Eine C++-Klasse zum automatischen Differenzieren

```
class Deriv
{
private:
  double value,deriv;
public:
  Deriv(double v=0.0,double d=0.0) { value=v; deriv=d; }
    .
    .
  friend Deriv operator*(Deriv x,Deriv y)
    { return Deriv(x.value*y.value,x.deriv*y.value+x.value*y.deriv); }
  friend Deriv operator/(Deriv x,Deriv y)
    { return Deriv(x.value/y.value,(x.deriv-x.value*y.deriv/y.value)/y.value); }

  friend Deriv sqrt(Deriv x)
    { return Deriv(sqrt(x.value),0.5*x.deriv/sqrt(x.value)); }
  friend Deriv exp(Deriv x)
    { return Deriv(exp(x.value),x.deriv*exp(x.value)); }
  friend Deriv log(Deriv x)
    { return Deriv(log(x.value),x.deriv/x.value); }
  friend Deriv sin(Deriv x)
    { return Deriv(sin(x.value),x.deriv*cos(x.value)); }
  friend Deriv cos(Deriv x)
    { return Deriv(cos(x.value),-x.deriv*sin(x.value)); }
  friend Deriv atan(Deriv x)
    { return Deriv(atan(x.value),x.deriv/(1.0+x.value*x.value)); }
    .
    .
};
```