[6.49] Taylor() polynomial function finds tangent line

The line tangent to a function y = f(x) at a point x = a is defined by the criteria that the line passes through point (a,f(a)) and the slope of the line is the derivative of f(x) at x = a. If the tangent line is

$$y = m \cdot x + n \qquad [1]$$

then $$m = \frac{d}{dx} f(x) \mid x = a \qquad [2]$$

and if f(a) = b, we find n with $$n = b - m \cdot a \qquad [3]$$

The tangent line is the first-order Taylor series polynomial at x = a, so we can use the built-in *taylor()* function to find the line equation. The syntax for *taylor()* is

```
taylor(expr, var, order [,point])
```

where *expr* is the expression, *var* is the indpendent variable, *order* is the polynomial order, and the optional *point* argument is the point at which to find the series expansion. To find the tangent line, we set *order* = 1 and *point* = a. For example, if

$$y = 2 \cdot x^2 - 7 \cdot x + 1$$

and we want to find the tangent line at x = 3, then use

```
taylor(2*x^2-7*x+1,x,1,3)
```

which returns 5(x - 3) - 2. Use *expand()* on this result to put it in the more common form of 5x - 17.

We can also use this method to find symbolic results. For example, suppose

$$y = \ln\left(x^2\right)$$

and we want to find the tangent line at x = a, then use

```
expand(taylor(ln(x^2),x,1,a))
```

which returns $$\frac{2 \cdot x}{a} + \ln\left(a^2\right) - 2$$

This method fails if the CAS cannot find the Taylor polynomial for the expression. This is the case for complicated user-defined functions. In this case, you can find the derivative numerically (see, for example, tip 6.26), then solve for the tangent line constant with equation [3] above.