PROCESSING DOCUMENTATION

ASSIGNMENT ONE

Entry One:
Being given our first assignment, processing is easily going to go downhill from here. A static image that comes up different every time it’s run. I don’t even know where to begin. So, I didn’t. Instead, today I stole some code from Rob’s lecture and played around with it. I call it ‘The Matrix’. Because it reminds me of the fighting scene in The Matrix between Neo and all the Agent Smiths. It wasn’t a complete waste of time though. I learnt how to use the for(); thing on processing. I need to learn how to make the randomness happen though.

```cpp
//Size
size(600, 200);

//Background Strokes
background(0);
stroke(255, 60);
for (int i = 0; i < 600; i++) {
  float r = random(10);
  strokeWeight(r);
  float offset = r * 5.0;
  line(i-20, 200, i+offset, 0);
}

//Line for SIN
noStroke();
smooth();
fill(0);
float angle = 0; // Angle to receive sine values from
float yOffset = 100; // Y offset NO
float scaleVal = 35; // Scale value for the wave magnitude
float angleInc = PI/40; // Increment between the next angle
for (int x = 0; x <= width; x += 5) {
  float y = yOffset + (sin(angle) * scaleVal);
  rect(x, y, 2, 4);
  angle += angleInc;
}

//Line for COS
for (int x = 0; x <= width; x += 5) {
  float y = yOffset + (cos(angle) * scaleVal);
  rect(x, y, 2, 4);
  angle += angleInc;
}
```
**Entry Two:**
Stealing code once again proves to be handy! I stole that random line code (that I used for the background of the matrix thing) and successfully changed it, to make random circles appear all over my canvas! I also thought this would be an awesome opportunity for playing around with colour and opacity. I have to say, I love the random function! My circles look so pretty with different colours! I even managed to get the strokes to be different thickness and weights! I call this one ‘Bubbles’

```javascript
//set size of page
size(600,200);
background(random(255));
stroke(255,60);
for (int i = 0; i < 600; i++) {
  float r = random(10);
  strokeWeight(r);
  stroke(random(255), random(255), random(255), random(255));
  fill(random(255), random(255), random(255), random(255));
  float offset = r;
  ellipse(random(600), random(200), width/6, height/2);
}
```
Entry Three:
Okay, I know I should have started my processing assignment by now, but I haven’t. I have no idea what to do and I’m slowly starting to panic. Rob said to do something based on real life, take some thing and code it. Well, that’s easier said then done. I was thinking about my awesome circles and I thought I could perhaps model my thing on cell reproduction. Nerdy, I know. But it would have been fun. The problem is, I can’t show cell division in a static image. Maybe when we move on to moving images. I wanted to start with one cell (just a pretty circle) and have it break apart and copy itself. The resulting two would do the same, and it would just keep going until my page was full. I don’t even know if it’s possible to code that.

Entry Four:
Thinking about the Cell thing I mentioned before, I wanted to make circles that randomly appear on a canvas but do not cover the whole thing, I did this by working from my previous bubbles thing and messing around with the values, after playing around with it all for a while, I found the thing that caused it. Couldn’t settle on a colour so I used random(); which I have really grown to love. I still have no idea what to do for my assignment!

```java
//test sketch for assignment one
size(300,300);
background(random(255), random(255), random(255));

//circle
for(int w=0; w<height; w+=50) {
    float h=random(width);
    stroke(5);
    ellipse(w,h,width/6,height/6);
}
```
Entry Five:
So, I still have no idea what to do for my assignment, but I saw ‘The Eye’ Last night and besides scaring me to death, it made me think about Variation. I decided to code a set of eyes that will change colour randomly every time it is run. I just needed something to play around with because I didn’t entirely understand the conditional statements and it was beginning to freak me out. I am beginning to get them now, and my eyes change colour! And this is the first piece of coding I have done where I didn’t steal any of my code! Woooooo!

```javascript
//set size of canvas
size(600,200);

//Background
background(255,160,122);

//Left-Eye Sclera
strokeWeight(3);
ellipse(150,100,300,200);

//Left-Eye Iris
float a = random(10);
float b = random(10);
if(a > b) {
    fill(0,0,255);
}
if(a < b) {
    fill(0,255,0);
}
if(a == b) {
    fill(255,0,0);
}
fill(0,255,0);

//Left-Eye Spark

//Right-Eye Sclera
fill(255);

//Right-Eye Iris

//Right-Eye Iris

//Right-Eye Pupil
fill(0,0,255);

//Right-Eye Pupil
fill(0,255,0);

//Right-Eye Pupil
fill(255,0,0);

//Right-Eye Pupil
fill(0,0,255);

//Right-Eye Pupil
fill(0,255,0);

//Right-Eye Pupil
fill(255,0,0);

//Right-Eye Pupil
fill(0,0,255);
```

//set size of canvas
ellipse(150,100,50,50);
//Left-Eye Spark

//set size of canvas
ellipse(150,100,300,200);
//Left-Eye Sclera
fill(255);

//set size of canvas
ellipse(150,100,150,150);
//Left-Eye Pupil
fill(0);

//set size of canvas
ellipse(450,100,300,200);
//Right-Eye Sclera
fill(255);

//set size of canvas
ellipse(450,100,150,150);
//Right-Eye Pupil
fill(0);

//set size of canvas
ellipse(450,100,50,50);
So, the eyes change colour between green and blue (sometimes red – in theory). At first I had one random number generate up to 100 and then had if (a<50) then it’s blue, if (a>50) then it’s green, But when I ran it the colours didn’t change as often as I would have liked them to. I would run it like 20 times in a row, and get blue everytime. So, I reduced the random to 10 with (a>5) and (a<5), which was better but still was too consecutive. So I made two random numbers and used (a>b) and (a<b). This made it a lot more random. I also coded it to turn red if a=b, however this has never occurred when I ran it. The only way I know it actually worked was I set both of the values of “a” and “b” to 2 and the eyes turned red.

I was pretty happy with this sketch because I used everything besides the ‘Loop’ that is required of us in our assignment and, as simple as it is, I am glad that I could code it all by myself. Now, I have to start my actual assignment. But I still have no idea what to do.

**Entry Six:**

Last night, my friends and I went to the Campbelltown show, which ended up being a complete waste of money, but not time! I finally have an inspiration for my first assignment. Fireworks!!

I want to code something that will look like fireworks and will look different every time it’s run. So far, the only thing I know that I will be doing is background(0);

Besides that I have no idea what to do. Right now I am thinking of just having random spirals and explosions that appear whenever you hit run. I will go start it now.
Entry Seven:

I successfully made an arc that changes colours. But I want a whole heap of them going around the same point (for the firework) but I can’t seem to figure it out. I know that I shouldn’t have to do each one individually. Rob said that programmers are lazy and never type more than they have to. So, surely they have made a way for me to put that into a loop.

That’s what I thought at least. It doesn’t work! I found the rotate(); function. And I am trying to use it. But my thing isn’t rotating! It is going all over the place! I have tried so many different ways. I might have to resort to individual drawings if this is going to keep being a pain.

```java
for (int x = 0; x < 600; x++) {
    stroke(random(255), random(255), random(255));
    strokeWeight(3);
    translate(width/2, height/2);
    rotate(PI/300);
    noFill();
    arc(300, 100, 50, 100, TWO_PI-PI/2, PI/2);
}
```

No matter what I do to the rotate thing it turns out the exact same! With the exception that sometimes it disappears. Changing the values of “x” and “y” make it even worse. They just spiral off in any one direction. There has got to be something I’m doing wrong. And I have no idea what it is! Now I wish I had invested in that Casey Reas book. There is nothing at all on the internet that helps! I don’t get it, it works in the “reference.”

I think I will definitely need to make an investment in that book.
**Entry Eight:**

So, I found this at http://www.vimeo.com/665084 and I really like it, I have no idea how the guy did it but obviously he used a rotation of some kind. He even says “This is a sketch I did tonight in the fabulous processing environment, exploring the rotate and translate commands.” So there is my proof that that’s how he did it.

He used translate(); as well. I have no idea what that does though. It’s in my code as well, but only because it was in the example. I guess its back to the drawing board.

**Entry Nine:**

I just about hate processing. The very helpful reference just told me that translate(); moves the object to where you tell it to. I see how this works in theory but in practice it doesn’t work at all.

So I am supposed to somehow make all the arcs translate one by one into the correct position. I know there is probably some kind of genius programming thing that does it for you. The problem is, I am not a genius programmer.

I tried putting all sorts of values into my translate(); command but none of them do anything useful. At best, they do nothing at all. I am going to demote my curve to a
line and try to work it out from there. Hopefully, then I can replace the line with an arc again and hopefully have it work.

Examples of translate(); not working at all for me. At all.

**Entry Ten:**
Kaz is my new hero. I was struggling with my line (which didn’t end up being any easier than the arc) when he came online. He did confuse me a little at first. But after repeating himself in about 3 different ways, I finally understood how the rotate and translate functions really work, and how they work together!

Translate is the point that I want them to rotate around! And I can use the radians(x); thing to convert from degrees to radians to make it easier to calculate for me (I forgot about that.) I also had to “untranslate” so I had to mark the same translate points with negative values. Otherwise it will mess up the rest of my code apparently.

This is what I got when I finally got rotate(); and translate(); working.
for (int x=0; x<500; x+=10) {
    translate(300,100);
    rotate(radians(10));
    strokeWeight(random(5));
    stroke(random(0,255), random(0,255), random(0,255), random(100,255));
    translate(-300,-100);
    noFill();
    line(300,100,300,200);
}

Entry Eleven:

float r=random(10);

for (int x=0; x<600; x+=10) {
    translate(500,50);
    rotate(radians(10));
    strokeWeight(random(5));
    stroke(random(255), random(255), random(255), random(255));
    translate(-500,-50);
    noFill();
    arc(500,100,250,100,PI,TWO_PI-PI/2);
}
Okay, so I got the arcs to rotate now using my newly working code. But I can only get them to rotate in one position.

I thought if I changed the translate values to random, then it would rotate around a random point. Right?. Wrong!

```plaintext
float r = random(10);
float p = random(600);
float s = random(200);

for (int x=0; x<600; x+=10) {
    translate(p, s);
    rotate(radians(10));
    strokeWeight(random(5));
    stroke(random(255), random(255), random(255), random(255));
    translate(-p, -s);
    noFill();
    arc(500, 100, 250, 100, PI, TWO_PI-PI/2);
}
```

Some pretty interesting designs came up, but because the arc was starting in a specific place, they didn’t join up at the centre. I thought I would solve that by changing the arcs locations as well. But it didn’t work out too well.

```plaintext
float r = random(10);
float p = random(600);
float s = random(200);

for (int x=0; x<600; x+=10) {
    translate(p, s);
    rotate(radians(10));
    strokeWeight(random(5));
    stroke(random(255), random(255), random(255), random(255));
    translate(-p, -s);
    noFill();
    arc(2*p, 2*s, 250, 100, PI, TWO_PI-PI/2);
}
```
Again, it gave out some interesting designs, but they didn’t exactly look like fireworks. So I decided on an alternative (that would also incorporate my IF statements).

I coded two different arc spiral things and I used IF and random numbers to determine which one will show up. Pretty much the same way that I made the eyes change between blue and green using the if(); statement. It worked pretty well except I hit a small bump when it alternated between my first arc spiral and complete blackness, but it turned out that I simply had my if();’s bracketed inside each other (thanks again, Kaz!)

I will add more to my fireworks but, am not sure of what to add yet. I might sleep on it.
Entry Twelve:

I added random coloured points in the background (So I had to place the code above my spiral ones) so that they could (hopefully) represent the extinguished fireworks in the sky. When fireworks stream out the last bit usually glows for a few seconds before it dies and that’s what I was trying to represent with the coloured points in the background. It was really easy to code, since I was playing around with all the random circles and lines before. But I think it makes a good contribution. I do still plan to add more, maybe random lines or something. But I’d have to make them come out of the same point so it looks like the fireworks are shooting up. I’ll play around with that later.

This is my code for the random points in the background.

```java
float r = random(10); // Give r a random value up to 10

// Spiral Fireworks
for (int x=0; x<600; x+=10) {
    if (r<5) { // If r is less than 5
        translate(500,50); // Rotate around point (500,50)
        rotate(radians(10)); // At 10 degrees
        strokeWeight(random(5)); // Random Stroke Weight up to 5
        stroke(random(0,255), random(0,255), random(0,255), random(100,255)); // Random Str
        translate(-500,-50); // End rotate around point
        noFill(); // Do not fill arcs
        arc(500,100,350,100,PI,TWO_PI-PI/2); // The arc to be rotated
    }
    if (r>5) { // If r is more than 5
        translate(100,100); // Rotate around point (100,100)
        rotate(radians(10)); // At 10 degrees
        strokeWeight(random(5)); // Random Stroke Weight up to 5
        stroke(random(255), random(255), random(255), random(100,255)); // Random Stroke
        translate(-100,-100); // End rotate around point
        noFill(); // Do not fill arcs
        arc(100,150,250,100,PI,TWO_PI-PI/2); // The arc to be rotated
    }
}

// The dots in background
for (int h = 0; h<600; h++) { // For the values of h
    strokeWeight(3); // Make StrokeWeight 3
    float y = random(200); // Assign y a random value up to 200
    stroke(random(255), random(255), random(255)); // Random colours - Full spectrum
    point(h, y); // The point to mark
}
```
And that is what it looks like combined with my spiral code.

**Entry Thirteen:**

I have added the random lines I talked about before. This was pretty simple too compared to the spiral thing that took forever. I just put the line on a loop and made it begin at any random point on the canvas and end at (600,200) this way it looks like they are beaming out of the same way. I set the stroke weight to random up to 5 and set the colour to random shades of blue. “i+=50” so that they do not occur too often. I like the result.

Then I did the same thing at (0,200) and changed it from blue to red to give it more colour and make it look more like a fireworks display (since they generally are not only one colour).
And when put all together it looked like this:

Then, thanks to Julia’s helpful suggestion I added smooth(); to my code and it gave my ‘fireworks’ a smoother, more flowing feel.

It is quite possible that I am finished. I just ran through the checklist and I have everything I am supposed to. (ie – Randomness, Variables, Conditionals, Loops, Drawing commands and colour.)

**Entry Fourteen:**
Now, I am definitely finished. But I have made some changes to my code! I still have the two different spirals, but I made them move around their own little areas by using translate() and pushMatrix() and popMatrix(). (Thanks to Michael who taught me how to use that!) I tried until early morning last night to get that to work! So that now, the images generated are a little more random than before. I had a bit of difficulty making the spirals stay on the canvas and not disappear completely, but after some period of trial and error I got it to show up. I had to do print(); so I could see which one was actually showing up and which one was disappearing, it was hard
to tell since the colours were the same, but I really wanted them to both be random. So I came up with the alternative of making one range between 0 and 150 and the other between 150 and 255, so that one is brighter than the other.