Maple for Today's Lab

- Note 1: Try to not use the percentage sign in Maple (although we used it quite a bit in our examples). Try to always assign your statements to a variable name, then use that in later statements.
- Note 2: When plotting several expressions on one graph, use the square brackets instead of the curly braces (as we did last time). There's a reason for this (explained below).

Equations versus Assignment

We will recall that Maple uses `:=` and `=` differently- The first is an assignment (Assign whatever is on the right side to the variable name on the left), while the latter is a statement of equality. For example,

```
A:=3*x+5;
B:=A=2;
solve(B,x);
```

In the first statement, the expression $3x+5$ is assigned to the variable A. Now we can simply use A whereever we needed $3x+5$, as in the second line.

Expressions versus Functions

Last time, we saw the difference in Maple between an *expression* and a *function* (and an *equation*). For example (line numbers for discussion only):

```
1 f:=x^2+5;
2 F:=x->x^2+3;
3 F(4);
4 subs(x=4,f);
5 plot([f,F(x)],x=-1..2);
```

Note the difference in Line 5 for plotting.

New Maple Commands and Ideas

In Maple, a **Set** is a sequence enclosed by curly braces, a **List** is a sequence enclosed by square braces. In a **set**, every element occurs only once- In a list, the order of the elements matters, and elements may be repeated (so typically we will use a **list**).

Here are some examples. First, we look at a sequence (with no delimiters), then construct a list of data to plot:

```
1 seq(n^2/(3-n^2),n=0..10);
2 limit(n^2/(3-n^2),n=infinity);
3 data:=seq([n,n^2/(3-n^2)],n=0..10);
4 plot([data],style=point,symbol=circle);
5 Seq1:=[seq(n^2/(3-n^2),n=0..5)];
6 Seq1[3];
```

Line 6 shows you how to access list elements. We will probably always work with lists (square brackets), and very seldom with sets (curly braces).

Maple can also work with series- Even the alternating harmonic series:

```
7 sum(1/n^2,n=1..infinity);
8 sum((-1)^n/n,n=1..infinity);
```

Power Series

Of course, no section on series would be complete without power series. A couple of definitions to start:

Definition: A power series based at $x = a$ is a polynomial (of possibly infinite degree) of the form:

$$c_0 + c_1(x-a) + c_2(x-a)^2 + c_3(x-a)^3 + \ldots = \sum_{k=0}^{\infty} c_k (x-a)^k$$

Definition: The Taylor series based at $x = a$ for $f(x)$ is a special power series. It is defined as:

$$f(a) + f'(a)(x-a) + \frac{f''(a)}{2!}(x-a)^2 + \frac{f'''(a)}{3!}(x-a)^3 + \ldots = \sum_{k=0}^{\infty} \frac{f^{(k)}(a)}{k!}(x-a)^k$$

Definition: The Maclaurin series is a special type of Taylor series where $a = 0$:

$$f(0) + f'(0)x + \frac{f''(0)}{2!}x^2 + \frac{f'''(0)}{3!}x^3 + \ldots = \sum_{k=0}^{\infty} \frac{f^{(k)}(a)}{k!}x^k$$

Example 1: Find the Taylor series to $f(x) = \sin(x)$ at $x = 0$ up to 7th order (the order here will be the highest power of the polynomial approximation). Plot it (with f), then plot the error between it and the original function:

NOTE: Put the whole plot command below on a single line, don't break it as shown- that was done for the margins.

```
f:=x->sin(x);
g:=taylor(f(x),x=0,8);
g:=convert(g,polynom);
plot([g,f(x)],x=-2*Pi..2*Pi,y=-2..2, linestyle=[dot,solid],
     color=black);

plot(abs(g-f(x)),x=-2*Pi..2*Pi);
```

Example 2: Find the Taylor series to $f(x) = \mathrm{e}^{-(x-1)^2}$ at $x = 1$ up to 15th order (the order here will be the highest power of the polynomial approximation). Plot it (with f), then plot the error between it and the original function- You might use a smaller interval- say $-1 \le x \le 3$

```
f:=x->exp(-x^2);
g:=taylor(f(x),x=1,16);
g:=convert(g,polynom);
plot([g,f(x)],x=-1..3, color=[red,green],title="Fun\nGraph",
     legend=["Taylor","Original"]);

plot(abs(g-f(x)),x=-1..3);
```

Doing Something Over and Over and Over

If you need to perform some computation over and over, consider using a "do loop". Here is a sample to try:

```
x:=sqrt(2.0);
for i from 1 to 10 do x:=sqrt(x); od;
```

What happened? *Result:* You should see that you first compute $\sqrt{2}$, then assign that result to x. Then compute $\sqrt{\sqrt{2}}$, then assign that result to x. Do that 9 more times.

In general, a loop to do something 10 times looks like this:

```
for i from 1 to 10 do xxx; od;
```

where xxx is done over and over again.

Here is another example, where we build the sequence a_k, where $a_1 = 3$, $a_2 = 1$, and:

$$a_{k+1} = 3a_{k-1} - a_k$$

```
a[1]:=3;  a[2]:=1;
for i from 2 to 8 do a[i+1]:=3*a[i-1]-a[i]; od;
```

And here is another example, where we print something to the screen **Typing Tip:** To get a new line (without Maple interpreting your commands), use `Shift + Enter` after each line, then just `Enter` when you're finished.

```
N:=9;  S:=14;
for k from 1 to N do
   S:=S + 1.0/i^2;
   print(i,S);
od;
```