APSC 380 : INTRODUCTION TO MICROCOMPUTERS
1997/98 WINTER SESSION TERM 1

Assignment 7 - State Machines in C

due Friday, November 7

Question 1

Design a state machine to control the rear exit door on a bus. To open the door a passenger has to step on a sensor, wait until the door is unlocked by the controller and push the door open. As a safety feature the controller turns on the brakes to make sure the bus will not start moving while a passenger is in the process of getting off the bus.

The controller has two outputs: a signal to turn on the brakes and a signal to lock the door. The controller has three inputs: a sensor that detects if the bus is moving, a sensor that detects if someone is waiting to exit through the door, and an sensor that detects if the door is open.

The controller should unlock the door only when the bus is stopped and a person is waiting to exit. Whenever the door is unlocked the brakes should be turned on. If the door is unlocked, it should be locked as soon as no more people are waiting to exit. It is possible for the door lock to be turned on while the door is open so so the brakes should never be turned off while the door is open.

List the inputs and outputs (use the names of the functions listed below), assign state names, give the output values for each state, show a state transition diagram and table.

Write a C program that implements the state machine. The program should get its input by using three integer functions declared as: `int moving(void) int person(void)` and `int dooropen(void)` that return 1 or 0 and do its output using two functions declared as `void lock(int on)` and `void brakes(int on)`.

Question 2

Design a state machine to detect what part of a sequence of input characters is part of a valid floating point number. The input to the state machine is a sequence of characters and the output is a sequence of characters: `y` if all of the character(s) seen so far are part of valid number, and `n` if an invalid character has been seen in the input.

A valid floating point number consists of the following:

- an optional sign (+ or -)
- an optional integer part consisting of one or more digits (0-9)
- a decimal point
- an optional fractional part consisting of one or more digits (0-9)
- an optional exponent part consisting of the letter e or E followed by an optional sign and one or two digits

For example, “+12.3E-33”, “0.”, and “-.” are valid floating point numbers, but “0”, “1.0E120”, “0x00” and “E3” are not.

List the inputs and outputs (use the names of the functions listed below), assign state names, give the output values for each state, show a state transition diagram and table. In the state transition table you may give a list of input characters or their complements for a transition rule.

Write a C program that implements the state machine. At each iteration the program should read one input character by using the function `char getche(void)` and output one character using the function `int putchar(int c)`.