
```

% n order taylor method
function output = Taylors(f, a, b, N, init, n, realF)
    syms T(t, y)%T(t, y) is entire taylor series, t and y are
    independent and dependent variables respectively
    %f is f(t, y) = y'(t)
    %a and b are range of this IVP
    %N is number of steps
    %init is given initial value
    %n is order of taylor method
    h = (b-a)/N; %step size
    w = init; %initial value
    t_ = a;% our t variable to input to functions
    toRet = zeros(3, N);

    %constructing symbolic function T(t, y)
    syms func(t, y) g(t, y) t y%func(t, y) to do everything
    symbolically, g(t, y) to differentiate on
    func(t,y) = f;%for g(t,y), to maintain symbolic y'(t)
    T(t, y) = h*func(t,y);%entire function
    g(t, y) = f;%will use for derivative ==> f^(n)(t, y)
    for i = 1: n-1
        g(t, y) = diff(g(t,y), t) + diff(g(t,y), y)*func(t,y); %this
        is the nth derivative of f(t, y) with respect to y
        T(t, y) = T(t, y) + (h^(i+1))/(factorial(i
+1))) * g(t,y); %combining all of them
    end

    %fprintf("VALUE%.5f\n", subs(T, 0, 1));
    %doing approximation here
    Taylor = matlabFunction(T(t, y));%symbolic functions are
    incredibly slow when substituting values, so I converted it to a
    matlabfunction
    for i = 1: N
        w = w + Taylor(t_,w);%main approximation line
        t_ = a + i*h;%updating t
        toRet(1, i) = t_;
        toRet(2, i) = w;
        toRet(3, i) = realF(t_);
    end
    output = toRet;
    return;
end

```

Not enough input arguments.

Error in Taylors (line 10)
h = (b-a)/N; %step size

Published with MATLAB® R2018a