop works. The variable `e` gets each element of `d` each time through the loop. The function definition looks like this:

```
function result = count(arg)
% COUNT - Count the number of occurrences of 'a' in the argument.
result = 0;
for e = arg
   if strcmp(e, 'a') result = result + 1;
   end
end
```

Then

```
count(['a', 'b', 'c', 'a', 'aa'])

ans =

     4
```

and

```
>> count({'a', 'b', 'c', 'a', 'aa'})

ans =
```

```
     2
```

Notice the difference between these. In the first instance, an array is the argument with one character forming each element of the array. In the second instance, a cell array is the argument with a string forming each element. Thus, the result of the count is different in the two cases.

2. The following program does the job:

```
% COUNTAS - Count the number of a's in the input.
while 1
    response = input('Enter a string:','s');
    if (strcmp(response,'quit') | ...
            strcmp(response,'exit'))
        break;
    end
    disp(count(response));
end
```

The 's' argument to the `input` function indicates that the input should be interpreted as a string, rather than being evaluated as a Matlab expression. The "..." indicates to Matlab that the line is continued.

3. Here is a definition of the update function:

```
function [next, output] = update(state, input)
% UPDATE - simple state machine update.
% The state argument should be 0 or 1.
% The input argument should be '0', '1', or 'absent'.
% The output result is '0', '1', or 'absent'.
% The next result is 0 or 1.
switch input
case '0'
    next = 0;
    if state output = '1';
    else output = '0';
    end
case '1'
    next = 1;
    if state output = '1';
    else output = '0';
    end
otherwise
    next = state;
    output = 'absent';
end
```

Notice that any state that is non-zero is interpreted as 1. Any input that is not 0 or 1 is interpreted as absent. Here is a program that executes this state machine in response to user input:

```
% DELAY - invoke the update function repeatedly.
state = 0;
while 1
    in = input('Enter 0, 1, absent, or quit: ', 's');
    if (strcmp(in, 'quit') | strcmp(in, 'exit'))
        break;
    end
    [state, output] = update(state, in);
    disp(output);
end
```