

```

function Test_AdaptiveSteppers(t0, tf)
    function func(t, y)
        [exp(t)]
    end

    y0, exact = [exp(t0)], exp(tf)
    methods = DopriStep, CarpStep, RKFStep

    for a=methods
        ts, ys = AdaptiveStepper(func, a, t0, copy(y0), tf)
        println("Actual error of ", a, ": ", abs(exact - ys[end][1]))
    end
end

#Test_AdaptiveSteppers(20, 21)

#### Systems ####

function SimpleSystem(t, y)
    [cos(t)]
end

function HarmonicOscillator(t, y)
    y1, ydot = y
    [ydot, 100*cos(20*t) - ydot/4.0 - y1]
end

function FourthOrderSystem(t, y)
    y1, ydot, yddot, ydddot = y
    [ydot, yddot, ydddot, exp(t)*cos(100*t)-7*ydddot-17*yddot-17*ydot-6*y1]
end

#t0, tf = 0, 15
#y0 = [0, 0, 0, 0]

#ts, ys = Stepper(FourthOrderSystem, BackwardEulerStep, t0, y0, tf, 0.01)
#ts, ys = MultiStepper(FourthOrderSystem, AdamsBash4, t0, y0, tf, 0.001)
#plot(ts, [a[1] for a=ys], "o-")

```