Programmer Manual

Programmer Manual

1. Problem Description

This program simulates a push button lock with five available inputs, A, B, C, D, or E. A transition table and an action table are populated using the Table ADT (for details see the Table programmer manual). These tables represent a finite state machine with states describing the behavior of the lock.

2. Data Types and Classes

The data types used in this program fall into two categories: predefined data types and programmer defined data types. The following subsections address the data types used.

A. Predefined Data Types

2.1 int

Variables:

tableSize	the amount of data the table can hold

2.2 char

Variables:

letter	the character entered into the lock

2.3 string

Variables:

state	the state the program is currently in

B. Programmer Defined Data Types

2.1 Pair

This class has:

Data members:	T1 first
	T1 second
Member functions:	Pair
	operator=
	operator<
	operator==
	makePair

2.2 Mapping

This struct has:

Data members:	char letter
	string state
Member function:	map

Variables:

the_table	array of pairs used to hold the table data

2.3 Table

This class has:

Data members:	int tableSize
	Pair* the_table
Member functions:	Table
	print
	insert
	remove
	lookup
	operator=
	empty
	full
	size
	isIn
	loadTable
	lock

Variables:

transition	transition table
action	action table

3. High Level Program Solution

Main Program
Get the transition table file name
Load the transition table data into a table
Get the action table file name
Load the action table data into a table
Run the lock simulation using the transition and action tables (see the Table programmer for more details)

4. Limitations and Suggestions

This current program only accepts characters A, B, C, D, and E. A larger set of key combinations could be used to provide more security. Similarly, there is only a four character password. The password could be set by the user and the program could dynamically determine the amount of characters needed to be entered.