Problem statement

A company is designing a new vending machine. The company wants you to develop an object-oriented software simulator so that they can see whether the machine that they are developing will meet the customer's and service people's needs.

The company's vending machine is similar to many of the candy vending machines at UNBC, but smaller. There are ten rows and eight columns of spring coils that can hold chips, candy, and other merchandise. Each spring coil can hold up to 10 items.

The machine operates in two modes, depending on whether or not the front door is open. When the door is shut it operates in vending mode. Customers may enter 5¢, 10¢, 25¢, $1.00, and $2.00 coins[1]. They may also press buttons labelled "A" through "J" to select the row, and "1" through "8" to select the column of the item they wish to purchase. They may also push the coin return button at any time.

In vending mode, the machine normally responds to coin entries by displaying the current total entered on a small display. When the machine's customers push one of the buttons their choice is displayed instead. When a customer completes a choice one of three things happens. Either the machine dispenses the requested item and releases the customer's change through the coin return; or it displays a message saying that the requested selection isn't available; or it displays a message saying that the customer hasn't entered enough money to purchase the item. When a customer presses the coin return button any unspent money in the machine is returned through the coin return.

When the door is open, the machine operates in restock mode and none of the normal vending functions are available. In this mode it is possible:

- to set the price for each of the spring coils;

- to refill or partially refill each of the spring coils;

- to empty the cash box;

- to refill or partially refill the change box; and

- to close the door.

The coin-handling mechanism of the vending machine consists of a coin entry slot; a coin return slot; a cash box; and a change box that has a column for each kind of coin. When a coin is put in the entry slot the machine can control whether it goes back out the coin return slot or gets dropped into the cash box. The machine can also direct the change box to drop a coin from a particular column into the coin return slot. Note that once coins enter the cash box the machine cannot move them elsewhere. When the change box is close to empty the machine can display a message asking the user to enter exact change only.

Your simulation should allow the simulation user either to directly control the events that happen or enter an automatic mode where customers and machine restockers arrive randomly and use/restock the machine. It should be able to help answer questions such as how much change needs to be in the change box, so that most of the time the machine runs out of stock before it runs out of change. In order to do this your simulation needs to keeps statistics as it runs and print them out on request, and needs to use random number generators to simulate customer behaviour.

---

[1] the vending machine rejects 50¢ coins