MCP #7

TALK

- Expected actions for user?
  1. select item and get price
  2. accept bills/coins
  3. dispense items purchased and return change
  4. refund when cancelling request
- Exceptions?
  1. Sold out item
  2. not enough $ inserted
  3. no change

EXAMPLE

Test Code	Behavior
VendingMachine vl = new VendingMachine()	construct new vending machine
Inventory i1 = new Inventory()	construct new inventory
Item item1 = new Item()	construct new item

BRUTE FORCE

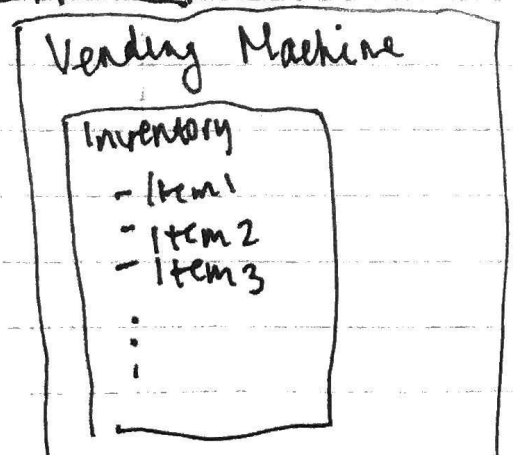
3 classes

Vending Machine	Inventory	Item
State - balance ($ inserted) - amount of change available - # of transactions Behavior - getters and setters for each state variable - insert $ ~~- get change for bought item~~ - refund/cancel request - purchase item	State - hashmap of items (key = itemId value = item) Behavior - getters and setters for each state variable - delete item	State - cost of item - name of item - id # - # of items Behavior - getters + setters - subtract item

Optimize

- can't think of how to optimize...

WALK THROUGH



Vending Machine has 1 inventory which contains the multiple items and their state

IMPLEMENT

```
public class VendingMachine {
    double balance;
    double changeAvailable;
    int numOfTransactions;
    Inventory inventory;
    public VendingMachine(double changeAvailable) {
        this.changeAvailable = changeAvailable; this.inventory = new Inventory();
    }

    public void insertMoney(double money) {
        balance += money
    }
    public void purchaseItem(int itemID) {
        checkBalance();
        inventory.deleteItem(itemID);
    }
    public void cancelRequest() {
        balance = 0;
    }
}
```

```
public class Inventory {
    HashMap items;

    public Inventory () {
        // create default inventory for time being
    }
    public void deleteItem ( int itemId ) {
        items.get(itemId). subtract ();
    }
}

public class Item {
    double cost;
    String name;
    private int itemCount;
    int id;

    public Item (double cost, String name, int itemCount, int id) {
        // set all these variables
    }
    public void subtract () { itemCount--; }
}
```

TEST

test each use case and exception