Automatic Differentiation

Compiling & Running - `make`

I used automake as recommended, so to compile and run it should hopefully be as simple as

$ ./configure

$ make

$ `./autodiff -t 10000000 < ./test/test1.in`

All source can be found in ./src and some test input files can be found in ./test. The following section describes the interface

Interface

The interface goes as follow, when you run autodiff without a file you will be prompted with the following screen

```
To start our function is f_0(x) = f_1(x)
Pick f_1(x)

The options are:
0 - As is          1 - polynomial     2 - trigonometric
3 - hyperbolic     4 - exponential    5 - logarithmic
The composers are:
6 - addition       7 - product
8 - quotient       9 - power
```

To construct any function we work from the outside in, so as to construct test function 3, $h(x)$, we would take the following steps

- $h(x) = h_1(x)/h_2(x)$ ... quotient
- $h_1(x) = exp(h_3(x))$ ... exponential
- $h_3(x) = x^{2.5}$ ... power ... As is
- $h_2(x) = h_4(x) + h_5(x)$ ... addition
- $h_4(x) = log(x)$ ... logarithmic ... As is
- $h_5(x) = \coth(x)$ ... hyperbolic ... coth ... As is

The commands can be found in ./test/test3.in. The following are the test functions

Test 1

$$f(x) = \sin(x) \tag{1}$$

Test 2

$$g(x) = \tan(x^2 + 2x + 1) \tag{2}$$

Test 3

$$h(x) = \frac{\exp(x^{2.5})}{\log(x) + \coth(x)} \tag{3}$$

Test 4

$$k(x) = \frac{5\tan(x^{2.5})}{x^3 + 25x + 9} \tag{4}$$

Speed Tests

When the test functions were implemented at run time vs directly implemented and run 10^7 times, the following times were recorded. As you can see the direct implementations were always faster than the run time implementations.

	t_{virtual}	t_{direct}	$t_{\text{direct}}/t_{\text{virtual}}$
$f(x)$	0.659322s	0.592783s	1.112248
$g(x)$	1.27494s	0.622899s	2.046784
$h(x)$	9.66068s	7.41847s	1.302247
$k(x)$	7.23954s	4.96945s	1.456809