1 Administrivia

- Project 1 out, due September 30 and October 4
- Lab 1 on Friday

2 State Machines Overview

State machines, or more formally finite state machines (FSMs), are a great tool to use to model the behavior of a system over time as it responds to various inputs. If correctly designed, a state machine is both general and non-ambiguous, and can provide valuable insight about how a system behaves. A state machine, in order to be fully defined, must have:

**Input alphabet** A well defined set of possible inputs

**Output alphabet** A well defined set of possible outputs

**States** A (finite) set of states

**Transitions** Rules that define how to move from state to state for each possible input and state combination

**Start state** The state that the machine will be in initially

Some of the reasons for using the state machine abstraction include:

- Being able to reason about your code behavior, both correct behavior and failure conditions.
- Providing a design language that directly translates to code (sometimes)
- Communicating with others effectively about your code’s behavior
- Encapsulating complex behavior into simple, parallel machines

3 Designing State Machines

One approach to designing a state machine:

1. Determine your initial state
2. Determine the set of inputs you will receive
3. For each possible input, how should each state respond?
   a. If a new state makes sense, create it
   b. Repeat until all states have transitions for all possible inputs

You can also start with all of the states already defined ahead of time if you know for sure what states you want to have.

4 Implementing State Machines

- Model the state machine as an instance of a class
- The state of the machine can be captured in the instance variables
- Transitions are handled by instance methods
- Not a direct mapping, in general
5 Examples

Figure 1: A simple button switch state machine

Figure 2: A turnstile state machine

Figure 3: A robot behavior for scoring balls in goals

Figure 4: A state machine representing ordering food at a cafeteria