10. (6 pts) Consider the following `world.Mystery` function.

A) What does `world.Mystery` return when the following call is made?

```
world.mystery value1 = 12  value2 = 7  value3 = 20
```

B) What does `world.Mystery` return when the following call is made?

```
world.mystery value1 = 3  value2 = 4  value3 = 2
```

C) Give arguments to the call to Mystery that will cause the number 8 to be returned.

value1 =

value2 =

value3 =
11. (10 pts) Consider the following Alice world that has three objects: goalpost, basketball and Plato.

The world starts as shown in the top figure. Write the Alice code to do the following steps in order.

a) Have the ball stay on Plato’s hand when Plato moves in parts b), c) and d).

b) Have Plato (and thus the ball) turn to face the goalpost.

c) Have Plato (and thus the ball) move to 5 meters in front of the goal posts.

d) Plato (and thus the ball) should jump up to the goal posts (it is about 40 percent of the height of the goal post), and turn around to face front still holding the ball. (as shown in the top right figure).

e) Finally, the ball should drop to the ground. (Plato should stay where he is)

12. (4 pts) Complete the following function called IsConeThisColor that has two parameters, a color named someColor and an object named object, and returns true if the cone is the color passed to the function. It returns false otherwise.
For example, in the picture above, the leftmost cone, Cone2, is red. The function call 
IsConeThisColor someColor=RED object=Cone2 
returns true.

13. (15 pts) Consider that an Alice world has several rectangular objects and that we 
want to be able to calculate the perimeter of the front of the objects. The perimeter 
is defined as twice the sum of the height and width of an object. For example in 
the figure below, suppose the pictureFrame (the object on the rightside in the 
figure) has dimensions of 1.5 meter in height and 1 meter in width. The perimeter 
of the pictureFrame would be 5, obtained by (2 * (1.5 +1)).

A) (5 pts) Complete the following function called Perimeter that returns the 
perimeter of an object. This function has one object parameter named item and 
returns the perimeter of item.
B) (8 pts) Complete the following function called 
ObjectWithSmallestPerimeter that returns an object. This function has three 
object parameters named object1, object2 and object3, and returns the object 
that has the smallest perimeter. You can assume that the three objects have different 
perimeters. You MUST call the function Perimeter that you wrote in Part A. You may 
assume that Perimeter works correctly, regardless of what you wrote in Part A.

C) (2pts) Assume an Alice world has the three objects toyBox2, pictureFrame, 
and bookshelf. Give the code to make the object with the smallest perimeter say “I 
have the smallest perimeter”. You MUST call the function 
ObjectWithSmallestPerimeter that you wrote in Part B. You may assume that 
this function works correctly, regardless of what you wrote in Part B.

14. (8 pts) Complete the following class method called hopAndFlip whose header 
is shown below. This method has one number parameter named howHigh. This 
method has the bunny move up the distance “howHigh” and then move down the 
distance ”howHigh”. If howHigh is greater than or equal to twice the bunny’s 
height, then the bunny does two flips while in the air. If howHigh is less than 
twice the bunny’s height, but greater than or equal to the bunny’s height, then the 
bunny does one flip while in the air. Otherwise the bunny does no flips while in 
the air.

For example, if the bunny is 0.6 meter tall, then in the first call 
below the bunny does no flip, in the second call the bunny does two 
flips and in the third call the bunny does one flip. In all three cases 
the bunny moves up then flips if he is suppose to, then moves back 
to the ground.

```
bunny.hopAndFlip howHigh = 0.4
bunny.hopAndFlip howHigh = 2
bunny.hopAndFlip howHigh = 1
```
10. (6 pts) Consider the following `world.Mystery` function.

A) What does `world.Mystery` return when the following call is made?

```
world.mystery num1 = 1  num2 = 2  num3 = 5
```

B) What does `world.Mystery` return when the following call is made?

```
world.mystery num1 = 4  num2 = 5  num3 = 3
```
11. (10 pts) Consider the following Alice world that has three objects: Chicken, lawnmower and tennisNet.

The world starts as shown in the top figure. Write the Alice code to do the following steps in order.

a) The chicken should face the lawnmower, and move over to it stopping 0.5 meters in front of it. Then have the chicken jump on top of the mower (the red part of the mower is .35m tall) and turn to face the same direction as the mower (as shown in the bottom left figure).

b) The chicken and mower should both face the tennis net.

c) The mower should move toward the net giving the chicken a ride (Give commands to move only the mower! the chicken should move along with the mower), stopping 1 meter from the tennis net and then jump over the net (clearing the net cleanly), landing on the other side of the net (as shown in the bottom right figure above).
12. (8 pts) Complete the following method called `PlaceInMiddle` whose header is shown below. This method has three object parameters named `person`, `structure1` and `structure2`. This method makes `person` invisible, moves `person` to exactly halfway between `structure1` and `structure2`, has `person` facing `structure1` and then makes `person` visible again.

For example, if the call to `PlaceInMiddle` shown below is made with the picture shown on the left above, then `skaterGirl` disappears, and reappears facing the `gazebo` and between the `gazebo` and `windmill` as shown in the picture on the right above.

```
world.PlaceInMiddle person = skaterGirl  structure1 = gazebo  structure2 = windmill
```

13. (33 pts)
A) (5 pts) Complete the following function called `ShorterObject` that returns an object. This function has two object parameters named `object1` and `object2`, and returns the object that is shorter in height. You can assume that one of the objects is shorter than the other one.

```
world.ShorterObject object1, object2
```

B) (8 pts) Complete the following function called `ShorterOfThree` that returns an object. This function has three object parameters named `item1`, `item2` and `item3`, and returns the object that is shorter in height. You can assume that the three objects have different heights. You may call the function `ShorterObject` that you wrote in Part A if you want to. You may assume that `ShorterObject` works correctly, regardless of what you wrote in Part A.
C) (8 pts) Complete the following function called `isOnlyOneThisColor` that returns a boolean. This function has four parameters: a color parameter named `objectColor` and three object parameters named `obj1`, `obj2` and `obj3`, and returns true if only one of the three objects is the color `objectColor`, otherwise it returns false. You can assume that at least one of the three objects has the color `objectColor`.

D) (12 pts) Complete the following method called `ChangeShortestRedToBlue`. This method has three object parameters named `shape1`, `shape2` and `shape3`, and changes the shortest red object to the color blue. You can assume that at least one of the three objects has the color red. For full credit you MUST call at least two of the functions `ShorterObject`, `ShorterOfThree` and `IsOnlyOneThisColor` that you wrote in Parts A, B and C. You may assume that those functions work correctly, regardless of what you wrote in Parts A, B and C.

Note the following. If the shortest object is red, you can just change it to blue. If only one object is red, you can change it to blue. If only two objects are red, you must figure out which of the red objects is the shortest and then change the shortest red object to blue.