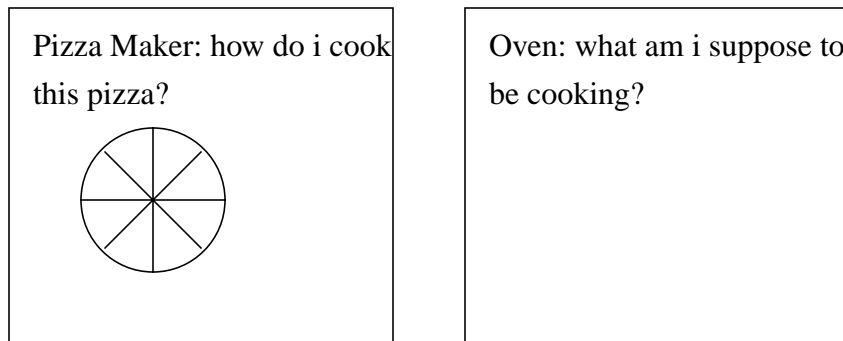


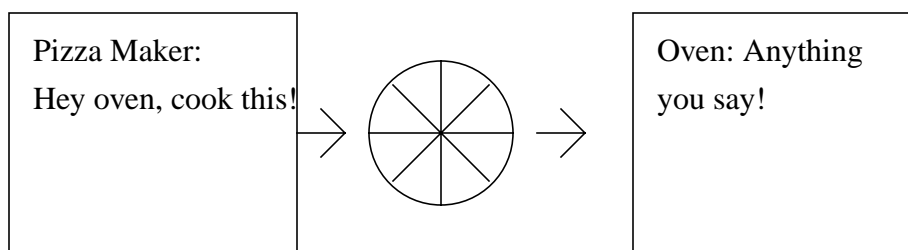
Parameters: The Mini-Lab

Goal: To teach how to pass parameters correctly. Every program from now on will require you to make methods that take certain parameters, so it's time you learn how to do it!

Why parameters: Parameters are necessary for much programming because they allow you to have many different objects that can effectively share data. Suppose we have a pizza maker object and an oven object. We want the oven to cook our pizza, but we made our pizza in our pizza maker class. How can we tell the oven which pizza to cook? Simple, make the pizza be a parameter.



Solution: Send the pizza as a parameter.



Result: A freshly cooked pizza!

What's that look like in java?

Glad you asked....

```

public class PizzaMaker extends GP.Containers.Applet {
    /**We need a reference to an oven to cook things.**/
    private Oven _oven;

    public void PizzaMaker() {
        /**We need to instantiate a new oven.**/
        _oven = new Oven();
        /**We want to make a new pizza, we pass the toppings in as parameters.**/
        Pizza ourPizza = new Pizza(cheese, pepperoni, mushrooms);
        /**Now we tell the oven to cook the pizza we just made. The oven knows which
        *pizza to cook because we send it ourPizza as a parameter. In this case, the parameter
        *is the pizza we want cooked. **/
        _oven.cookPizza(ourPizza);
    }
}

```

```

public class Oven extends Dominoes.Interactive.Oven {

    public void Oven() {
        /**You have to preheat the oven.**/
        this.heatUpOven();
    }

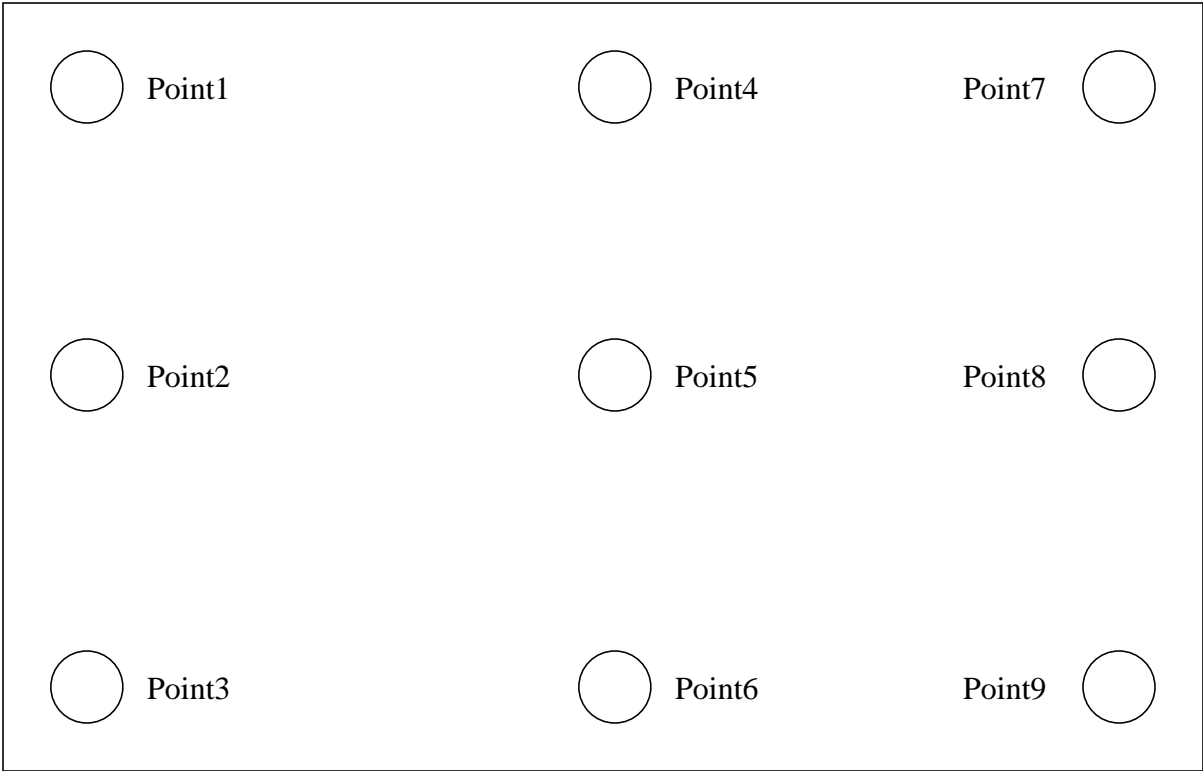
    /**
    * This method takes a pizza as a parameter. The setup is as follows
    * (<type of object your passing in> <the local variable name you want it to be called>
    */

    public void cookPizza(Pizza pizzaToCook) {
        /**So we passed in something of type Pizza and called it pizzaToCook.**/
        /**Now we can use the oven's cook method to cook the pizza.**/
        this.cook(pizzaToCook);
    }
}

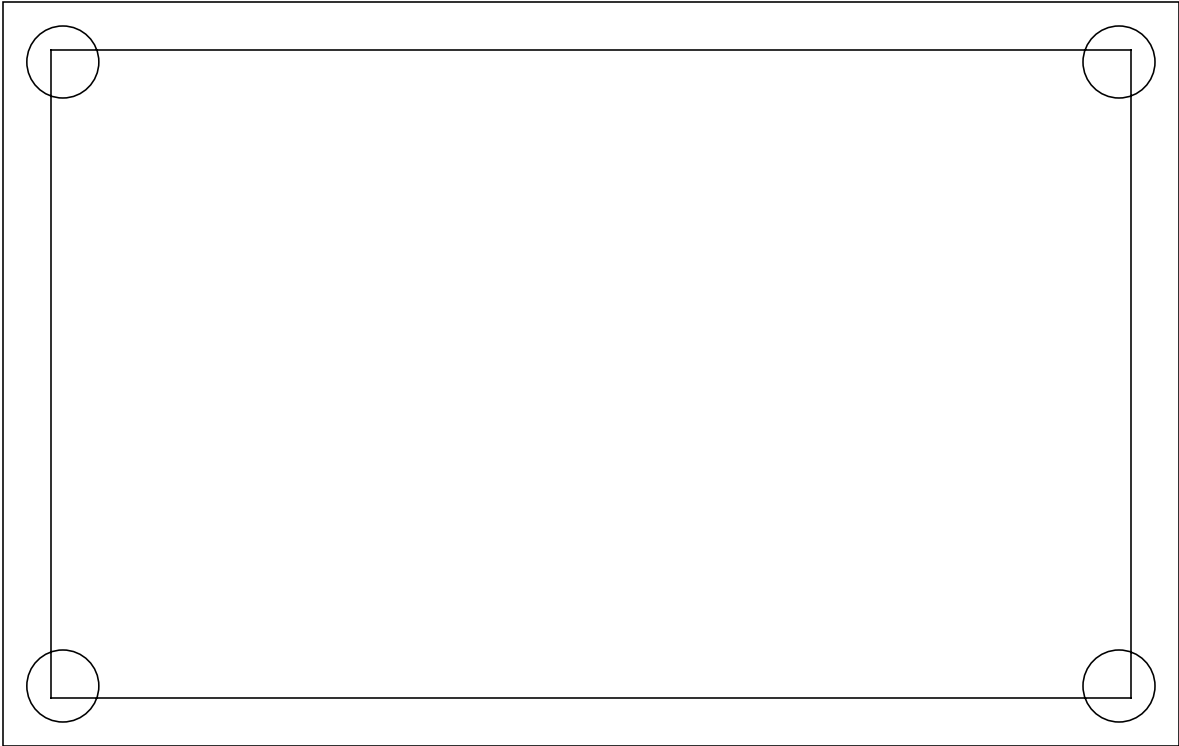
```

And now the lab:

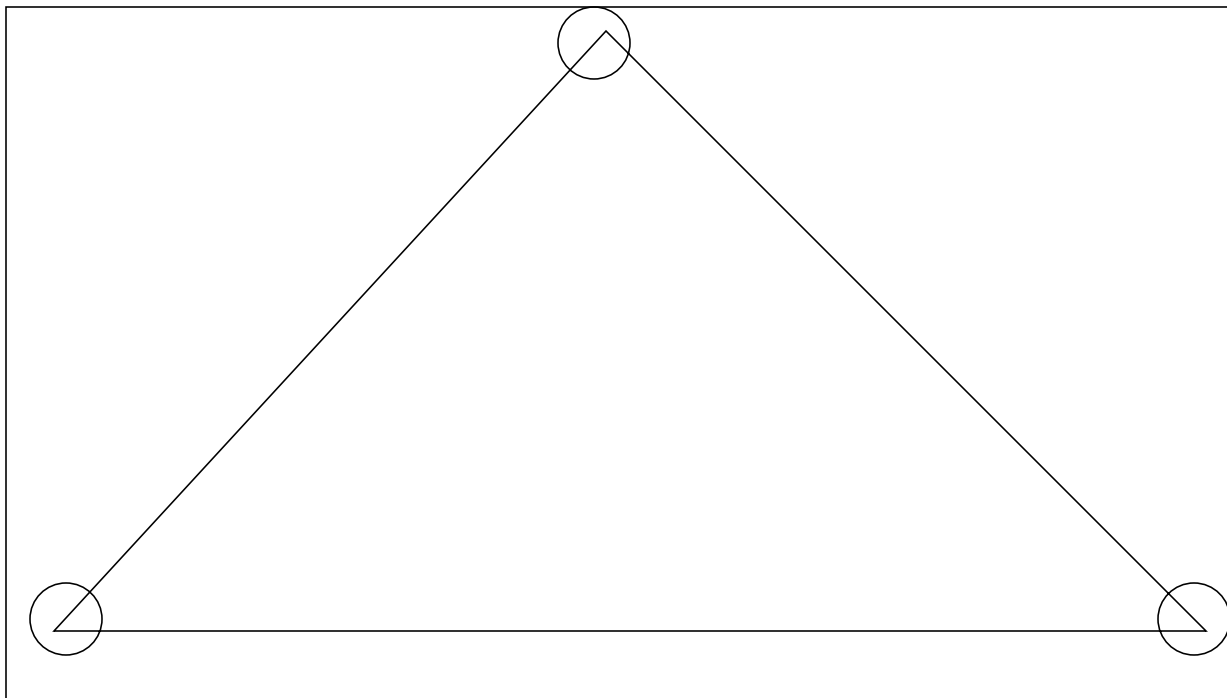
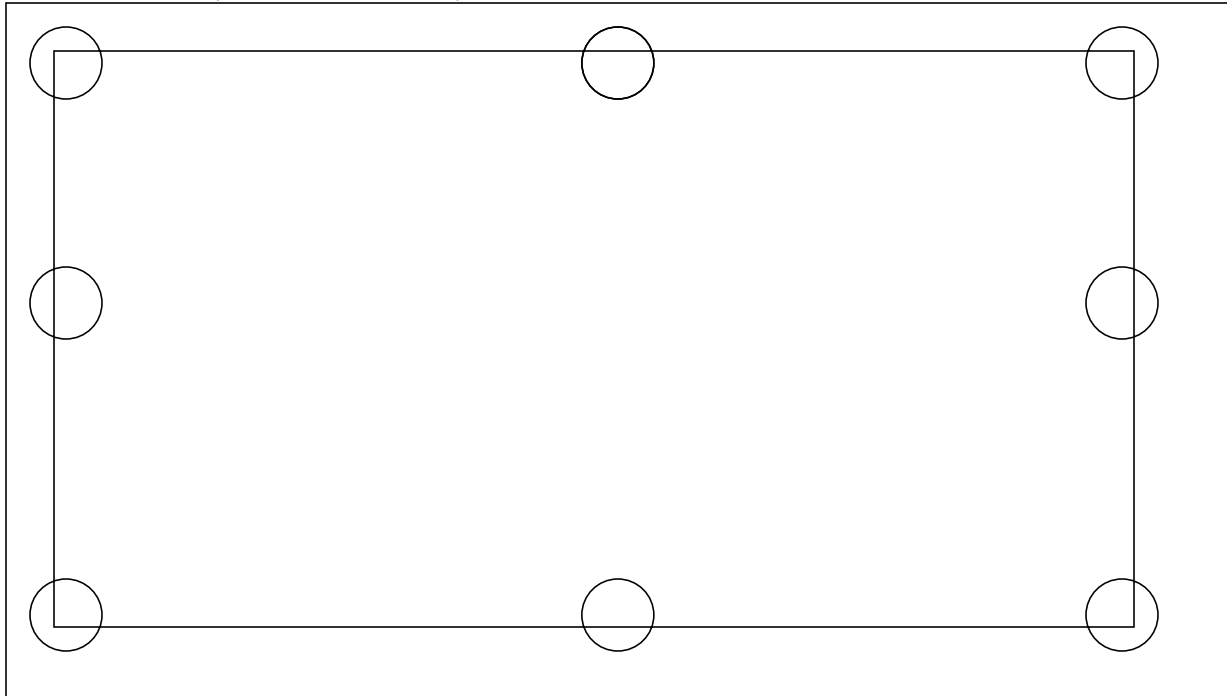
In this lab you are given a layout that has 9 points. Your job is to use the Machine object to draw the following shapes. In order to help you, the machine object has a Draw method that you can call. To use the Draw method you must send it 2 of the 9 points you are given, and it will connect these two points for you. Each shape will require multiple method calls, but once you get the hang of it, it's pretty easy. Take a look at the next page to see how the points are laid out and an example of how you would use the method.

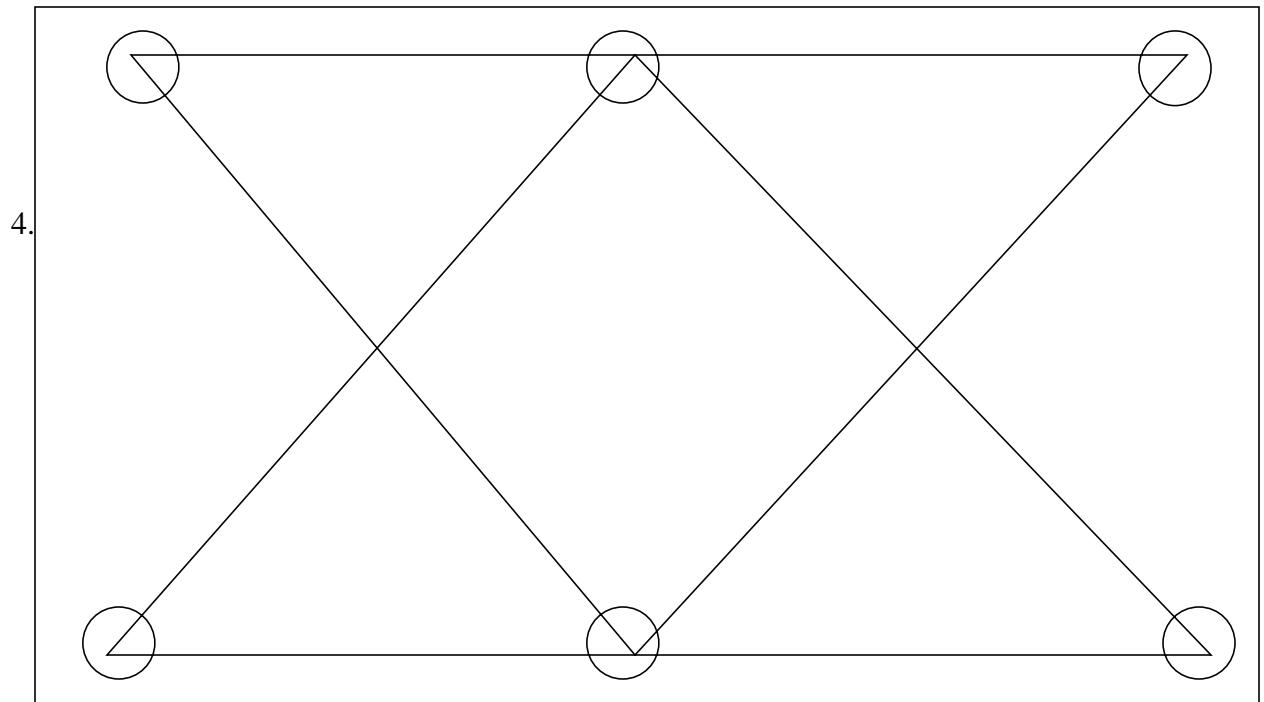
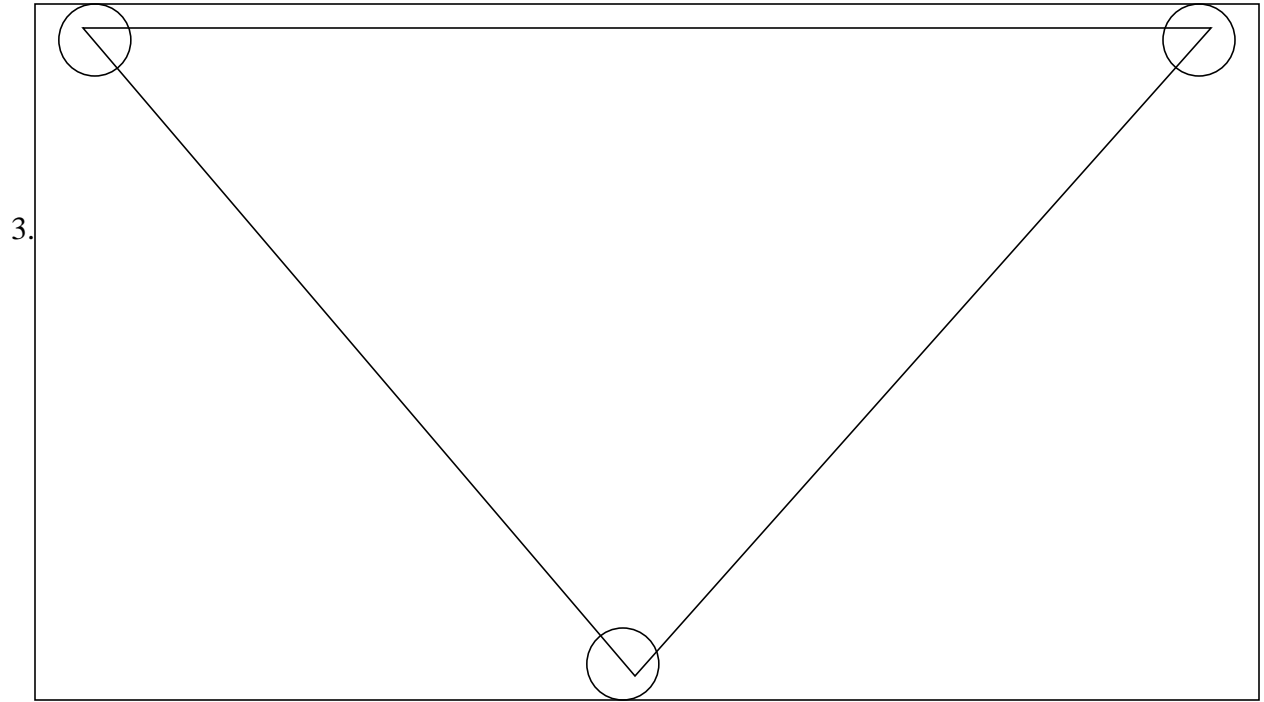


Exercise 1: Draw a square using the four corner points.

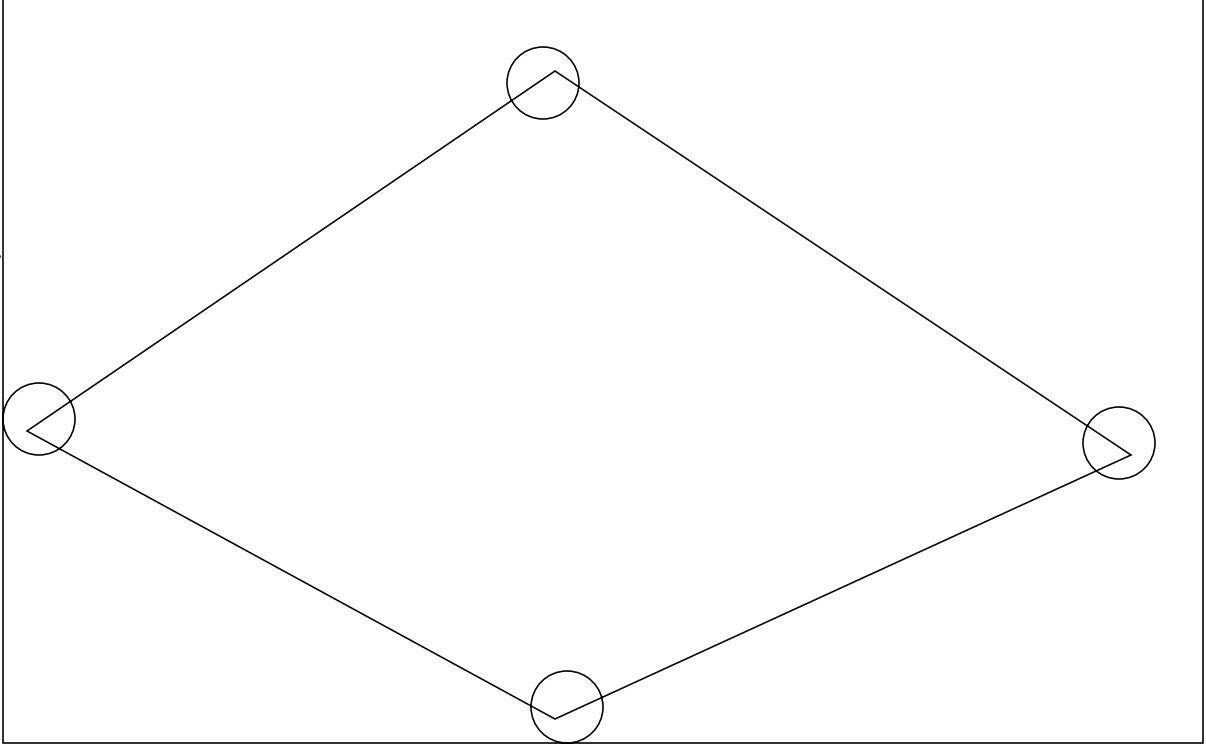


We'll give you this one: `_machine.Draw(POINT1, POINT3);`
`_machine.Draw(POINT3, POINT9);`
`_machine.Draw(POINT9, POINT7);`
`_machine.Draw(POINT7, POINT1);`

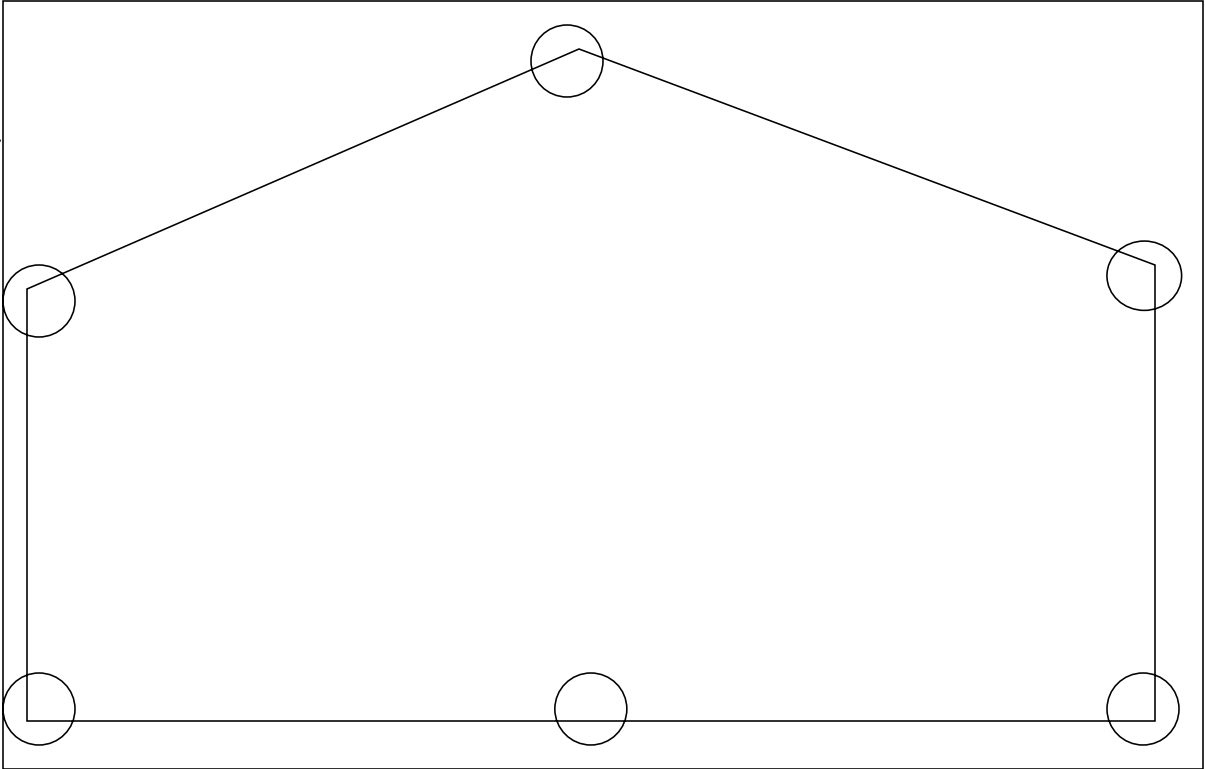




5.



6.



Now try making your own shapes.

More fun with parameters: Now that you understand how to pass points around, lets take a look at some more interesting uses of parameters. When you draw things on the screen, they need to know where they are being drawn. Up until now you've relied on magic to position everything for you. In essence, every time you've wanted to draw something on the screen, we have used our own container (place to draw) to draw it. In general, you will have to supply the place you want to draw things. But what kind of a thing is this? Well, an Applet extends a GP.Containers.Applet, so your Applet is a container! That means that you can draw anything you want it! So how do you pass the Applet? Simple, this always refers to the class that its in, so passing this as a parameter in the Applet class sends the Applet to the drawer. Give it a whirl.....

Making different colors: We have also supplied a set of colors you can use for this assignment to change the color of the applet. Try passing a color in like this:

```
machine = new Drawer(this, RED);
```

The colors we have are: RED, BLUE, GRAY, GREEN, ORANGE, VIOLET, and WHITE.

Even MORE FUN: Try passing "disco" like this `_machine = new Drawer("disco")`. This passes the string of letters disco to the Drawer.