How To Use SolvIVDifD.m

1. Verify that SolvIVDifD.m is the appropriate template file to use
   a. Equations of the following form are to be solved for the value of t where one of the z_i reaches a known value, z_{if}, and for the corresponding values of the other z_j, $j \neq i$

$$\frac{dz_1}{dt} = f_1(t, z_1, z_2, \cdots, z_n); \quad z_1(t_0) = z_1^0$$
$$\frac{dz_2}{dt} = f_2(t, z_1, z_2, \cdots, z_n); \quad z_2(t_0) = z_2^0$$
$$\vdots$$
$$\frac{dz_n}{dt} = f_n(t, z_1, z_2, \cdots, z_n); \quad z_n(t_0) = z_n^0$$

2. Save a copy of SolvIVDifD.m as *newname*.m in the current MATLAB working directory or in a directory that is in the MATLAB search path ("*newname*" should be some meaningful file name)
3. Change the function declaration statement to match the filename from step 2
   a. from: `function [t_f,z] = SolvIVDifD`
   b. to: `function [t_f,z] = newname`
4. Find the comment indicating the location of the first required file modification
   a. Replace

```
% EDIT HERE (Required modification 1 of 6):
        % define universal and experimental constants here
```

   b. With statements defining variables and assigning their values for each constant that appears in the problem being solved
      i. The values should be entered or converted to consistent units
5. Find the comment indicating the location of the second required file modification and change the lines that follows the comment
   a. from:

```
dzdt = [
    % Evaluate dz1/dt = f1(t, z1, z2, z3, ..., zn) here
    % Evaluate dz2/dt = f2(t, z1, z2, z3, ..., zn) here
    % and so on through fn, one per line
];
```

   b. so that the first line within the square brackets evaluates the function f_1 in step 1.a, the second line evaluates the function f_2 in step 1.a, and so on.
6. Find the comment indicating the location of the third required file modification and change the line that follows the comment
   a. from:

```
t0 = ; % insert the independent variable initial value here
z0 = [
    % insert the initial values of dependent variables z1 here
    % insert initial values for z2, z3, ..., one per line
];
```

b. so that the initial value of the independent variable, t_0, is assigned to `t0` and

c. so that the first line within the square brackets equals $z_1(t_0)$ from the equations in step 1.a, the second line equals $z_2(t_0)$, and so on.

7. Find the comment indicating the location of the fourth required modification and change the line that follows

   a. from `tf = ; % insert a large number, as just instructed`

   b. so that a very large number is assigned to `tf`

8. Find the comment indicating the location of the fifth required modification and change the line that follows

   a. from `stop_when = ; % replace this comment with the stopping criterion`

   b. so that the variable named `stop_when` will equal zero when the dependent variable, z_i, reaches its known final value, z_{if}

9. Find the comment indicating the location of the sixth and final required file modification and change the lines that follows the comment

   a. from:

```
% Enter code to calculate any other desired quantities using the
% results contained in z. (Alternatively, z will be returned when this
% template file terminates; other quantities can then be calculated
% at the MATLAB command prompt using the returned values
```

   b. so that any additionally needed quantities that depend upon the unknowns are calculated

      i. Do not use semicolons at the ends of these statements; if you do, they will not appear in the output

      ii. If you want to be able to use these quantities after this function has completed its calculations, they must be added to the list of returned variables and change the function call below, accordingly

10. If the ODEs in step 1.a are stiff, change the solver from ode45 to ode15s
11. Save the modified version of newname.m (where newname is the filename chosen in step 2)
12. Execute the file by typing the following at the MATLAB command prompt (again using “newname” to represent the filename chosen in step 2): `[t_f,z] = newname`
13. The following quantities will be listed in the MATLAB command window

    a. results of the code entered in step 9.b

    b. the final value of the independent variable, t_f (t_f)

    c. the values of the dependent variables, z, evaluated at t_f ($\underline{z}(t_f)$)

14. If t_f is equal to the value assigned to `tf` in step 7.a

   a. return to step 7.a and assign a larger value to `tf`
   b. jump to step 11 and proceed from that point

15. The following quantities will be returned and available within the MATLAB workspace
   a. t_f (scalar) from step 13.b
   b. z (column vector) from step 13.c