Your Turn!

YourTurn1 (part a): HomePhone

HomePhone
-phoneNumber : int
+HomePhone(phoneNumber : int)
+makeCall(numberToCall : int) : void

prints → **HomePhone is dialing 5685844**

YourTurn1 (part b): PayPhone

PayPhone

-moneyAccepted : int

+acceptCoin(coin : Coin) : void

+makeCall(number : int) : void

<<enumeration>>
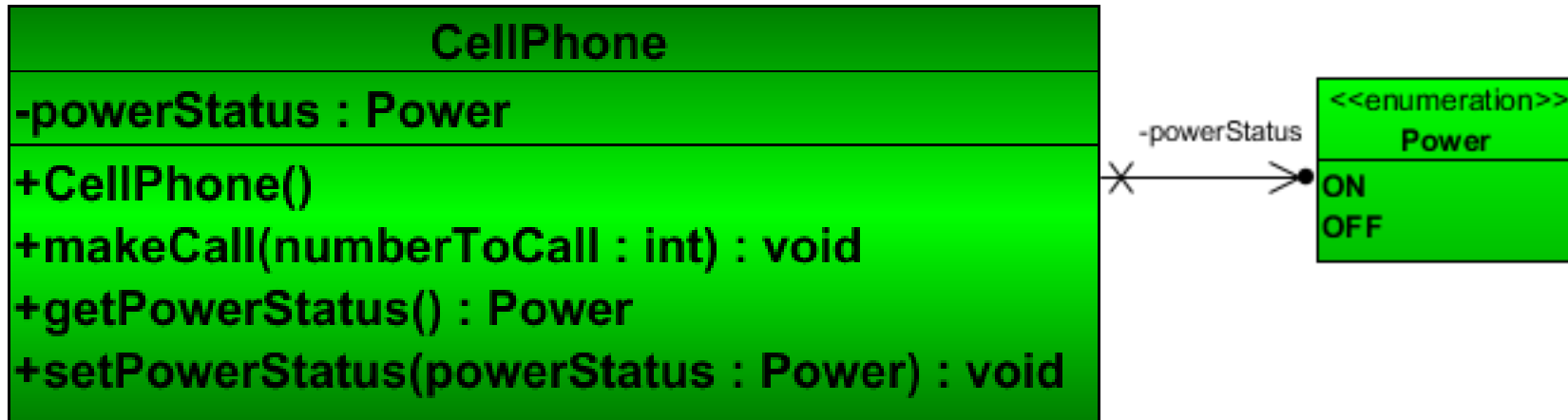**Coin**

NICKEL
DIME
QUARTER
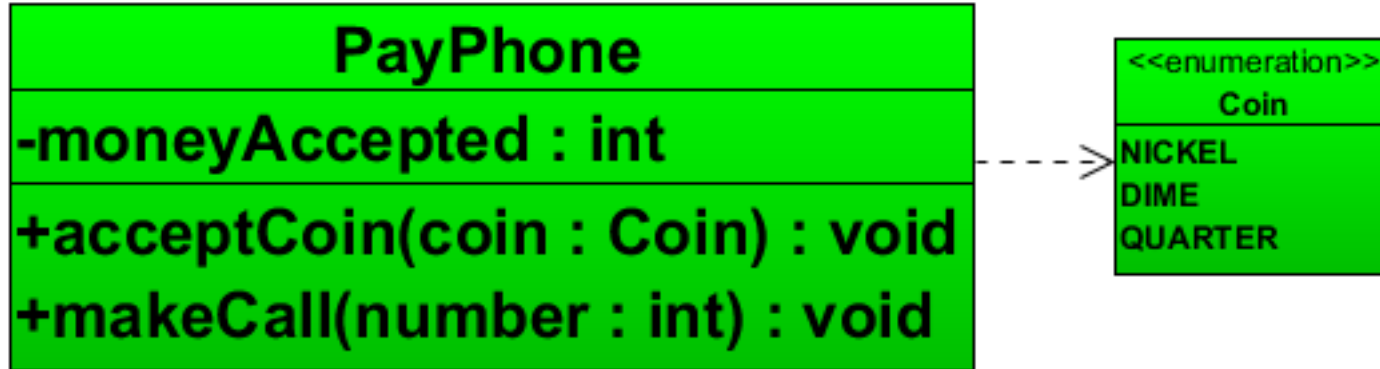
- A PayPhone can accept either nickels (5 cents), dimes (10 cents), or quarters (25 cents)

- Can make calls (accepts the number to call) if at least 25 cents have been deposited into it)

PayPhone is dialing 5685844

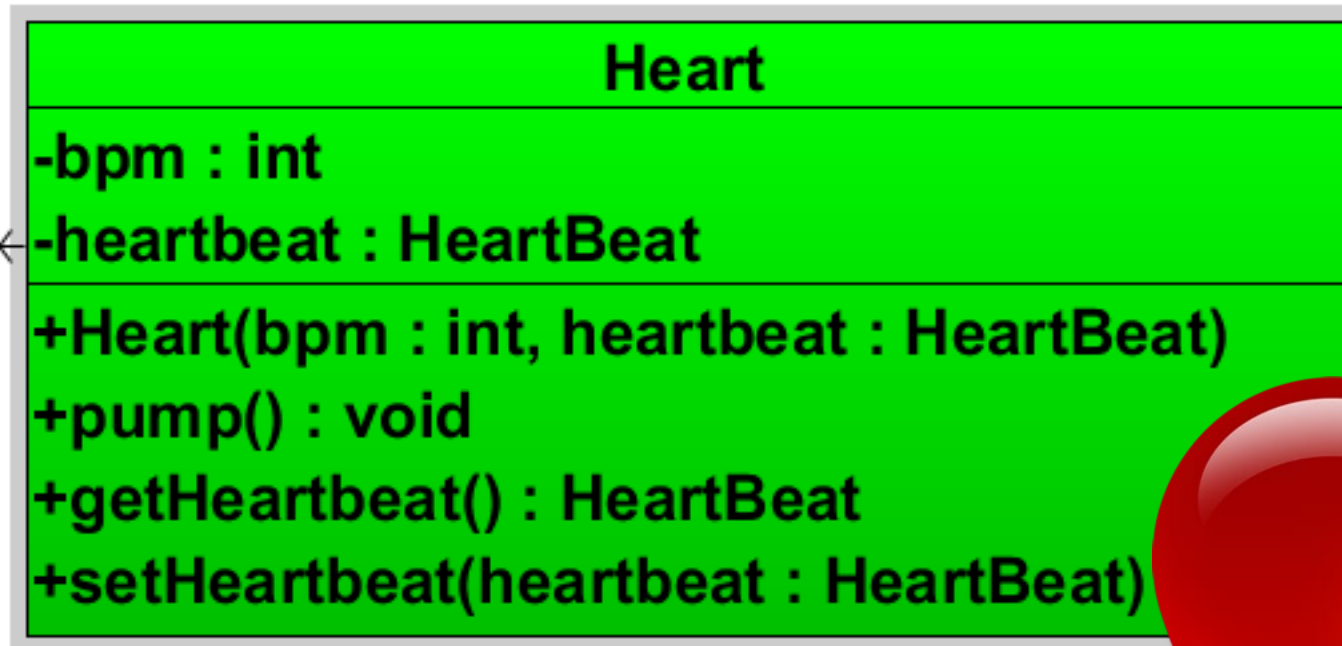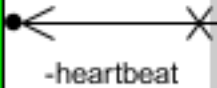OR Sorry, not enough change inserted

YourTurn1 (part c): CellPhone

CellPhone
-powerStatus : Power
+CellPhone() +makeCall(numberToCall : int) : void +getPowerStatus() : Power +setPowerStatus(powerStatus : Power) : void

-powerStatus

<<enumeration>> Power
ON OFF

CellPhones can't make a call unless their powerStatus is ON
CellPhones should be turned off when they are first instantiated

CellPhone is dialing 5685844

OR Sorry, CellPhone power is turned off

YourTurn1 (part d): Heart

<<enumeration>>
HeartBeat

PITTERPATTER
THUMPITYTHUMP
BOOMCHICKA
FLIPITYFLOP

-heartbeat

Heart

-bpm : int
-heartbeat : HeartBeat

+Heart(bpm : int, heartbeat : HeartBeat)
+pump() : void
+getHeartbeat() : HeartBeat
+setHeartbeat(heartbeat : HeartBeat)

pump will print the heartbeat out as many times as bpm is set to

PITTERPATTER..PITTERPATTER..PITTERPATTER

Yourturn1 (part e)

- Write a Start Class to test all of your Classes

- Add the code to make an array of 3 hearts (use the shortcut method) and have each heart pump for one minute

- Draw the memory of the jvm at the point immediately after the array was created.