COMPUTER #1 – MATH 1260, FALL 2014

DUE FRIDAY SEPTEMBER 19TH

You should follow the instructions below. At various points I ask questions about what you are entering. You should type your answers in the mathematica worksheet via the plus command, "plain text". While this is due on Friday, you merely need to print out your work in class today. You do not have to complete the worksheet.

We start with the reparametrization from a couple weeks ago (that we failed to do in class). Consider the curve

$$\vec{v}(t) = \langle \cos(t), \sin(t), (2/3)t^{3/2} \rangle$$

1. First plot it using the function

```
ParametricPlot3D[{Cos[t], Sin[t], (2/3)*t^(3/2)}, {t, 0, 3*Pi}]
```

Be very careful to use the same sorts of parentheses as I do. Try changing the `3*Pi` to another value. What are you changing?

2. Let's see if we can use the computer to help us compute the arc length. First lets differentiate our parametrization.

```
tangent = D[{Cos[t], Sin[t], (2/3)*t^(3/2)}, t]
```

Next let's compute the length of the tanget vector as a function of time.

```
tLength = (tangent[[1]]^2 + tangent[[2]]^2 + tangent[[3]]^2)^(1/2)
```

What do you think the `[[3]]` represents? Regardless, let's integrate this from $t = 0$ to a.

```
aLength = Integrate[tLength,{t,0,a}]
```

The conditional expression is because Mathematica is concerned the integral may not converge (Mathematica doesn't know that a is a positive real number). Try it again with

```
aLength = Integrate[tLength, {t, 0, a}, Assumptions -> a > 0]
```

Finally, let's try one more time but now we use the built in `ArcLength` function.

```
ArcLength[{Cos[t], Sin[t], (2/3)*t^(3/2)}, {t, 0, a}]
```

Did you get the same answer?

3. But we wanted to parameterize with respect to arc length. Finally let's find our parametrization.

```
myReparam = Solve[aLength == b, a]
```

This will solve for `a` in terms of `b` (the length). You'll need to probably modify the above with `[[#]]` in order to just get the real solution (instead of the complex solutions too).

Now that we have the reparametrization, let's check that it really works. Set

```
t1 = a/.myReparam[[1]][[1]]
```

Finally think about what value

```
ArcLength[{Cos[t1],Sin[t1],(2/3)*t1^(3/2)}, {b,5,7}]
```

should produce (what do the 5 and 7 represent), does it?

Besides merely being better able to do computations without mistakes, it can also graph a number of functions.

4. Given an implicit equation, we can plot it. Look at the following equations, can you guess what they look like before you plot them? (the last one is not one we've considered before)

```
ContourPlot3D[{x^2 + y^2 == z^2}, {x, -2, 2}, {y, -2, 2}, {z, -2, 2}]
ContourPlot3D[{x^2 + y^2 + z^2 == 1}, {x, -2, 2}, {y, -2, 2}, {z, -2, 2}]
ContourPlot3D[{x^2 - y^2 == z}, {x, -2, 2}, {y, -2, 2}, {z, -2, 2}]
ContourPlot3D[{x*y/(x^2+y2) == z}, {x, -2, 2}, {y, -2, 2}, {z, -2, 2}]
ContourPlot3D[{x^2*y - z^2 == 0}, {x, -2, 2}, {y, -2, 2}, {z, -2, 2}]
```

Now, use Mathematica to draw a ellipsoid with semi-principal axes (radii) of lengths 2, 5 and 7. Make sure the entire ellipse is visible.

5. You can plot more than one 3D shape simultaneously in Mathematica. For instance consider the plane and the sphere.

```
ContourPlot3D[{{x^2 + y^2 + z^2 == 1, x + y + z == 0}}, {x, -2, 2}, {y, -2, 2}, {z, -2, 2}]
```

6. Sometimes it is more convenient to give a parametric presentation. In the below, the first list has the x-coordinate as a function of x and y, the y-coordinate as a function of x and y and finally the z-coordinate as a function of x and y. Here is the function from yesterday.

```
ParametricPlot3D[{x, y, x*y/(x^2 + y^2)}, {x, -2, 2}, {y, -2, 2}]
```

Plot the upper half of the unit sphere using a parametric plot.

Finally, consider the following parametric plot.

```
ParametricPlot3D[{a,b^2,a*b},{a,-2,2},{b,-2,2}]
```

This is a parametric plot of one of the contour plots above. Figure out which one. Can you give an algebraic justification (by hand) of why these two commands plot the same object?

7. Finally, let us verify Clairaut's Theorem in a special case. Try the following, and try to understand it.

```
f = Exp[x*y]*Cos[x^2+2*y]
fx = D[f,x]
fy = D[f,y]
fxx = D[fx,x]
fxy = D[fx,y]
fyx = D[fy,x]
fyy = D[fy,y]
fxy == fyx
```

The use of the double equals sign `==` at the end returns `true` if the two sides are equal. This is to be compared with the single equals `=` or `:=`.

8. Try drawing some other fun equations in Mathematica like $y^2 * z - x * (x - z) * (x + z)$ and see what they look like.