TI-84 Program Euler

Re: text p.123 for TI-85 implementation, 4th ed

Re: text p.114 for TI-84 plus implementation, 5th ed

\[ Y' = F(x, y) \]

Comment

PROGRAM:EULERSTR
Program title

:ClrHome Clear the home screen

:Input “Y’=” Str1 Input \( Y' \) i.e. \( \frac{dy}{dx} \), press ALPHA first, store \( Y' \) to the string variable Str1

:String->Equ(Str1, \( Y \) ) Converts string into an equation and store the equation to \( Y \)

:Input “\( X' = \)” \( X \) Initial x value, i.e. \( x \) note: subscript for \( x \) near end of catalog

:Input “\( Y' = \)” \( Y \) Initial y value, i.e. \( y(X) \)

:Input “\( X \) MAX=” \( M \) Final x value i.e. \( x_0 \leq x \leq M \)

:Input “NO. OF STEPS=” \( N \) Number of steps

:(\( M-X \))/\( N \)->\( H \) Step size

:0->\( L \) Initialize the counter \( L \) to zero

:ClrAllLists Sets to 0 the dimension of all lists in memory (use catalog)

:For(I,1,\( N \)) Begin the loop

:ClrHome Clear the home screen to start the output

:L+1->\( L \) Update the counter

:\( X \)->\( L_1(L) \) Store \( X \) in list 1

:\( Y \)->\( L_2(L) \) Store \( Y \) in list 2

:\( Y' \)->\( F \) Assign the function value of \( Y' \) to \( F \)

:(\( L_1(L) \))+\( H \)->\( X \) Updated x value
Euler iteration, updated approximate Y value, i.e. add H*F

Display x and approximate y values

Requires user intervention to press the enter key

End of the loop

Update L counter

Add last X value to \( L_1 \) list

Add last Y value to \( L_2 \) list

Delete from memory the contents of \( Y_i \)

Turn on all stat plots

Define Plot1 as a scatter plot with lists \( L_1 \) and \( L_2 \)

Redefine the viewing window to display all points

Display the graph

Requires user intervention to press the enter key

In the graph, X values are in \( L_1 \) and Y values are in \( L_2 \)

Ends program execution and returns to home screen when “DONE” is printed

Notes:

1. Confirmation of the results when executing the above code can be seen on p.118 (4th ed) and p. 110 (5th ed) in the text in figure 2.4.8. In this example, prompts for \( Y' \) requires the input \( X+Y \) (i.e. ALPHA X+ALPHA Y). The prompt for \( X \) requires the input 0, the prompt for \( Y \) requires the input 1, the X MAX prompt requires input of 1, and the NO. OF STEPS prompt requires input of 10. Whenever a pause occurs on the TI screen, press enter. At the end, a scatter plot of the (x,y) values will appear, press enter and DONE will appear on the home screen indicating that program execution has ended.
2. Another example to try is: Consider $0 \leq x \leq 0.5$ and we want only 2 steps with $y'=y$ and $y(0)=1$. For this example the actual solution is $y = Ce^x = e^x$. At the prompt for $Y'$ enter $Y$, at the prompt for $X_0$ enter 0, at the prompt for $Y_0$ enter 1, the X MAX prompt requires input of 0.5, and the NO. OF STEPS prompt requires input of 2. A summary of generated values for this Euler problem follows:

<table>
<thead>
<tr>
<th>X</th>
<th>0</th>
<th>0.25</th>
<th>0.5</th>
</tr>
</thead>
<tbody>
<tr>
<td>Y actual</td>
<td>1</td>
<td>1.28403</td>
<td>1.64872</td>
</tr>
<tr>
<td>Y approximate</td>
<td>1</td>
<td>1.25</td>
<td>1.5625</td>
</tr>
</tbody>
</table>

Note that 5 place accuracy was used to show the comparison with results using the RKSTRING code for RUNGE-KUTTA.

3. When creating the TI-84 program, use the PGRM key to access commands.

4. When done creating the program, 2nd Quit.

5. One has to press the enter key whenever the Pause instruction is encountered in the program. This is done purposely to allow the user to read all lines of output.

6. -> is created with STO key.

7. To add lines to the code:

   a) I first added as many lines as needed with the entry INPUT, i.e. 2nd insert->PGRM->select INPUT under I/O->enter, then do this repeatedly for as many lines as needed

   b) Then rewrite over the newly created INPUT lines

   c) Note that the PlotsOn command is from 2nd catalog