Application of Numerical Problem Solving in Chemical Engineering Coursework

Presenters: Robert P. Hesketh, Rowan University; Michael B. Cutlip, University of Connecticut

Offered: Wednesday 9:30 am-noon, Thursday 9:30 am-noon

This workshop will provide hands-on experience in the use of interactive problem solving software to participants. Emphasis will be placed on the application of PolyMath 6 for PCs and a new version of PolyMathLite 1.1 for Android Smartphones and Tablets. The workshop presenters will give multiple examples of how numerical problem solving can be integrated into common chemical engineering courses. Participants will be encouraged to integrate numerical methods into their courses so that their students will understand and appreciate the types of problems and efficiencies that solutions using numerical methods can bring to problem solving and modeling of chemical systems. The PolyMath 6 and revised PolyMathLite 1.1 software will be provided to all the participants for this workshop and future use for a full calendar year at no cost. This software is provided by the CACHE Corporation and PolyMath Software. Participants will be required to bring a laptop with the ability to run Windows software.

ASEE Chemical Engineering Division Summer School Workshop - 2017

Application of Numerical Problem Solving in CHEG Coursework

Robert P. Hesketh, Dept. of Chemical Engineering, Rowan University, Glassboro, NJ 08028
Michael B. Cutlip, Dept. of Chemical and Biomolecular Engineering, Univ. of Connecticut, Storrs, CT 06269-3222

Goals: This workshop will provide hands-on experience in the use of interactive problem solving software to participants. Emphasis will be placed on the application of PolyMath 6 for PCs and a new version of PolyMathLite 1.1 for Android Smartphones and Tablets. Participants will be encouraged to integrate numerical methods into their courses so that their students will understand and appreciate the types of problems and efficiencies that solutions using numerical methods can bring to problem solving and modeling of chemical systems.

Scope and Content: The workshop presenters will give multiple examples of how numerical problem solving can be integrated into common chemical engineering courses. The PolyMath 6 and revised PolyMathLite 1.1 software will be provided to all the participants for this workshop and future use for a full calendar year at no cost. This software is provided by the CACHE Corporation and PolyMath Software. Participants will be required to bring a laptop with the ability to run Windows software. The software will be provided as individual copies for student use on their own computers or through local networks that can be accessed by students. Note that many CHEG departments have existing site licenses for PolyMath for Windows OS and for PolyMathLite for Android OS. As for future access, inexpensive educational pricing of the network versions or the individual educational copies will provide the Windows and Android software to the students.

Method of Delivery: Overviews and demonstrations of the PolyMath 6 for PCs and the new PolyMathLite 1.1 for Android Smartphones and Tablets will be presented as the participants will tackle problems with guidance from the instructors. This hands-on problem solving workshop will encourage participants to solve some typical problems on their own using their laptop PCs or Android Phones/Tablets. Problems will require the solution of simultaneous linear equations, simultaneous nonlinear equations, simultaneous differential equations as well as data analysis and regression. The instructors and other knowledgeable volunteers will circulate through the workshop participants to assist with any difficulties and answer individual questions.

Example problems requiring numerical solutions will be provided from main subject areas of CHEG. These will include basic calculations, thermodynamics, fluid mechanics, heat transfer, mass transfer, chemical reaction engineering, phase equilibria, distillation, process dynamics and control, and biochemical engineering.

Throughout the problem solving exercises, the workshop participants will experience the ease of solving these complex problems using PolyMath software; either using PolyMath 6 and PolyMathLite 1.1. The ease in solving complex problems is based on the following attributes:

1) The intuitive interface is probably the easiest-to-use general problem solving software currently available
2) There is complete interchangeability of the problem code between PolyMath 7 and PolyMathLite 1.1 in both directions. The code developed in one package will execute on the other package without changes.
3) No command language or other details to remember - exceptional HELP always available
4) Problem entry mimics mathematical equations so entry into software is very easy
5) New full-screen color-coded editor speeds problem entry or modification
6) Variable names are selected by the user with upper and lower cases distinguished
7) Easy and intuitive equation entry (differential equations, nonlinear equations, etc.)
8) Software identifies undefined variables as a MAJOR AID to problem entry
9) Errors in problem entry are identified and syntax errors must be correct to enable solution
10) Problem code is automatically ordered on execution and problem equations may be entered in any order
11) A MATLAB m-file is automatically generated in the PolyMath report output. This m-file can be directly imported and used in MATLAB.
12) Graphical output of the solution can be automatically generated and then easily exported to other documents or saved as files
13) The PolyMath Report contains the complete solution for the problem and aids in the student dissemination and documentation for a homework problem.

14) Related technical papers and problem libraries available from the PolyMath web site
15) Friendly and efficient support is provided that often comes directly from the authors

Take Home Materials: In addition to a copy of the POLYMATH software, a flash drive will be provided to each participant that will contain the workshop materials, handouts, references and computer files. A special website will be created for use during the workshop and this will be made available to participants and to all summer school attendees that can be accessed when they return home.

Presenter(s):

Robert P. Hesketh is a Professor of Chemical Engineering at Rowan University. He received his B.S. in 1982 from the University of Illinois and his Ph.D. from the University of Delaware in 1987. Robert's research is in reaction engineering, novel separations including supercritical fluids, crystallization and ultrafiltration, green engineering, and the chemistry of gaseous pollutant formation and destruction related to combustion processes. Robert has received over 4.4 million in external funding for educational and technical research projects. Robert has presented his educational innovations in international and national meetings and workshops including the 2002 and 1997 ASEE ChE Summer Schools. Robert's dedication to teaching has been rewarded by receiving several educational awards including the 2006 Chester F. Carlson, 2002 Robert G. Quinn Award, 1999 Ray W. Fahien Award.

Michael B. Cutlip is an Emeritus Professor within the Chemical and Biomolecular Engineering Department at the University of Connecticut and has served as department head and director of the university's Honors Program. He has B. Ch. E. and M. S. degrees from Ohio State and a Ph. D. from the University of Colorado. He has been the Chair and National Program Chair for the ASEE Chemical Engineering Division plus he co-chaired the ASEE Summer School for Chemical Engineering faculty in 2002. His current interests include the development of general software for numerical problem solving and application to chemical and biochemical engineering. Dr. Cutlip is also managing director of Polymath Software that develops and provides problem solving software to higher educational institutions and to individual professional and academics users.

Timing: The workshop requires a time slot that consists of two hours, but it would be preferable to have a single time slot of 2.5 hours. The time slot should not be broken up as the presenters and attendees need to set up their computers and attach any needed power supplies.

Specific Logistical Needs: The workshop will be held in a classroom with computer-friendly tiered seating. Electrical power will be readily available for keeping computers charged. Projection of the presenter's computer screen and other workshop materials will clearly indicate the materials under discussion. There should be WI-FI available in the classroom so that internet sites can be easily accessed by the presenter and the participants.

Additional Review Materials: A demonstration of the type of website that will be created for the new PolyMath 6 and the PolyMathLite 1.1 is available from the link:

polymath-software.com/problemsolvingworkshop.

This website was created for the African Engineering Educational Association Workshop and a similar site will be created and updated for the Summer School. It will feature the new PolyMath 7 for PCs and the PolyMathLite 1.1 for Android Smartphones and Tablets (available from Google Play Store). In preparation for reviewing this Workshop site, please make sure that the latest free Adobe Acrobat Reader DC is installed as the default Acrobat Reader software in your PC.

The following links are to a related book and recent paper:

Problem Solving in Chemical and Biochemical Engineering with POLYMATH, Excel, and MATLAB, 2nd Edition

Enabling Extensive Numerical Problem Solving on Smartphones and Tablets

Schedule of Topics:

1. Overview of Workshop
2. Load POLYMATH Program or go to Hands on Topic 1
3. Hands on Topic 1: Introduction to Polymath and its Error Codes.docx
4. Integration of numerical methods in ChE Courses
5. Capabilities of POLYMATH (PC and Android devices)
6. Hands on Topic 2: Examples of Problems from ChE courses

ChE Course	Problem Name	Numerical Method Illustrated	Equations
Fluids	Unsteady-state tank drainage using a siphon tube (similar to POLYMATH text 8.14) *C&S8-14soln.pdf*	Solution of an first order ordinary differential equation (DEQ)	$\frac{dh_T}{dt} = v_{out}\frac{A_{out}}{A_{tank}}$ $v_{out} = f(h_T)$
	Calculations involving Friction Factors for Flow in Pipes (POLYMATH Text 8.7) and *pipeflow homework* *frictionfactorcalcsoln.pdf* *Excel Tutorial Solver Add-Ins rev4.pdf* Siphon Experiment *Siphon Calcs & graphs.xlsx* *siphon.pol*	Solution of a system of simultaneous nonlinear algebraic equations (NLE)	$\frac{\Delta P}{\Delta L} = 2f_F\frac{\rho v^2}{D}$ $f_F = f(\varepsilon/D\,,Re)$ $Re = \rho v D/\mu$
Advanced Fluids	NonNewtonian fluid flow through a pipe (POLYMATH Text 8.2c) *NonNewtonian C&S 8.2 solutions & comsol.pdf* NonNewtonian fluid flow through an annulus (POLYMATH Text 8.4) *NonNewtonian C&S8.4 polymath&comsol & 3.8-8 solutions 2017.pdf*	Solution of 2 simultaneous first order ordinary differential equations with split boundary value conditions and comparison with solution using COMSOL which is an advanced finite element program	$\frac{d(r\tau_{rx})}{dr} = -\frac{dP}{dx}r$ $\tau_{rx} = -K\left(\frac{dv_x}{dr}\right)\left(\left\lvert\frac{dv_x}{dr}\right\rvert\right)^{(n-1)}$

ChE Course	**Problem Name**	**Numerical Method Illustrated**	**Equations**
Mass Transfer or Separations	Slow Sublimation of a solid Sphere (POLYMATH Text 10.3) *C&S10-3 solution.pdf*	Solution of multiple ODE's with split boundary conditions	$\frac{d(r^2 N_A)}{dr} = 0$ $N_A = -\frac{C_T \mathfrak{D}_{AB}}{(1 - y_A)} \frac{dy_A}{dr}$
	Gas Absorption Column (Geankoplis Example 10.7-1) *concentratedSO2Ex10.7-1Solution.pdf*	Solution of multiple ODE's with split boundary conditions	$\frac{d(Gy)}{dz} = -K_y a A_c (y - y^*)$ $\frac{d(Lx)}{dz} = -K_y a A_c (y - y^*)$
Reaction Engineering	Catalytic Packed Bed *CREFoglerElements11-8.pdf*	Solution of multiple ODE's for mole, energy, and momentum balances	$\frac{dF_i}{dW} = -r_i$ $\frac{dT}{dW} = \frac{Ua(T_a - T)/\rho_b + \Delta H_{rxn} r_A}{\sum F_i C_{pi}}$ $\frac{dP}{dW} = -\frac{\rho_0}{\rho} \beta_0$
	Determination of rate expressions for a catalytic reaction (POLYMATH text 11.16) *C&S11.16 soln.pdf*	Example of the nonlinear regression of data with LHHW reaction rate	$r_A = \frac{-kK_A p_A}{1 + K_A p_A + K_B p_B}$
	Stiff Ordinary Differential Equations in Chemical Kinetics (POLYMATH text 6.2)	Solution of a stiff system of simultaneous ordinary differential eqauations.	Problem by Gear (1969)
Heat Transfer	Unsteady-state cooling of a sphere (POLYMATH Text 9.13) *C&S9-13CoolingSphereSoln.pdf both POLYMATH & Comsol Solutions*	Application of the numerical method of lines to solve a partial differential equation that involves the solution of simultaneous ordinary differential equations and explicit algebraic equations.	$\frac{\partial T}{\partial t} = \frac{1}{r^2 \rho C_p} \frac{\partial}{\partial r}\left(kr^2 \frac{\partial T}{\partial r}\right)$
Distillation or Separations	Batch Distillation of a benzene-toluene or water-Ethanol Mixture (POLYMATH text 6.7 and 12.10) *C&S6-7BatchDistillationSolution.pdf*	Solution of a differential-algebraic system of equations using the controlled integration technique or differentiated bubble point equation.	$\frac{dn_1}{dt} = -\dot{n}_1 = \dot{n}_T y_1$ $\frac{dn_T}{dt} = -\dot{n}_T$ $y_1 = K_1 x_1$ $error = 1 - K_1 x_1 - K_2 x_2$ $\frac{dT}{dx_1} = K_c(error)$

ChE Course	**Problem Name**	**Numerical Method Illustrated**	**Equations**
Process Control	Dynamics and Control of a Stirred Tank Heater (POLYMATH Text 13.6) *C&S13-6x.pol*	Solution of ODE's, generation of step functions, simulation of a proportional integral controller.	$\frac{dT}{dt} = \frac{WC_p(T_i - T) + q}{\rho V C_p}$ $\frac{dT_0}{dt} = \left[T - T_0 - \frac{\tau_d}{2}\left(\frac{dT}{dt}\right)\right]\frac{2}{\tau_d}$ $q = q_s + K_c(T_r - T_m) + \frac{K_c}{\tau_I}(errsum)$ $\frac{d(errsum)}{dt} = T_r - T_m$
Chemical Engineering Principles (ChE Intro Class)	Steady-state Material Balances on a Separation Train (POLYMATH Text 2.4)	Solution of simultaneous linear equations	Overall component balances
Thermodynamics	Molar Volume and Compressibility factor from Van Der Waals Equation (POLYMATH Text 2.1)	Solution of nonlinear algebraic equation	$\left(P + \frac{a}{V^2}\right)(V - b) = RT$ $a = \frac{27}{64}\left(\frac{R^2 T_c^2}{P_c}\right)$ $b = \frac{RT_c}{8P_c}$

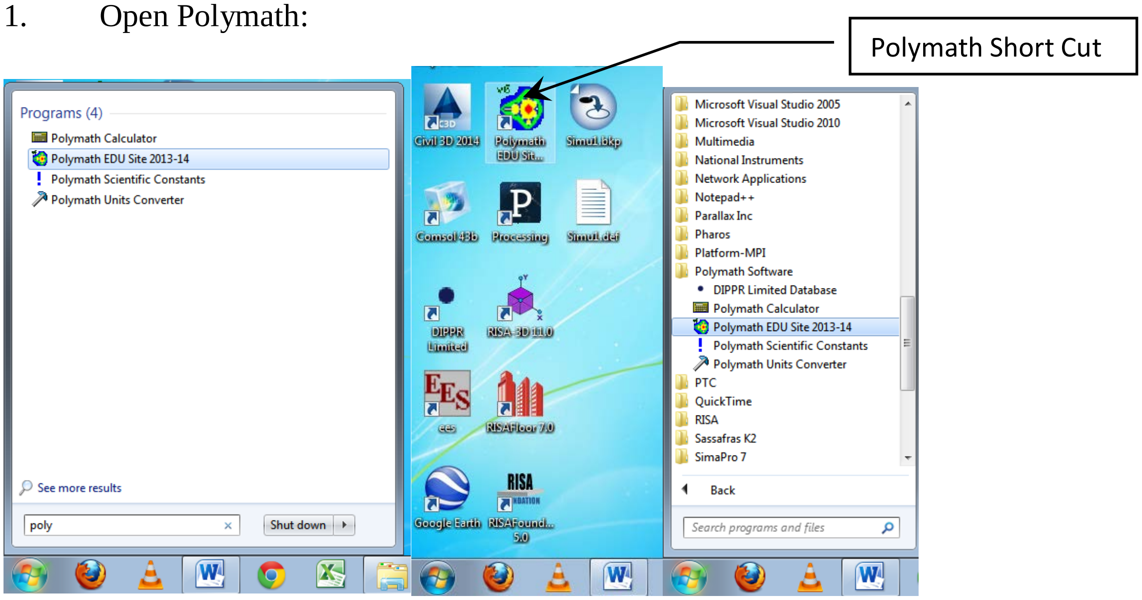
Introduction to POLYMATH based on Polymath and Excel Tutorial for Process Fluid Transport

---

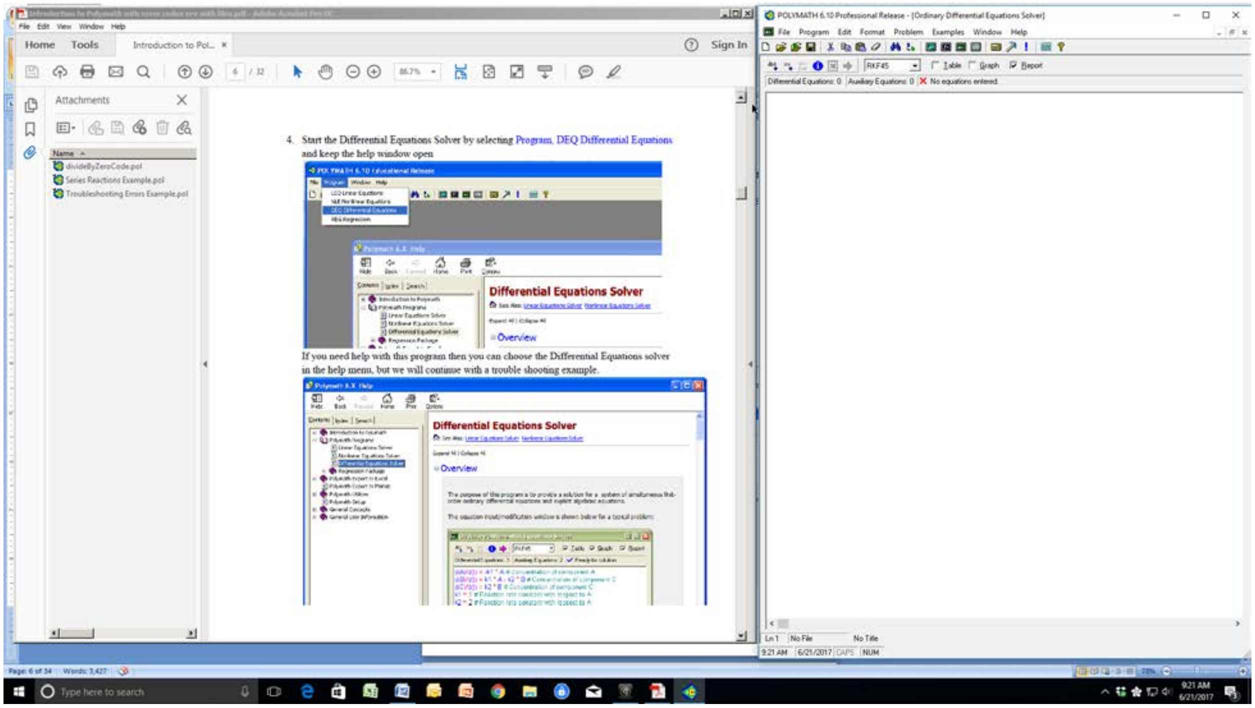
POLYMATH tutorial Objectives: A student will be able to

1. Enter and solve 3 differential equations for a batch chemical reactor problem .
2. Use the built-in dialog box buttons to enter a differential equation and its initial condition. This form is useful since you will not forget to enter the initial condition.
3. Prepare a word document that contains all required information for homework solution. Students must do more than just turn in a polymath program file. They must show how the model equations were derived, answer the questions, produce graphs and sample calculations.
4. How to copy POLYMATH output into an excel spreadsheet such that the produced output has headers in the first row.
5. use the trouble shooting DEQ Message list to determine that a variable has been defined more than once or has not been defined
6. Identify problems that cause a program to stop running such as a divide by zero error
7. Use the comment feature in the polymath program (#)
8. How to use an if - then - else statement

1. Open Polymath:



2. We suggest that you open two windows so that you can see this pdf file and the polymath program as shown below.

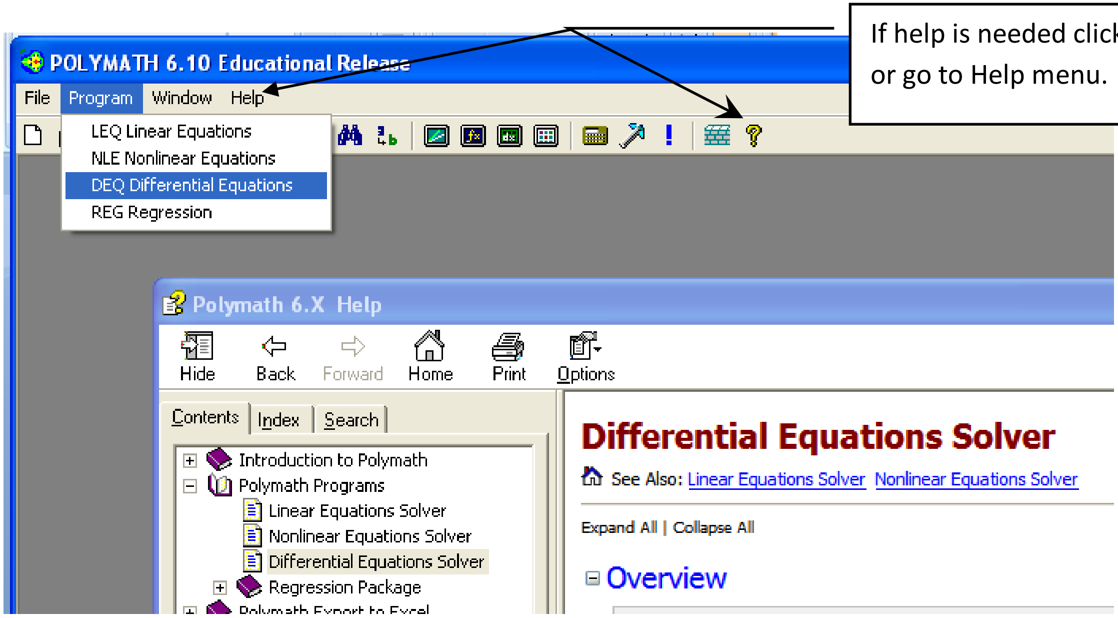


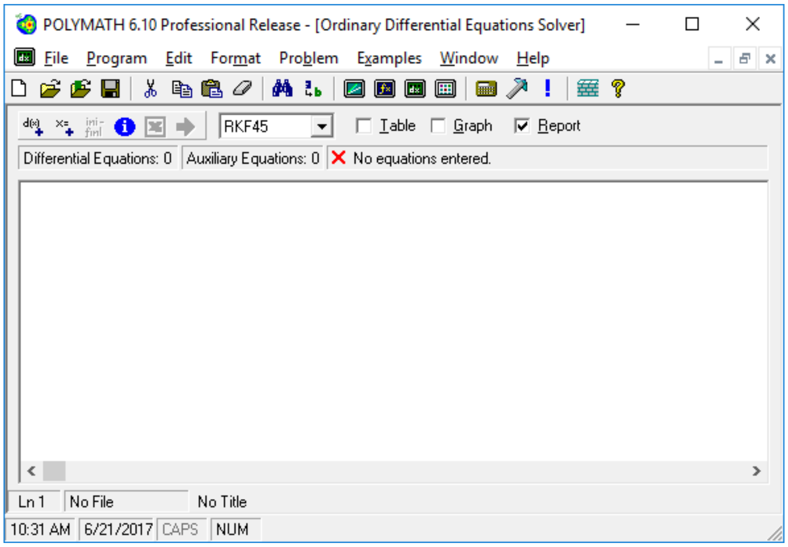
4. Start the Differential Equations Solver by selecting Program, DEQ Differential Equations and keep the help window open

If you need help with this program then you can choose the Differential Equations solver in the help menu, but we will continue with a trouble shooting example.

3. Start the Differential Equations Solver by selecting Program, DEQ Differential Equations and keep the help window open

If help is needed click on Question Mark or go to Help menu.





RKF45 ☐ Table ☐ Graph ☑ Report

Differential Equations: 0 | Auxiliary Equations: 0 | No equations entered.

Figure 1: Blank screen of Differential Equation Solver

4. Now you will create a simple POLYMATH file: Series Reactions Example.pol. We will enter the 3 differential Equations and supporting explicit algebraic equations similar to that given in the POLYMATH help example. (The only difference is that concentrations are defined as CA, CB and CC).

This example is based on a batch reactor with 2 simultaneous chemical reactions in series.

$$A \overset{k_1}{\rightarrow} B \overset{k_2}{\rightarrow} C$$

A component mole balance is constructed for each chemical species. Since this is a batch reactor, then the mole balances are differential equations. If this problem was an assigned homework problem then the first page of the problem would be a hand written setup of the problem on green engineering paper. This page would contain:

a) Setup of the component species mole balances including a diagram of the process (process flow diagram, pfd)

b) Initial conditions

c) Sample calculations showing that the correct units have been used and an order of magnitude estimate of the results.

For example the mole balances for A, B and C are given by

$$\frac{d(C_A)}{dt} = -k_1 C_A \tag{1}$$

$$\frac{d(C_B)}{dt} = k_1 C_A - k_2 C_B \tag{2}$$

$$\frac{d(C_C)}{dt} = k_2 C_B \tag{3}$$

The initial conditions in the batch reactor at t=0 min are $C_A = 1\ \mathrm{kmol/L}$, $C_B = 0\ \mathrm{kmol/L}$ and $C_C = 0\ \mathrm{kmol/L}$. These are known as initial values. The integration will proceed from 0 min to t=3 min. The rate constants are $k_1 = 1\ \mathrm{min}^{-1}$ and $k_2 = 2\ \mathrm{min}^{-1}$.

Sample calculations of all equations are required to be submitted on green engineering paper. These calculations will help you to troubleshoot your program. Sample calculations for explicit equations should be straight forward always showing the number and units. For differential equations I suggest that you show an **order of magnitude** estimate as the sample calculation.

For this example for the batch reactor mole balance using the initial conditions the initial change in concentration of A with time is:

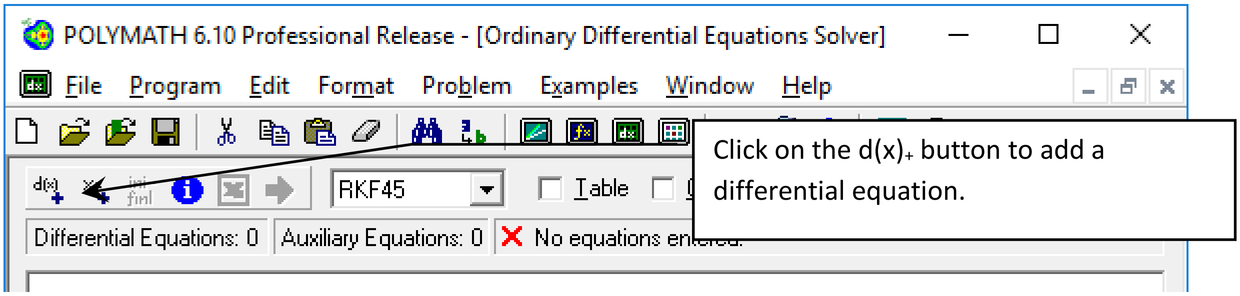
$$\left.\frac{d(C_A)}{dt}\right|_{t=0} = -1\,min^{-1}\left(1\frac{kmol}{L}\right) = -1\,kmol/(L\,\text{min}) \qquad (4)$$

An estimate of the value of concentration after 1 minute would be
(NOTICE that this is not a correct integration of the differential equation. This is ONLY an ESTIMATE and a check on the units. To check for order of magnitude changes an assumption is made that the rate is constant.)

$$\int_{C_A=1\,kmol/L}^{C_A} d(C_A) \sim \int_{t=0min}^{1\,min} -1\,kmol/(L\,\text{min})dt = C_A - 1kmol/L = -1\,kmol/L \qquad (5)$$

The above result gives the final value of the concentration of A to be zero. In other words if the reaction rate was at 1 kmol/(L min) for 1 minute, then there would be no reactant A left. It is then up to the student doing the problem to evaluate if this reaction rate is what was specified or should the rate be 10 times lower. The above equations do not need to be typed, and can be written by hand and then scanned in B&W using your phone with an free App such as CamScanner. This scan will then be inserted into the word document that will be submitted on Blackboard.

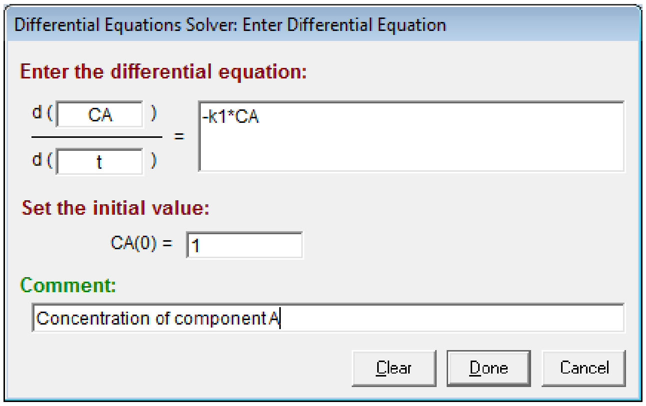
5. Of special note is the usefulness of the wizard menu's in entering a differential equation. This will write the equation using the proper syntax. For example to enter the first differential equation click on the d(x)$_+$ button. d(CA) / d(t) = -k1*CA #Concentration of component A and then the initial condition that concentration of A at t=0 is 1.



Click on the d(x)$_+$ button to add a differential equation.

Figure 2: Add a differential Equation

6. Fill out the form shown below and then select done.



Enter the differential equation:

d(CA) / d(t) = -k1*CA

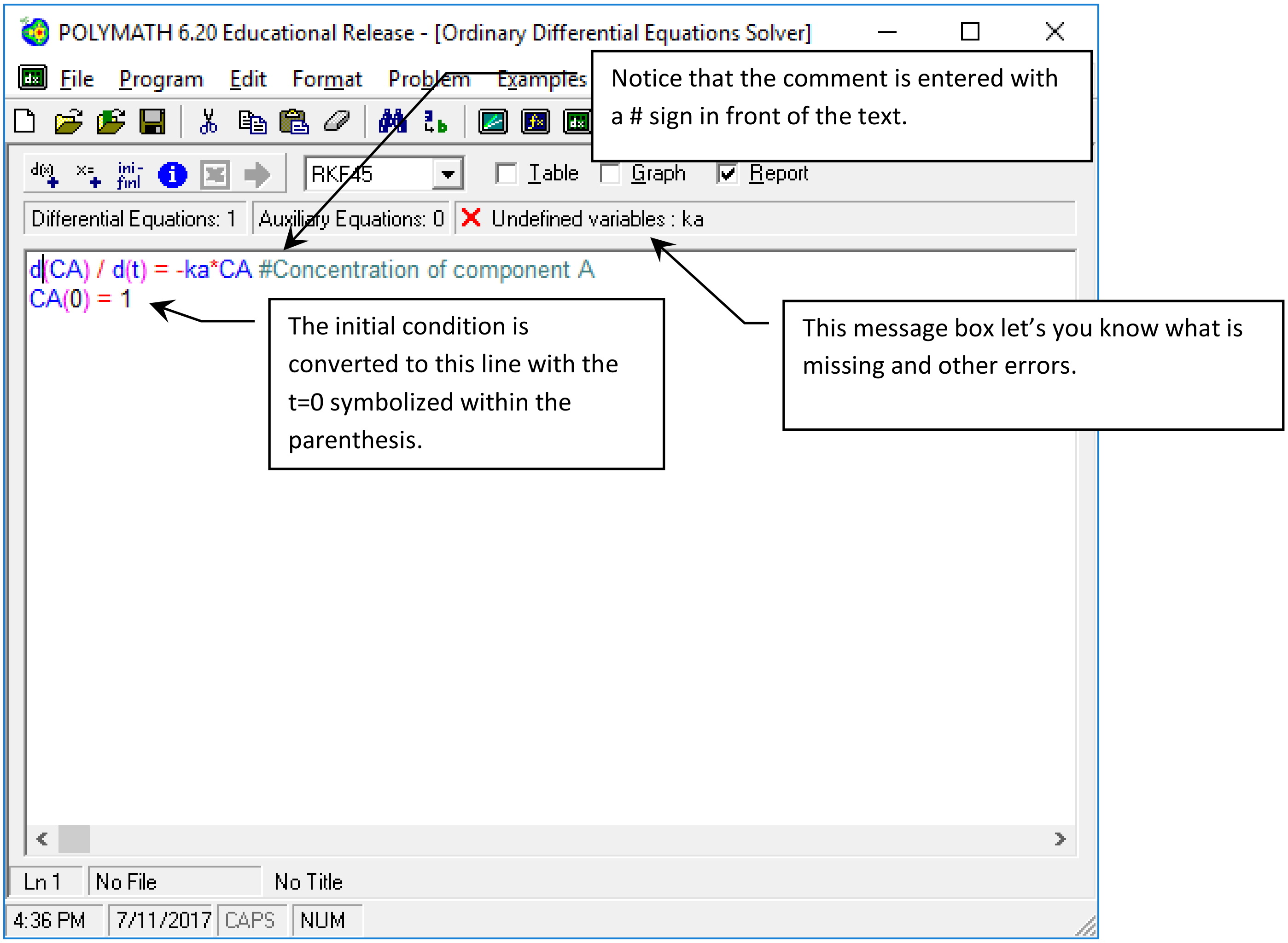
Set the initial value:

CA(0) = 1

Comment:

Concentration of component A

Figure 3: Fill out the menu screen as shown



RKF45 ☐ Table ☐ Graph ☑ Report

Differential Equations: 1 | Auxiliary Equations: 0 | Undefined variables : ka

```
d(CA) / d(t) = -ka*CA #Concentration of component A
CA(0) = 1
```

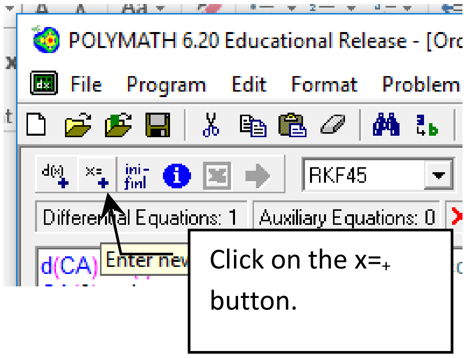
Notice that the comment is entered with a # sign in front of the text.

The initial condition is converted to this line with the t=0 symbolized within the parenthesis.

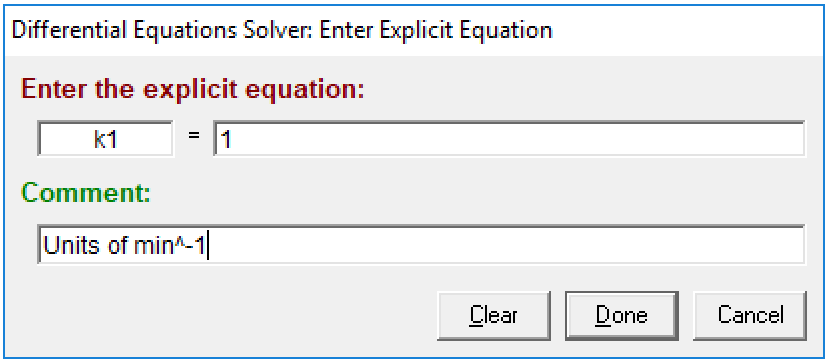
This message box let's you know what is missing and other errors.

Figure 4: Result of filling out the menu for one differential equation

7. Notice that an error has appeared stating that there is an undefined variable, k1. To remove this error you can add the explicit equation k1=1 min^{-1}. Open up the wizard for an explicit equation



Click on the x=+ button.


Differential Equations Solver: Enter Explicit Equation

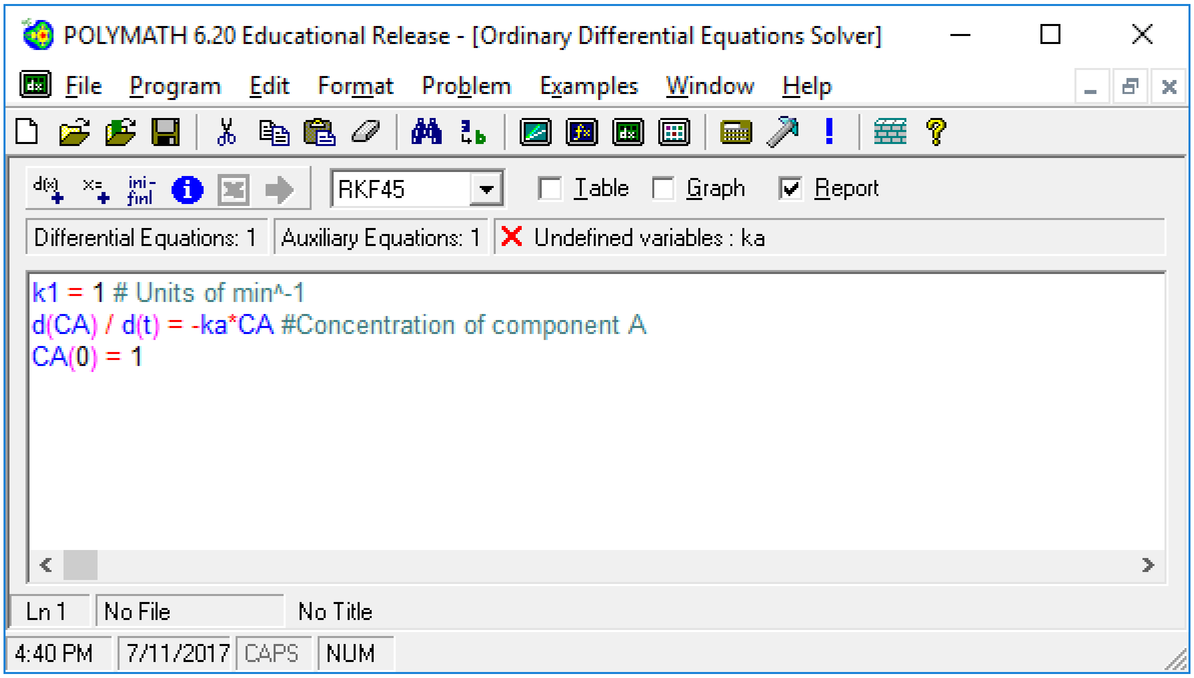
Enter the explicit equation:

k1 = 1

Comment:

Units of min^-1

Figure 5: Explicit Equation Entry Form



RKF45 ☐ Table ☐ Graph ☑ Report

Differential Equations: 1 Auxiliary Equations: 1 ✗ Undefined variables : ka

```
k1 = 1 # Units of min^-1
d(CA) / d(t) = -ka*CA #Concentration of component A
CA(0) = 1
```

Ln 1 No File No Title

4:40 PM 7/11/2017 CAPS NUM

Figure 6: Result of entering explicit equation

8. Enter the second differential equation using the wizard. Notice that the second time that you open this wizard the independent variable of time is already entered.

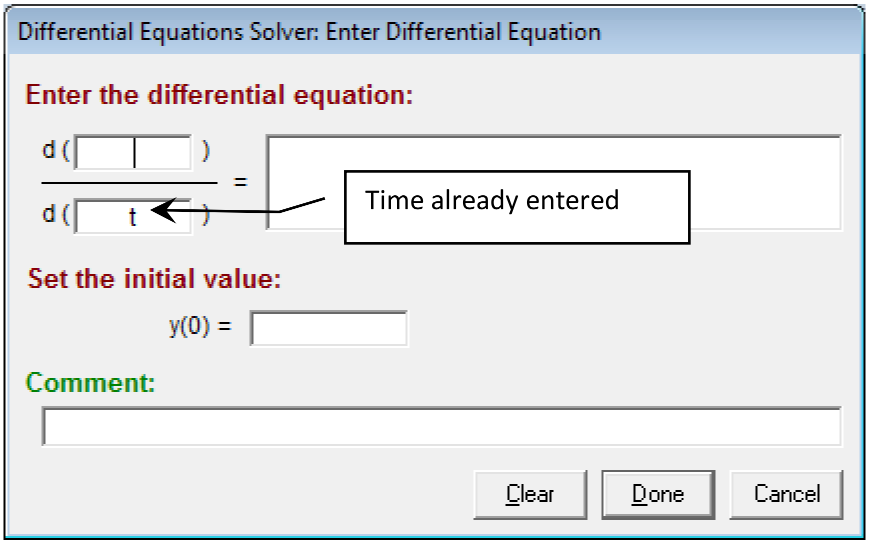
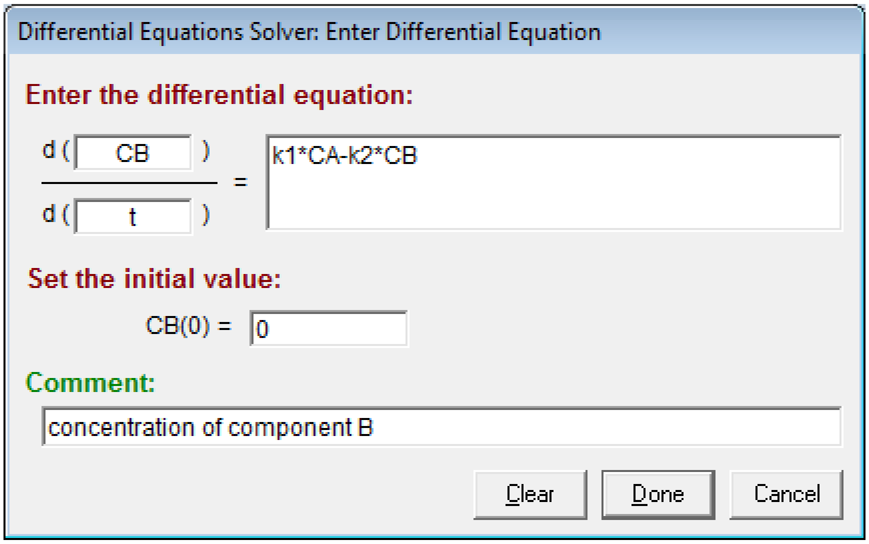


Figure 7: Second use of ODE wizard



d(CB) / d(t) = k1*CA-k2*CB

Set the initial value:

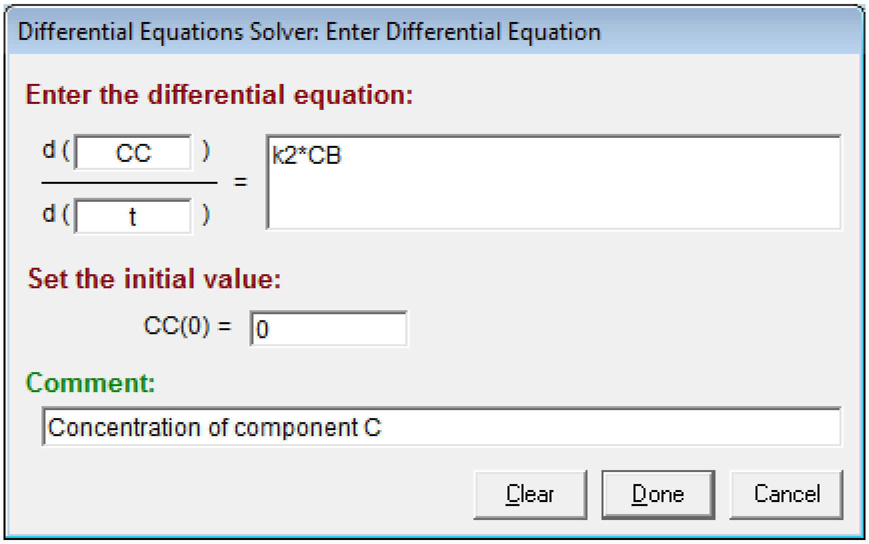
CB(0) = 0

Comment:

concentration of component B

Figure 8: Second ODE

9. Finally enter the 3rd ODE



d(CC) / d(t) = k2*CB

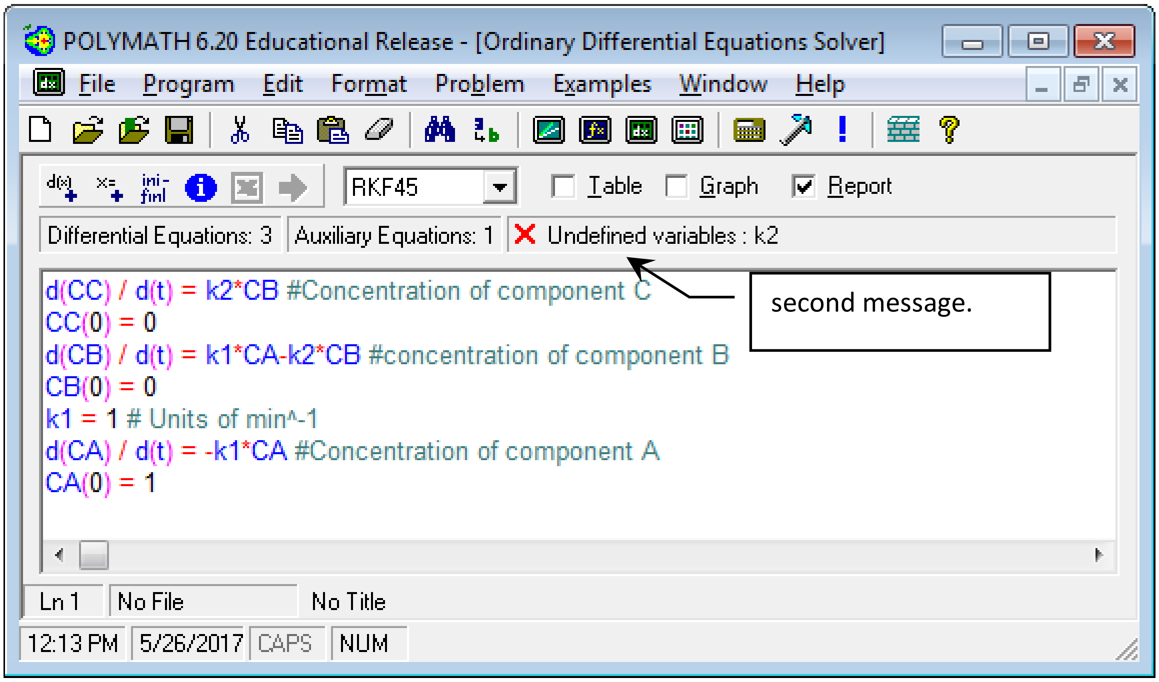
Set the initial value:

CC(0) = 0

Comment:

Concentration of component C

10. The result of entering the 3 ODE's is



RKF45 ☐ Table ☐ Graph ☑ Report

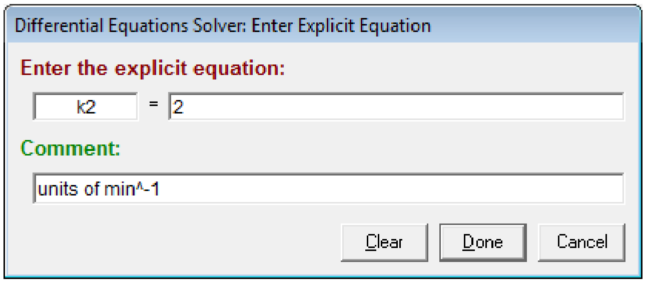
Differential Equations: 3 | Auxiliary Equations: 1 | ✗ Undefined variables : k2

```
d(CC) / d(t) = k2*CB #Concentration of component C
CC(0) = 0
d(CB) / d(t) = k1*CA-k2*CB #concentration of component B
CB(0) = 0
k1 = 1 # Units of min^-1
d(CA) / d(t) = -k1*CA #Concentration of component A
CA(0) = 1
```

Ln 1 | No File | No Title
12:13 PM | 5/26/2017 | CAPS | NUM

second message.

11. To remove the error marked above add the value of the second rate constant

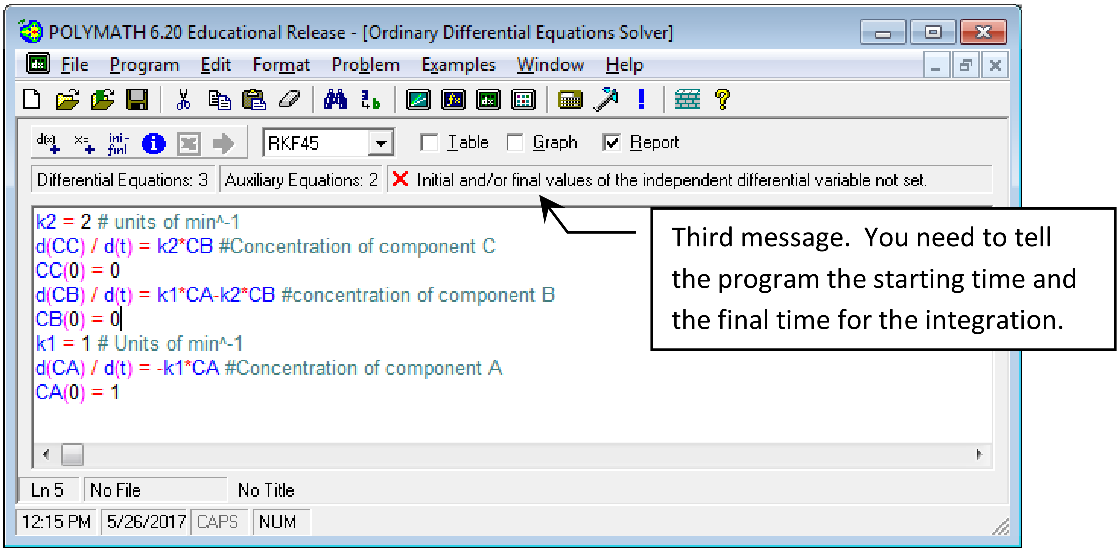

Differential Equations Solver: Enter Explicit Equation

Enter the explicit equation:

k2 = 2

Comment:

units of min^-1



RKF45 ☐ Table ☐ Graph ☑ Report

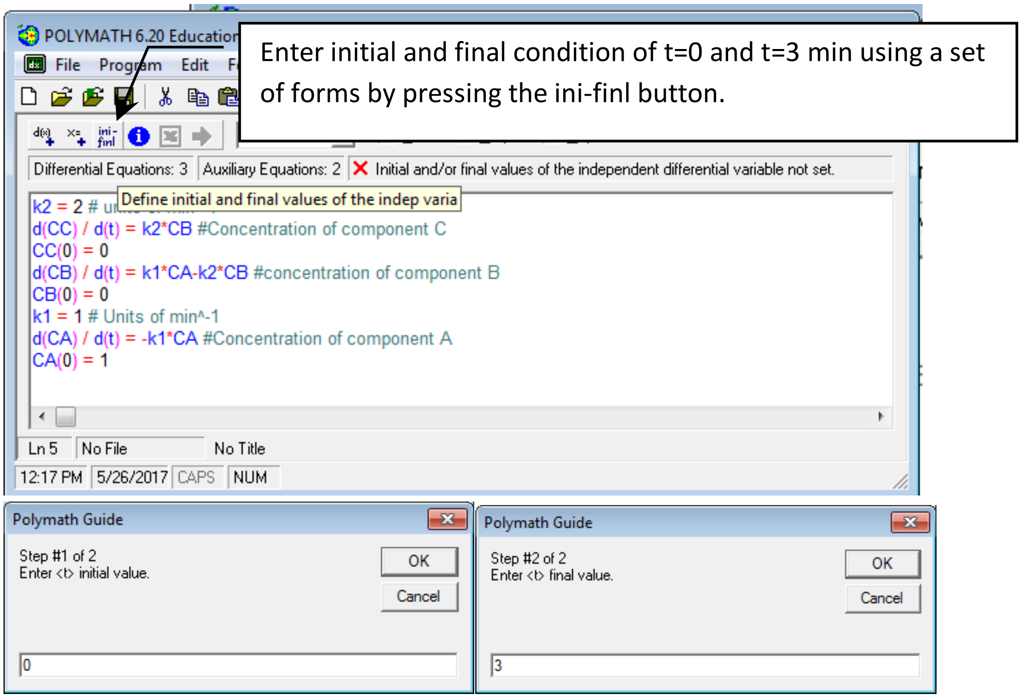
Differential Equations: 3 | Auxiliary Equations: 2 | ✗ Initial and/or final values of the independent differential variable not set.

```
k2 = 2 # units of min^-1
d(CC) / d(t) = k2*CB #Concentration of component C
CC(0) = 0
d(CB) / d(t) = k1*CA-k2*CB #concentration of component B
CB(0) = 0
k1 = 1 # Units of min^-1
d(CA) / d(t) = -k1*CA #Concentration of component A
CA(0) = 1
```

Ln 5 | No File | No Title
12:15 PM | 5/26/2017 | CAPS | NUM

Third message. You need to tell the program the starting time and the final time for the integration.

12. To remove the final marked error go to the ini-finl button to enter the initial and final values for the integration of the time variable. In this case t=0 to t=3 min.



Enter initial and final condition of t=0 and t=3 min using a set of forms by pressing the ini-finl button.

Differential Equations: 3 | Auxiliary Equations: 2 | ✕ Initial and/or final values of the independent differential variable not set.

```
k2 = 2 # u
d(CC) / d(t) = k2*CB #Concentration of component C
CC(0) = 0
d(CB) / d(t) = k1*CA-k2*CB #concentration of component B
CB(0) = 0
k1 = 1 # Units of min^-1
d(CA) / d(t) = -k1*CA #Concentration of component A
CA(0) = 1
```

Ln 5 | No File | No Title

12:17 PM | 5/26/2017 | CAPS | NUM

Polymath Guide

Step #1 of 2
Enter <t> initial value.

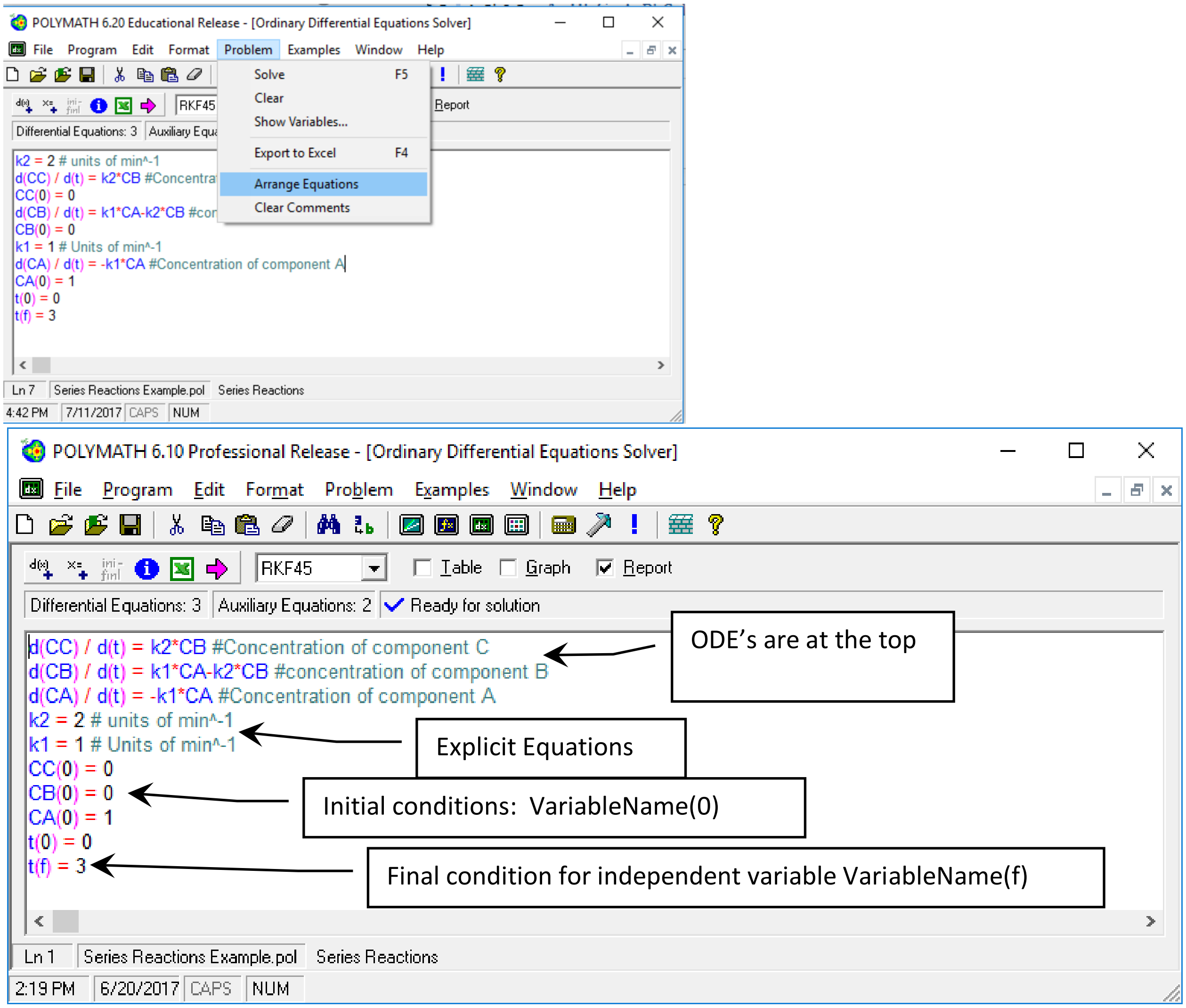
0

Polymath Guide

Step #2 of 2
Enter <t> final value.

3

13. Now the program can be run since the purple arrow appears, but I recommend that you use the Arrange equation feature to order your equations. This will be easier for your professor to troubleshoot and/or grade when put in this order.



```
k2 = 2 # units of min^-1
d(CC) / d(t) = k2*CB #Concentra
CC(0) = 0
d(CB) / d(t) = k1*CA-k2*CB #con
CB(0) = 0
k1 = 1 # Units of min^-1
d(CA) / d(t) = -k1*CA #Concentration of component A
CA(0) = 1
t(0) = 0
t(f) = 3
```

```
d(CC) / d(t) = k2*CB #Concentration of component C
d(CB) / d(t) = k1*CA-k2*CB #concentration of component B
d(CA) / d(t) = -k1*CA #Concentration of component A
k2 = 2 # units of min^-1
k1 = 1 # Units of min^-1
CC(0) = 0
CB(0) = 0
CA(0) = 1
t(0) = 0
t(f) = 3
```

ODE's are at the top

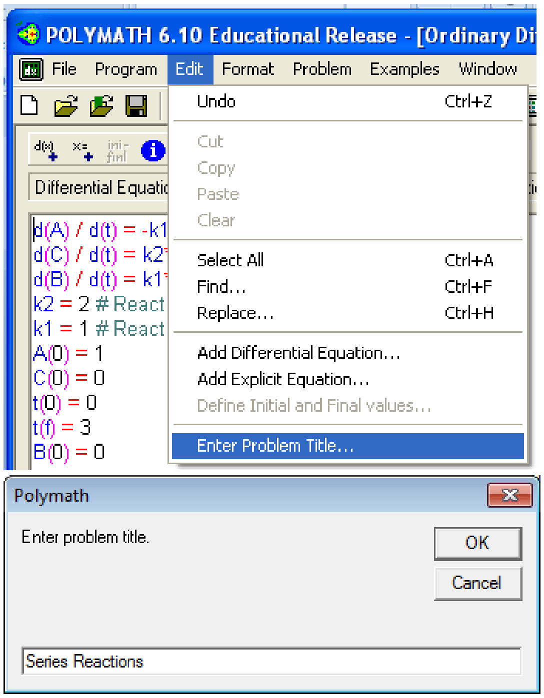
Explicit Equations

Initial conditions: VariableName(0)

Final condition for independent variable VariableName(f)

Figure 9: Result of the arrangement of equations (the ordering of the initial conditions may be different to that shown above)

14. Now enter a problem title by selecting Edit, Enter Problem Title…

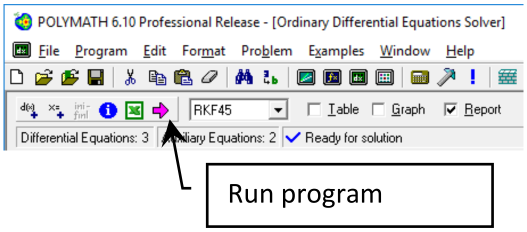


```
d(A) / d(t) = -k1
d(C) / d(t) = k2
d(B) / d(t) = k1
k2 = 2 # React
k1 = 1 # React
A(0) = 1
C(0) = 0
t(0) = 0
t(f) = 3
B(0) = 0
```

Enter problem title.

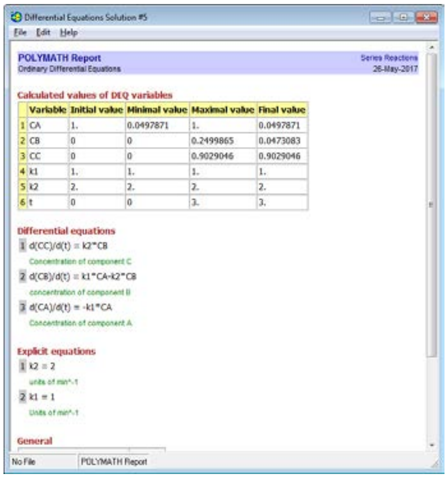
Series Reactions

15. Next save the program with a file name and then run the program by pressing on the pink arrow. The default output is the POLYMATH Report.



RKF45 ☐ Table ☐ Graph ☑ Report

Differential Equations: 3 Auxiliary Equations: 2 ✓ Ready for solution



POLYMATH Report
Ordinary Differential Equations
Series Reactions
26-May-2017

Calculated values of DEQ variables

	Variable	Initial value	Minimal value	Maximal value	Final value
1	CA	1.	0.0497871	1.	0.0497871
2	CB	0	0	0.2499865	0.0473083
3	CC	0	0	0.9029046	0.9029046
4	k1	1.	1.	1.	1.
5	k2	2.	2.	2.	2.
6	t	0	0	3.	3.

Differential equations

1 d(CC)/d(t) = k2*CB
Concentration of component C
2 d(CB)/d(t) = k1*CA-k2*CB
concentration of component B
3 d(CA)/d(t) = -k1*CA
Concentration of component A

Explicit equations

1 k2 = 2
units of min^-1
2 k1 = 1
Units of min^-1

General

Figure 10: Default output from POLYMATH

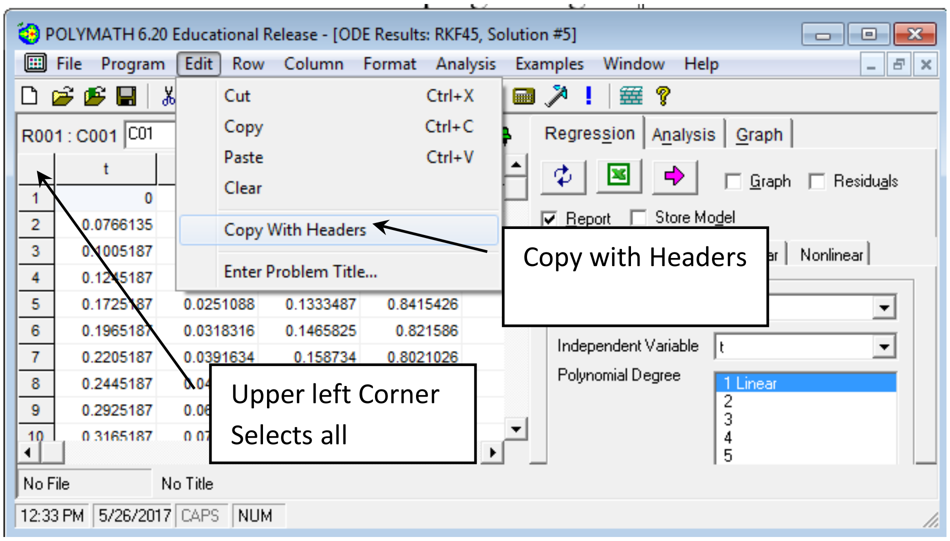
16. Polymath has a program for graphs but you can also produce a graph in excel. In this case you should do the following

1. Select the Table output button

2. Run the program again

3. Select the table window

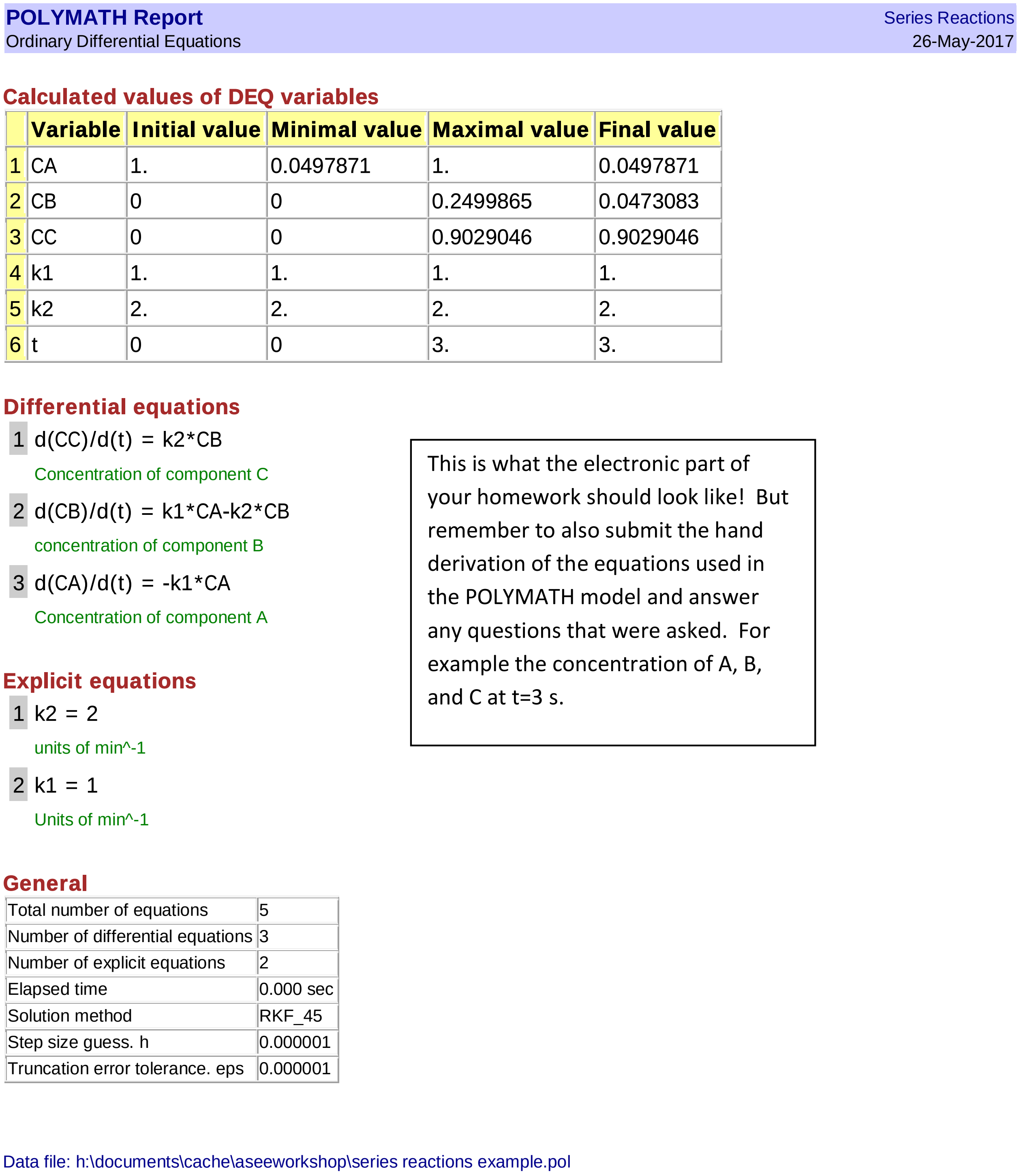
4. Click on the upper left corner of the table (similar to excel)



5. Then select Edit, Copy With Headers. (This will copy the names of the variables as well as the numbers)

6. Paste this into an excel spreadsheet and produce a graph with all titles given and labels.

For homework assignments with POLYMATH I have students paste a pdf of the handwritten derivation of equations, the POLYMATH Report and the graph into a word document.



POLYMATH Report Series Reactions
Ordinary Differential Equations 26-May-2017

Calculated values of DEQ variables

	Variable	Initial value	Minimal value	Maximal value	Final value
1	CA	1.	0.0497871	1.	0.0497871
2	CB	0	0	0.2499865	0.0473083
3	CC	0	0	0.9029046	0.9029046
4	k1	1.	1.	1.	1.
5	k2	2.	2.	2.	2.
6	t	0	0	3.	3.

Differential equations

1 d(CC)/d(t) = k2*CB

Concentration of component C

2 d(CB)/d(t) = k1*CA-k2*CB

concentration of component B

3 d(CA)/d(t) = -k1*CA

Concentration of component A

Explicit equations

1 k2 = 2

units of min^-1

2 k1 = 1

Units of min^-1

General

Total number of equations	5
Number of differential equations	3
Number of explicit equations	2
Elapsed time	0.000 sec
Solution method	RKF_45
Step size guess. h	0.000001
Truncation error tolerance. eps	0.000001

Data file: h:\documents\cache\aseeworkshop\series reactions example.pol

This is what the electronic part of your homework should look like! But remember to also submit the hand derivation of the equations used in the POLYMATH model and answer any questions that were asked. For example the concentration of A, B, and C at t=3 s.

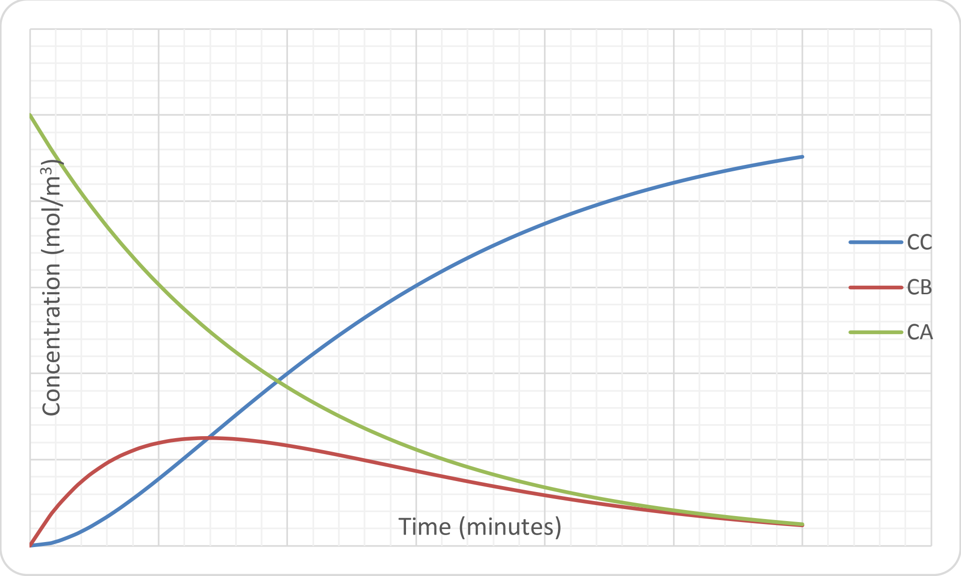


Figure 11: Concentration Profiles of a Series Reaction

Troubleshooting Example

As you can see POLYMATH is a very easy program to use. In the previous batch reactor example you probably produced a POLYMATH Report in less than 5 minutes (or if you did it a second time it would be even shorter). Some of the problems that you solve will be more complex than this and it is useful to see how to avoid errors in placing your model equations into polymath. Of course one of the biggest errors that students make is that they don't write out the equations. In addition they don't place numbers with units in them to make sure that all the units are consistent. This is probably the biggest error done by students. Numbers and units!!! Unfortunately POLYMATH can not help the student with errors in units other than putting the units in comment fields. What follows is the errors that POLYMATH can identify.

17. Common errors by students in using Polymath software:

    1. What the user thinks are the same variables, but the computer uses them as different variables. This happens when you incorrectly spell a variable name or you do not match upper or lower cases. (e.g. $Tau \neq tau$)

    2. Defining a variable more than once. Once a variable is on the left hand side of the equals sign then it is considered defined by the program.

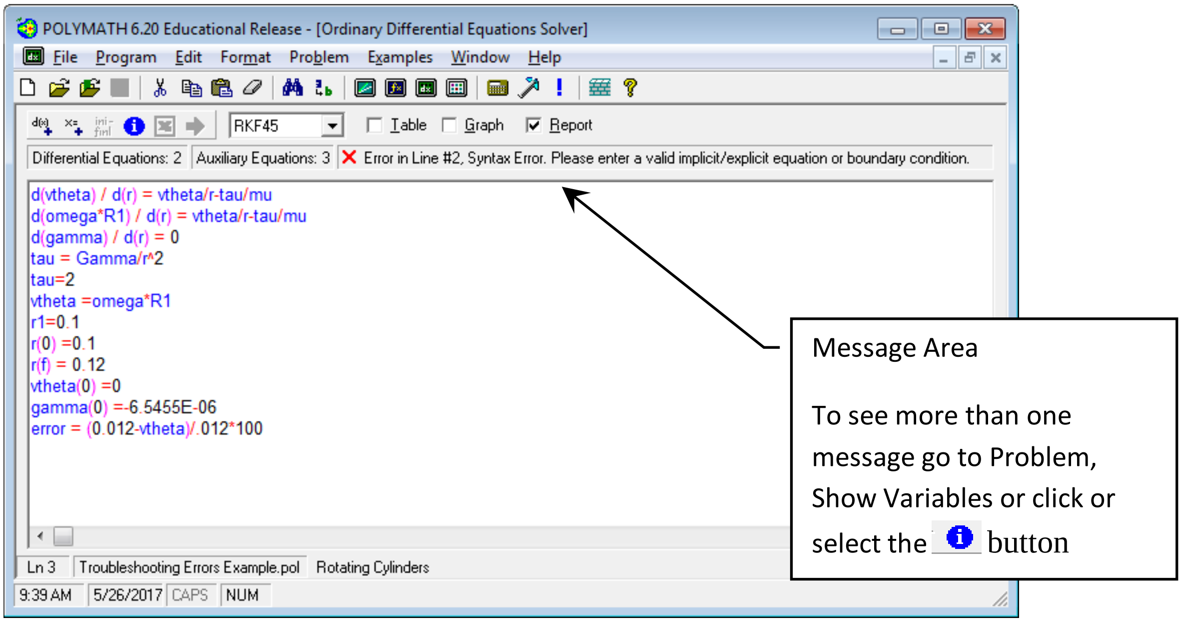
    3. Dividing by zero

    4. Using too many parenthesis. POLYMATH uses the standard order of operators which is exponent, multiplication/division, addition/subtraction: ^, (* or /), (+ or -) which is invoked working from left to right in an expression.

Cut and paste the following program in the POLYMATH ODE Solver or load the program troubleshooting errors example.pol. Examine each of these errors and then correct them as directed in the error explanation below:

```
d(vtheta) / d(r) = vtheta/r-tau/mu
d(gamma) / d(r) = 0
tau = Gamma/r^2
tau=2
vtheta =omega*R1
r1=0.1
r(0) =0.1
r(f) = 0.12
vtheta(0) =0
gamma(0) =-6.5455E-06
error = (0.012-vtheta)/.012*100
```

18. To see the errors in your program look in the message area or alternatively to see a full list: Select Problem from the Polymath menu and then Show Variables… to see the following errors in this program. Alternatively you could just press on the information button ℹ



RKF45 ☐ Table ☐ Graph ☑ Report

Differential Equations: 2 | Auxiliary Equations: 3 | ✗ Error in Line #2, Syntax Error. Please enter a valid implicit/explicit equation or boundary condition.

```
d(vtheta) / d(r) = vtheta/r-tau/mu
d(omega*R1) / d(r) = vtheta/r-tau/mu
d(gamma) / d(r) = 0
tau = Gamma/r^2
tau=2
vtheta =omega*R1
r1=0.1
r(0) =0.1
r(f) = 0.12
vtheta(0) =0
gamma(0) =-6.5455E-06
error = (0.012-vtheta)/.012*100
```

Ln 3 | Troubleshooting Errors Example.pol | Rotating Cylinders

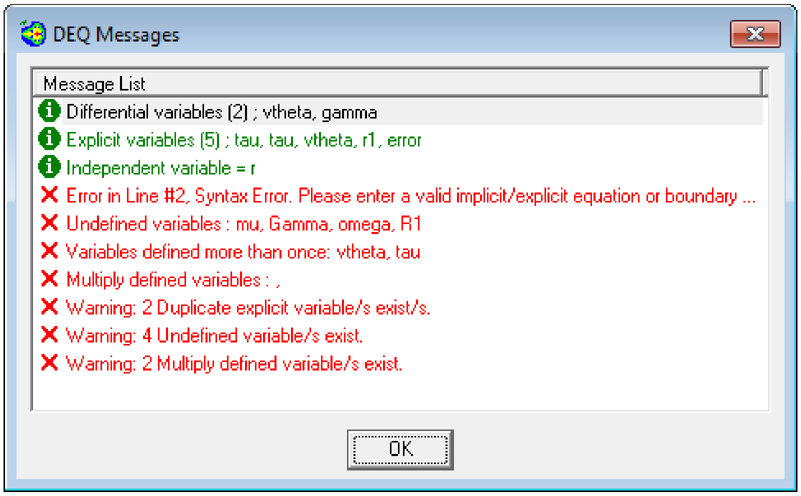
9:39 AM | 5/26/2017 | CAPS | NUM

Message Area

To see more than one message go to Problem, Show Variables or click or select the ℹ button

Figure 12: Original Code

Figure 13: Request to show all of the errors by selecting Problem and then Show Variables



```
ℹ Differential variables (2) ; vtheta, gamma
ℹ Explicit variables (5) ; tau, tau, vtheta, r1, error
ℹ Independent variable = r
✗ Error in Line #2, Syntax Error. Please enter a valid implicit/explicit equation or boundary ...
✗ Undefined variables : mu, Gamma, omega, R1
✗ Variables defined more than once: vtheta, tau
✗ Multiply defined variables : ,
✗ Warning: 2 Duplicate explicit variable/s exist/s.
✗ Warning: 4 Undefined variable/s exist.
✗ Warning: 2 Multiply defined variable/s exist.
```

Figure 14: Original listing of variables and errors

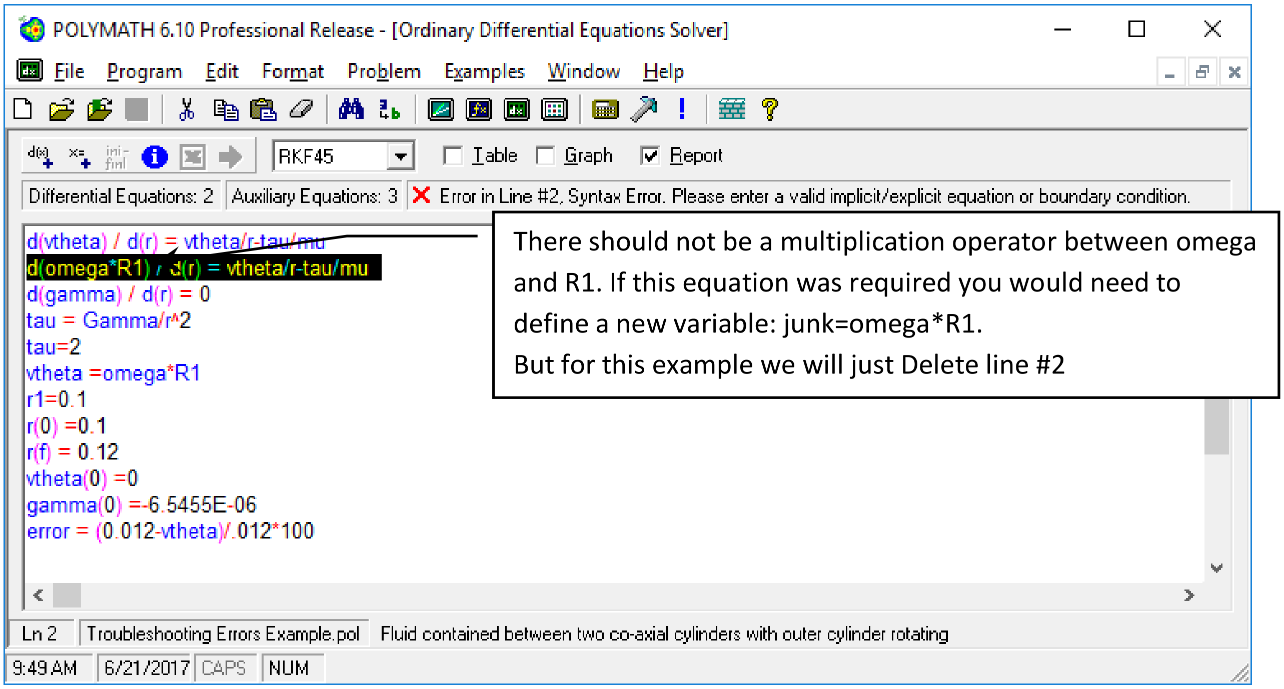
Explanation of Errors in Message:

Differential variables (2); vtheta, gamma There are three ordinary differential equations, but one of the equations has a syntax error and is not recognized. See below.

Explicit variables (5); tau, tau, vtheta, r1, error There are 5 explicit (variable =) equations. Notice that it has tau twice which means that it was defined more than once as an explicit equation. This is an error that is also mentioned in the "Multiple (This is a spelling error!) defined variables"

Independent variable = r The ODE's are with respect to only one variable and that variable is r. This is OK

19. Now for the RED X's **X** Error in Line #2, Syntax Error. Please enter a valid implicit/explicit equation or boundary condition. d(omega*R1) / d(r) = vtheta/r-tau/mu The error in this equation is the use of an operator on the left hand side which is not allowed. If this equation was required you would need to define a new variable: junk=omega*R1. The correct form is using the line #1 ODE so to correct this error we will delete line 2.



RKF45 ☐ Table ☐ Graph ☑ Report

Differential Equations: 2 | Auxiliary Equations: 3 | ✕ Error in Line #2, Syntax Error. Please enter a valid implicit/explicit equation or boundary condition.

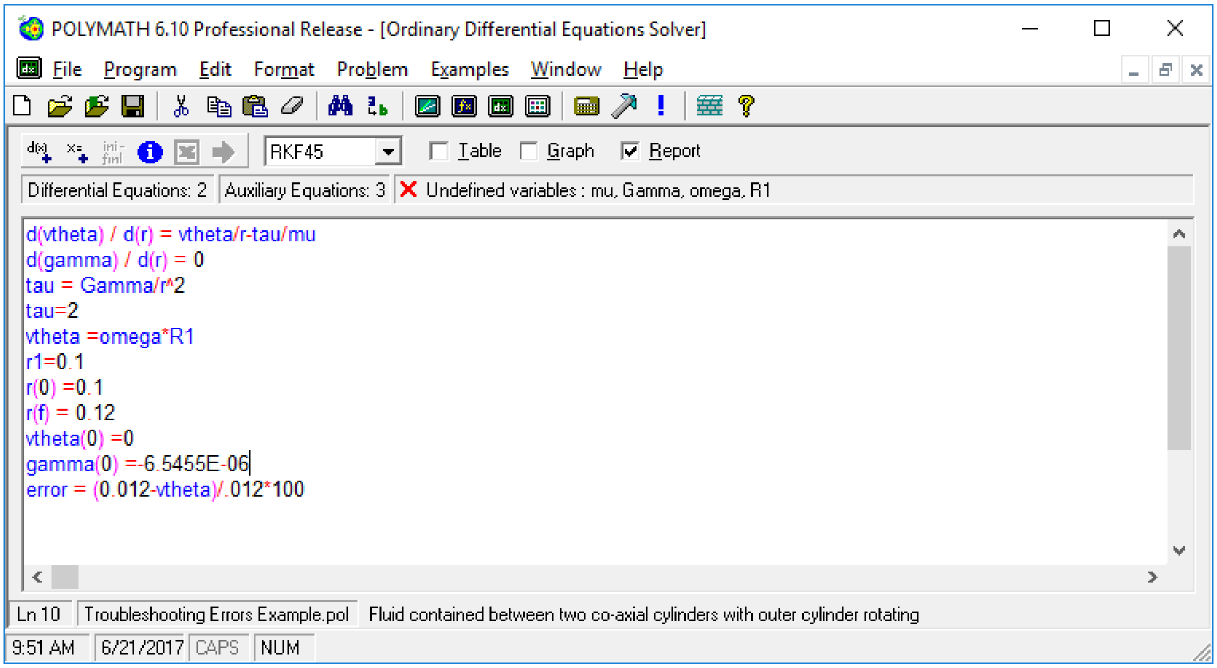
```
d(vtheta) / d(r) = vtheta/r-tau/mu
d(omega*R1) / d(r) = vtheta/r-tau/mu
d(gamma) / d(r) = 0
tau = Gamma/r^2
tau=2
vtheta =omega*R1
r1=0.1
r(0) =0.1
r(f) = 0.12
vtheta(0) =0
gamma(0) =-6.5455E-06
error = (0.012-vtheta)/.012*100
```

There should not be a multiplication operator between omega and R1. If this equation was required you would need to define a new variable: junk=omega*R1.
But for this example we will just Delete line #2

Ln 2 | Troubleshooting Errors Example.pol | Fluid contained between two co-axial cylinders with outer cylinder rotating

9:49 AM | 6/21/2017 | CAPS | NUM

Figure 15: Error in line #2



RKF45 ☐ Table ☐ Graph ☑ Report

Differential Equations: 2 | Auxiliary Equations: 3 | ✗ Undefined variables : mu, Gamma, omega, R1

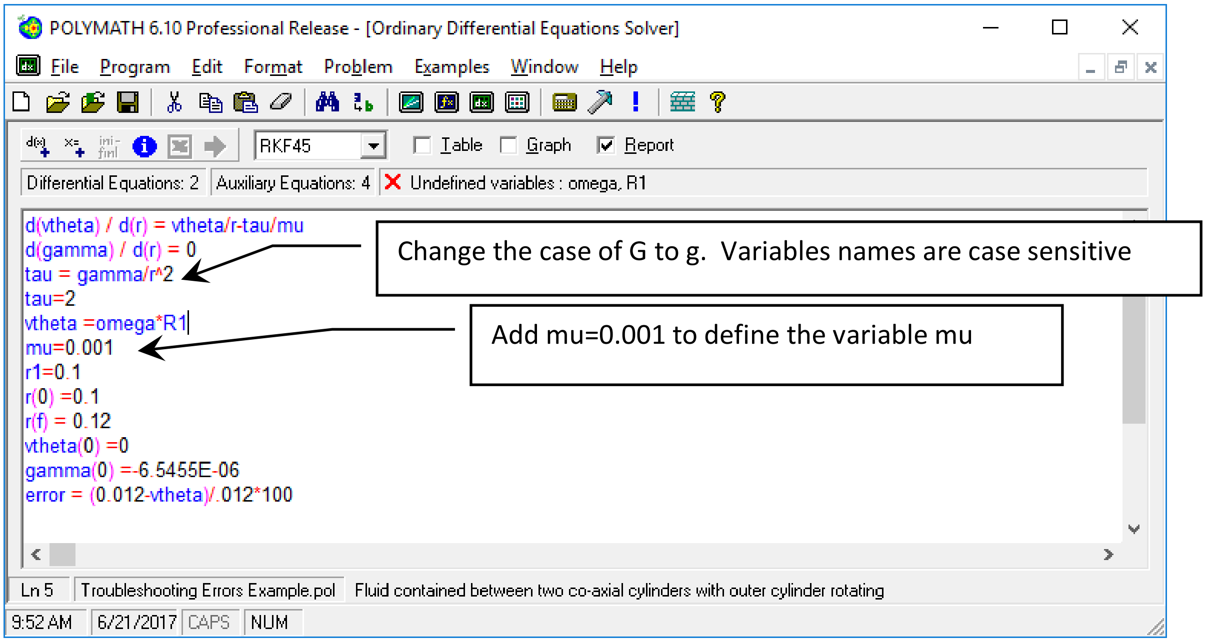
```
d(vtheta) / d(r) = vtheta/r-tau/mu
d(gamma) / d(r) = 0
tau = Gamma/r^2
tau=2
vtheta =omega*R1
r1=0.1
r(0) =0.1
r(f) = 0.12
vtheta(0) =0
gamma(0) =-6.5455E-06
error = (0.012-vtheta)/.012*100
```

Ln 10 | Troubleshooting Errors Example.pol | Fluid contained between two co-axial cylinders with outer cylinder rotating

9:51 AM | 6/21/2017 | CAPS | NUM

Figure 16: Resulting code after deleting line #2 from above. To see this code you need to click on another line and the message box will update to the next error.

20. ✗ Undefined variables : mu, Gamma, omega, R1 The variable mu is used in line 1, but not defined. **You need to add an explicit equation mu=0.001**. The reason that Gamma is undefined is that it never appears on the left hand side of the equal sign, but the variable gamma does! **To correct this error change the case of the g in Gamma**. The remaining undefined variables we will correct in the next error.



RKF45 ☐ Table ☐ Graph ☑ Report

Differential Equations: 2 | Auxiliary Equations: 4 | ✗ Undefined variables : omega, R1

```
d(vtheta) / d(r) = vtheta/r-tau/mu
d(gamma) / d(r) = 0
tau = gamma/r^2
tau=2
vtheta =omega*R1
mu=0.001
r1=0.1
r(0) =0.1
r(f) = 0.12
vtheta(0) =0
gamma(0) =-6.5455E-06
error = (0.012-vtheta)/.012*100
```

Change the case of G to g. Variables names are case sensitive

Add mu=0.001 to define the variable mu

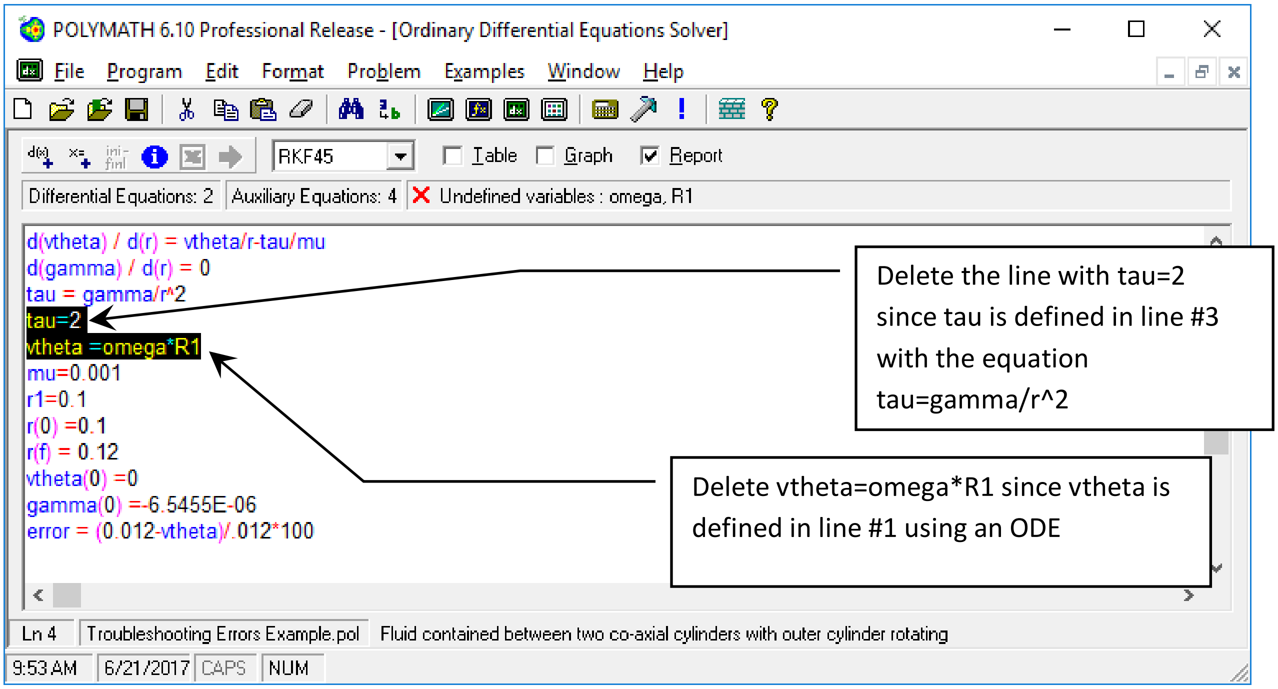
Ln 5 | Troubleshooting Errors Example.pol | Fluid contained between two co-axial cylinders with outer cylinder rotating

9:52 AM | 6/21/2017 | CAPS | NUM

Figure 17: Result after adding definition of mu and change case of G to g

21. X Variables defined more than once: vtheta, tau There are two equations that have tau = on lines 3 and 4. The variable vtheta has been defined twice; once in line #1 as an ordinary differential equation and the second time in line #5 shown below. Equation 5 (vtheta =omega*R1) must be deleted in this problem. A variable is defined by having it on the left hand side of the equal sign. Additionally there can only be one variable on the left hand side of an equal sign.

Also on line #4 **you should delete the equation (**tau = 2**)** since it is already defined in the equation above it (#3).



Differential Equations: 2 Auxiliary Equations: 4 ✗ Undefined variables : omega, R1

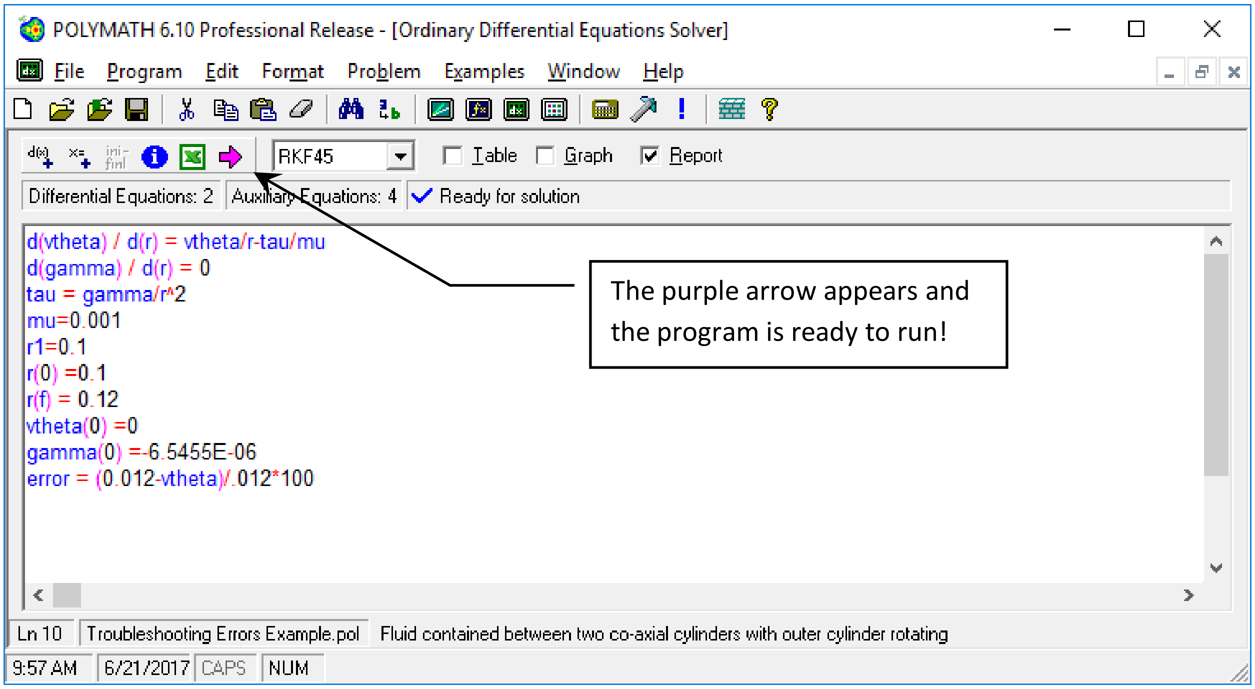
```
d(vtheta) / d(r) = vtheta/r-tau/mu
d(gamma) / d(r) = 0
tau = gamma/r^2
tau=2
vtheta =omega*R1
mu=0.001
r1=0.1
r(0) =0.1
r(f) = 0.12
vtheta(0) =0
gamma(0) =-6.5455E-06
error = (0.012-vtheta)/.012*100
```

Delete the line with tau=2 since tau is defined in line #3 with the equation tau=gamma/r^2

Delete vtheta=omega*R1 since vtheta is defined in line #1 using an ODE

Figure 18: tau and vtheta corrections

22. **Now the program is ready to run:**



RKF45 ☐ Table ☐ Graph ☑ Report

Differential Equations: 2 | Auxiliary Equations: 4 | ✓ Ready for solution

```
d(vtheta) / d(r) = vtheta/r-tau/mu
d(gamma) / d(r) = 0
tau = gamma/r^2
mu=0.001
r1=0.1
r(0) =0.1
r(f) = 0.12
vtheta(0) =0
gamma(0) =-6.5455E-06
error = (0.012-vtheta)/.012*100
```

The purple arrow appears and the program is ready to run!

Ln 10 | Troubleshooting Errors Example.pol | Fluid contained between two co-axial cylinders with outer cylinder rotating

9:57 AM | 6/21/2017 | CAPS | NUM

Figure 19: Program is ready to run

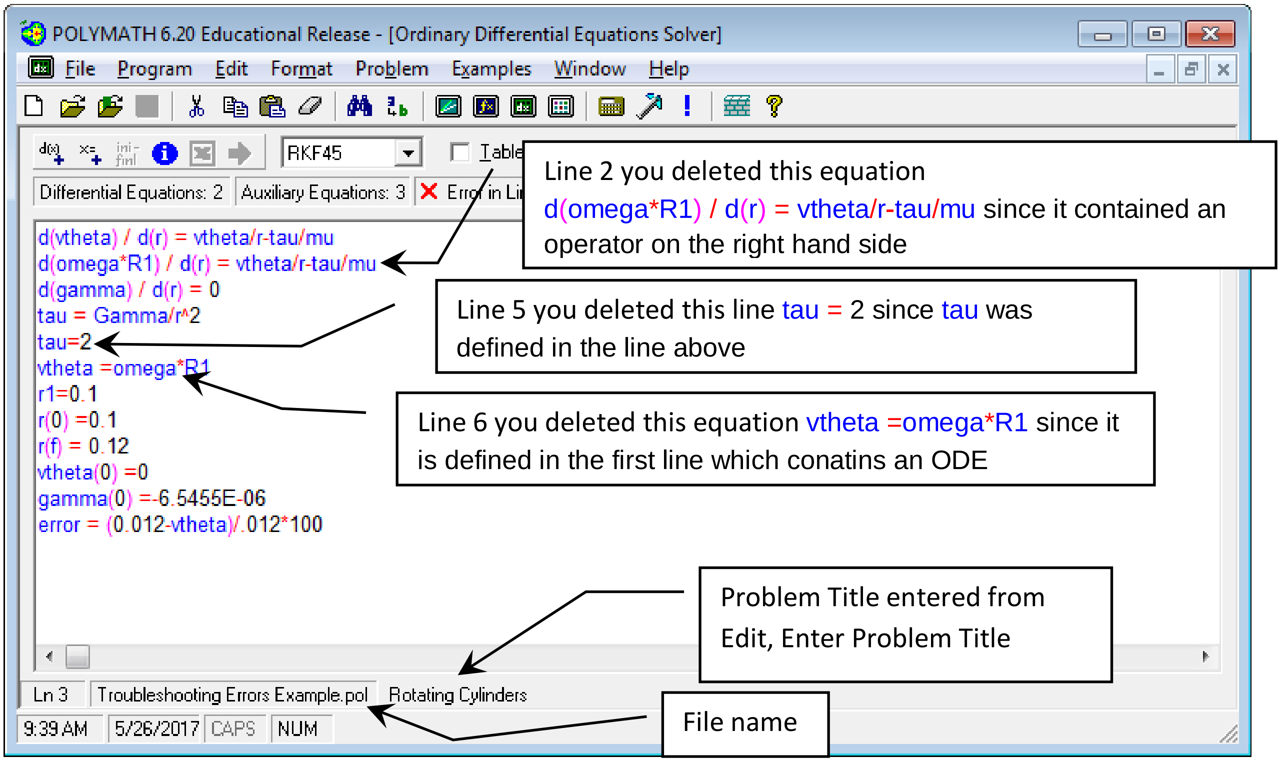
23. The following errors from the original code were fixed by following the steps above:

X Multiply (sp Multiple) defined variables : There are 2 variables (tau and vtheta) defined more than once that are involved in multiplication (This seems to be a strange error code). You already deleted the equation on line 4 (tau = 2) since you can only define it once.

X Warning: 2 Duplicate explicit variable/s exist/s. Again these are vtheta, tau.

X Warning: 4 Undefined variable/s exist. This is also giving the number of variables undefined as 4.

X Warning: 2 Multiply defined variable/s exist. Again a repeat of the above.



```
d(vtheta) / d(r) = vtheta/r-tau/mu
d(omega*R1) / d(r) = vtheta/r-tau/mu
d(gamma) / d(r) = 0
tau = Gamma/r^2
tau=2
vtheta =omega*R1
r1=0.1
r(0) =0.1
r(f) = 0.12
vtheta(0) =0
gamma(0) =-6.5455E-06
error = (0.012-vtheta)/.012*100
```

Line 2 you deleted this equation d(omega*R1) / d(r) = vtheta/r-tau/mu since it contained an operator on the right hand side

Line 5 you deleted this line tau = 2 since tau was defined in the line above

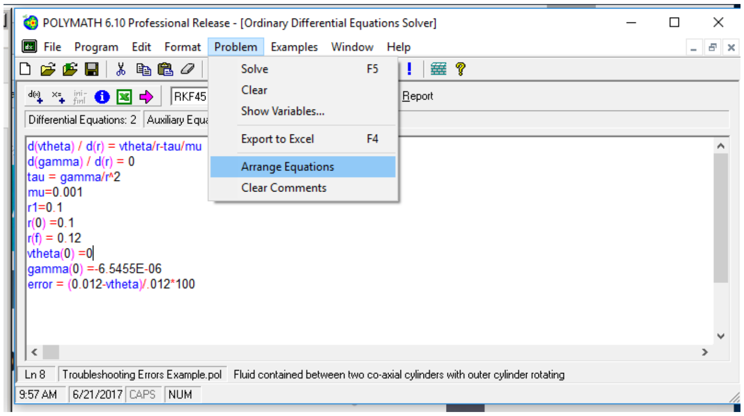
Line 6 you deleted this equation vtheta =omega*R1 since it is defined in the first line which conatins an ODE

Problem Title entered from Edit, Enter Problem Title

File name

Figure 20: Original program will errors

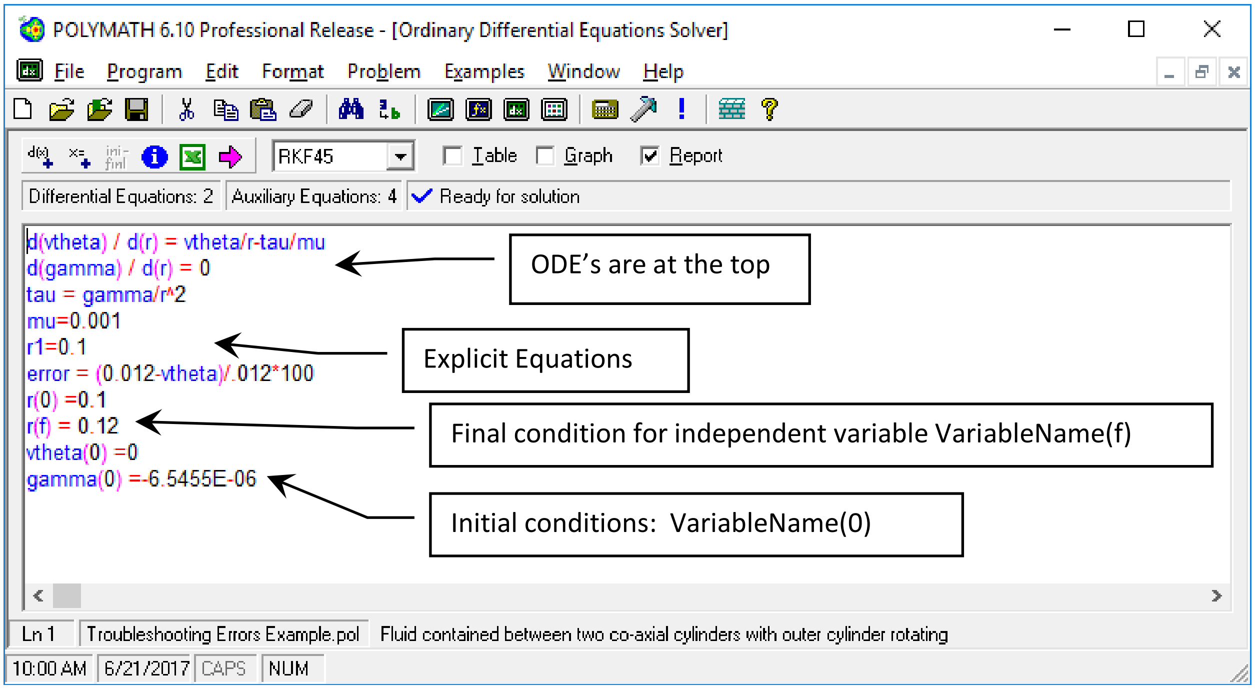
24. One last step is to have POLYMATH automatically order your equations. This is optional but very useful for the professor to grade your homework. Go to the menu and select Problem and then Arrange equations.



```
d(vtheta) / d(r) = vtheta/r-tau/mu
d(gamma) / d(r) = 0
tau = gamma/r^2
mu=0.001
r1=0.1
r(0) =0.1
r(f) = 0.12
vtheta(0) =0
gamma(0) =-6.5455E-06
error = (0.012-vtheta)/.012*100
```

Figure 21: Arrange Equations

25. The result after arranging equations will put the ODE’s at the top, followed by explicit eqautions and then initial and final conditions as shown below.



Differential Equations: 2 | Auxiliary Equations: 4 | ✓ Ready for solution

```
d(vtheta) / d(r) = vtheta/r-tau/mu
d(gamma) / d(r) = 0
tau = gamma/r^2
mu=0.001
r1=0.1
error = (0.012-vtheta)/.012*100
r(0) =0.1
r(f) = 0.12
vtheta(0) =0
gamma(0) =-6.5455E-06
```

ODE’s are at the top

Explicit Equations

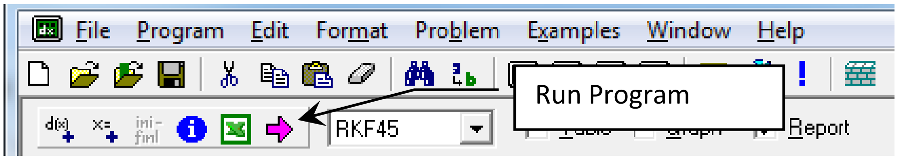
Final condition for independent variable VariableName(f)

Initial conditions: VariableName(0)

Ln 1 | Troubleshooting Errors Example.pol | Fluid contained between two co-axial cylinders with outer cylinder rotating

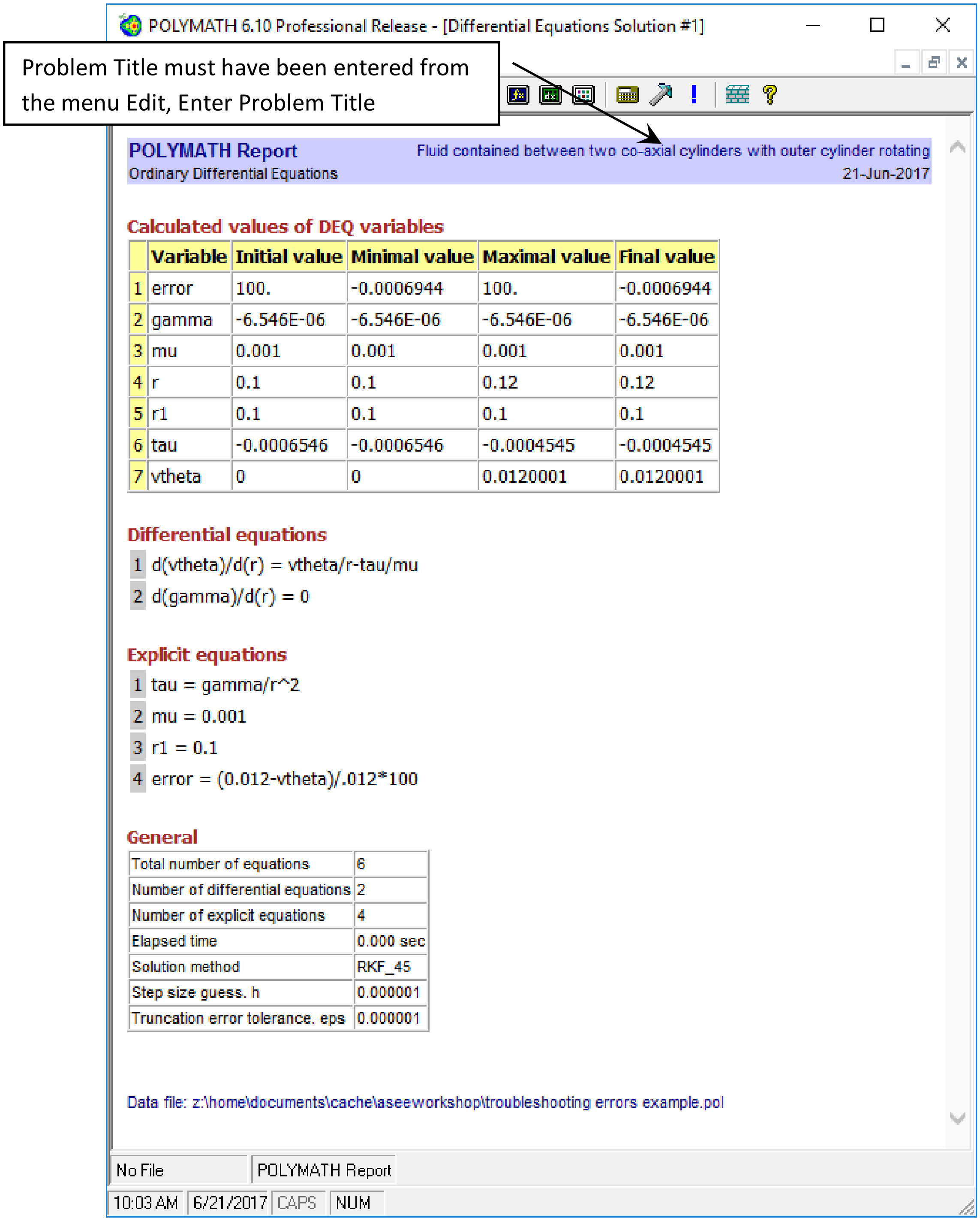
10:00 AM | 6/21/2017 | CAPS | NUM

26. Now run your corrected program by pressing the Pink arrow



Run Program

27. Your program should run and produce an output page like the following:



Problem Title must have been entered from the menu Edit, Enter Problem Title

POLYMATH Report Fluid contained between two co-axial cylinders with outer cylinder rotating
Ordinary Differential Equations 21-Jun-2017

Calculated values of DEQ variables

	Variable	Initial value	Minimal value	Maximal value	Final value
1	error	100.	-0.0006944	100.	-0.0006944
2	gamma	-6.546E-06	-6.546E-06	-6.546E-06	-6.546E-06
3	mu	0.001	0.001	0.001	0.001
4	r	0.1	0.1	0.12	0.12
5	r1	0.1	0.1	0.1	0.1
6	tau	-0.0006546	-0.0006546	-0.0004545	-0.0004545
7	vtheta	0	0	0.0120001	0.0120001

Differential equations

1 d(vtheta)/d(r) = vtheta/r-tau/mu

2 d(gamma)/d(r) = 0

Explicit equations

1 tau = gamma/r^2

2 mu = 0.001

3 r1 = 0.1

4 error = (0.012-vtheta)/.012*100

General

Total number of equations	6
Number of differential equations	2
Number of explicit equations	4
Elapsed time	0.000 sec
Solution method	RKF_45
Step size guess. h	0.000001
Truncation error tolerance. eps	0.000001

Data file: z:\home\documents\cache\aseeworkshop\troubleshooting errors example.pol

For homework assignments you will be required to copy and paste this page into a word document that will also contain answers to questions, graphs and sample calculations that will be uploaded to Blackboard.

POLYMATH Report
Ordinary Differential Equations

Fluid contained between two co-axial cylinders with outer cylinder rotating
21-Jun-2017

Calculated values of DEQ variables

	Variable	Initial value	Minimal value	Maximal value	Final value
1	error	100.	-0.0006944	100.	-0.0006944
2	gamma	-6.546E-06	-6.546E-06	-6.546E-06	-6.546E-06
3	mu	0.001	0.001	0.001	0.001
4	r	0.1	0.1	0.12	0.12
5	r1	0.1	0.1	0.1	0.1
6	tau	-0.0006546	-0.0006546	-0.0004545	-0.0004545
7	vtheta	0	0	0.0120001	0.0120001

Differential equations

1 d(vtheta)/d(r) = vtheta/r-tau/mu
2 d(gamma)/d(r) = 0

Explicit equations

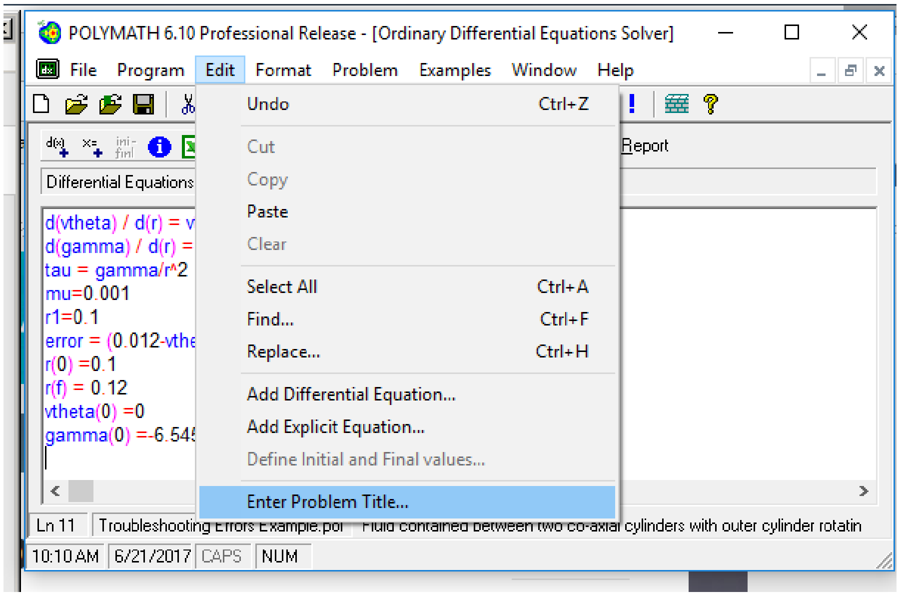
1 tau = gamma/r^2
2 mu = 0.001
3 r1 = 0.1
4 error = (0.012-vtheta)/.012*100

General

Total number of equations	6
Number of differential equations	2
Number of explicit equations	4
Elapsed time	0.000 sec
Solution method	RKF_45
Step size guess. h	0.000001
Truncation error tolerance. eps	0.000001

Data file: z:\home\documents\cache\aseeworkshop\troubleshooting errors example.pol

Remember to always enter a problem title by selecting Edit, Enter Problem Title…



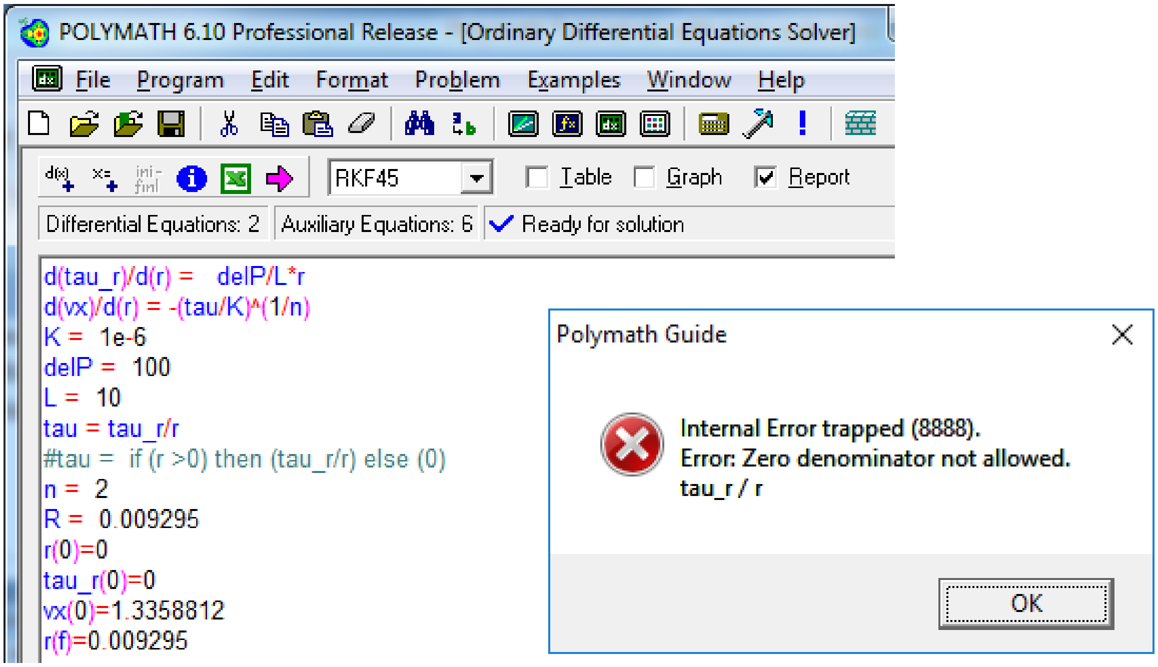
```
d(vtheta) / d(r) = v
d(gamma) / d(r) =
tau = gamma/r^2
mu=0.001
r1=0.1
error = (0.012-vthe
r(0) =0.1
r(f) = 0.12
vtheta(0) =0
gamma(0) =-6.545
```

Figure 22: Always enter a problem title so that the professor will know what problem you solved.

28. Rename and Save this program for submission on blackboard. The new filename should have your last name as part of the title.

29. Other types of errors cause a program to stop running. This error starts after you press the Pink arrow to start solving the problem below and the solution stops with an error message window titled Polymath Guide. Below is a different example to show this error. Again copy and paste the below code into an ODE solver (or load the program divideByZeroCode.pol) and then run the code

```
d(tau_r)/d(r)= delP/L*r
d(vx)/d(r)=-(tau/K)^(1/n)
K=1e-6
delP=100
L=10
tau=tau_r/r
#tau=if(r>0) then (tau_r/r) else(0)
n=2
R=0.009295
r(0)=0
tau_r(0)=0
vx(0)=1.3358812
r(f)=0.009295
```



RKF45 ☐ Table ☐ Graph ☑ Report

Differential Equations: 2 | Auxiliary Equations: 6 | ✓ Ready for solution

```
d(tau_r)/d(r) =   delP/L*r
d(vx)/d(r) = -(tau/K)^(1/n)
K =  1e-6
delP =  100
L =  10
tau = tau_r/r
#tau =  if (r >0) then (tau_r/r) else (0)
n =  2
R =  0.009295
r(0)=0
tau_r(0)=0
vx(0)=1.3358812
r(f)=0.009295
```

Polymath Guide

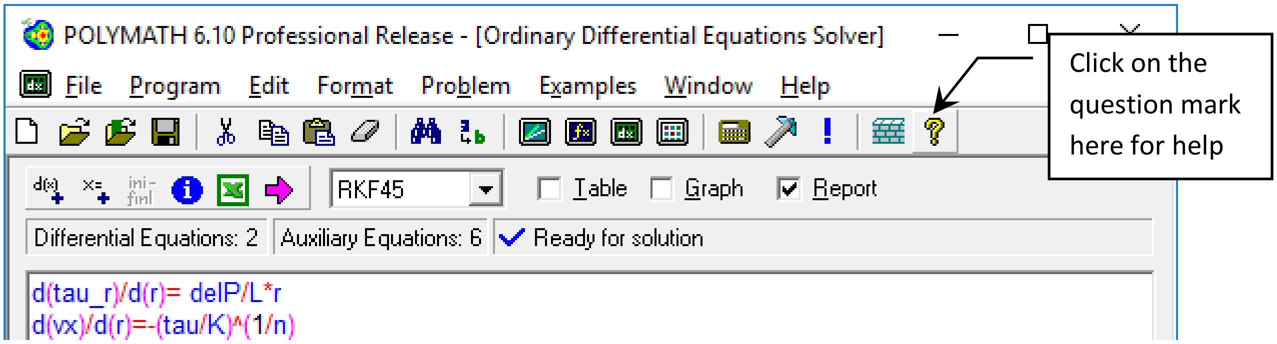
Internal Error trapped (8888).
Error: Zero denominator not allowed.
tau_r / r

Figure 23: Divide by zero example after pressing the pink arrow

In line 6, the value of r at the beginning of the integration is zero (r(0)=0) so the program tries to divide by zero in the expression in line 6 (tau = tau_r/r). This causes the program to stop and this error code appears. To correct this error, the program needs an If, Then Else statement which has been commented out using a pound sign. This is shown in line #7: #tau = if (r >0) then (tau_r/r) else (0). The pound sign # is used as a comment marker. Delete the line: tau=tau_r/r and then remove the # that is used as a comment marker. Anything after the comment marker is ignored by the solver. Now the program will run.

30. Save this program for submission on blackboard. Again the filename should have your last name in it.

Additional tips for troubleshooting are found in the help menu for ODE's which can be found by clicking on the yellow question mark or pressing the F1 key.



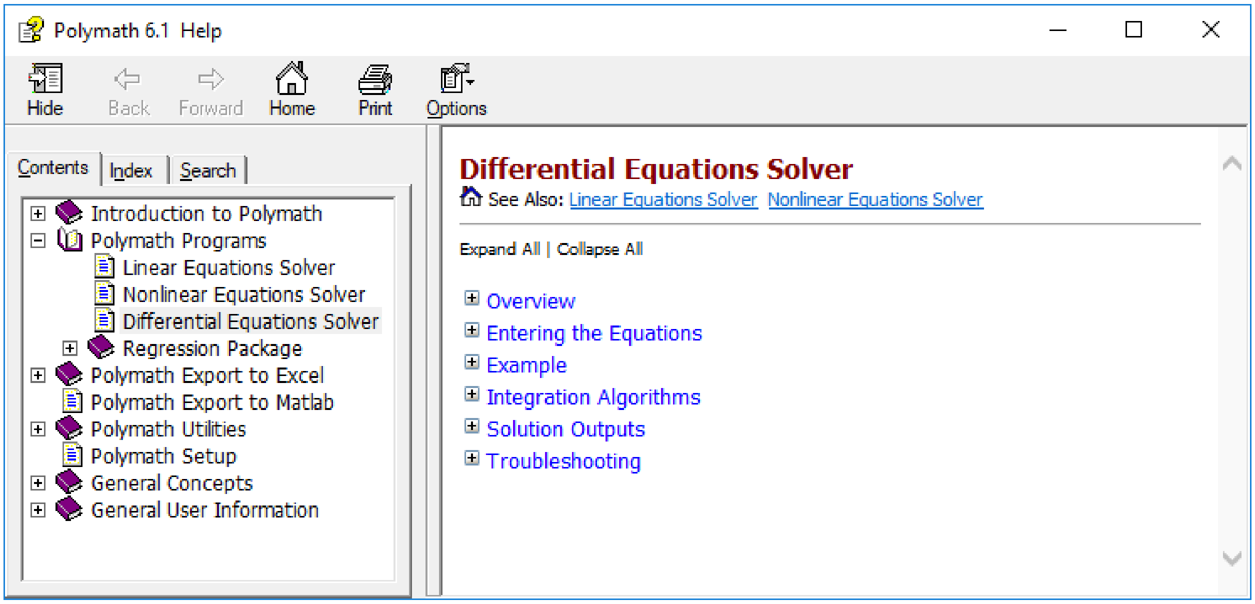
Click on the question mark here for help

RKF45 ☐ Table ☐ Graph ☑ Report

Differential Equations: 2 | Auxiliary Equations: 6 | ✓ Ready for solution

```
d(tau_r)/d(r)= delP/L*r
d(vx)/d(r)=-(tau/K)^(1/n)
```

Now scroll down or click the Collapse All to go to the section marked trouble shooting as shown below. Of particular interest will be what happens if Integration progresses very slowly and you need to use a stiff algorithm:

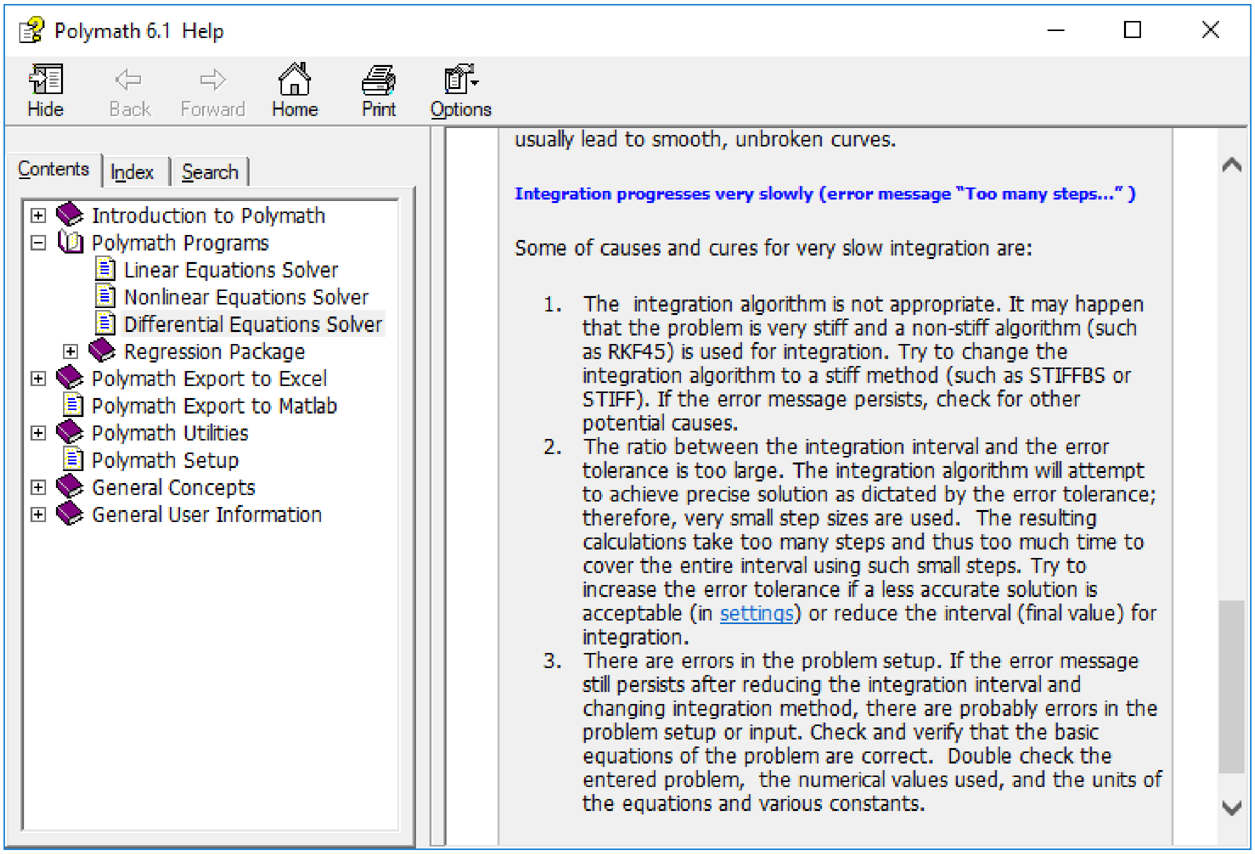


Differential Equations Solver

See Also: Linear Equations Solver Nonlinear Equations Solver

Expand All | Collapse All

⊞ Overview
⊞ Entering the Equations
⊞ Example
⊞ Integration Algorithms
⊞ Solution Outputs
⊞ Troubleshooting



usually lead to smooth, unbroken curves.

Integration progresses very slowly (error message "Too many steps...")

Some of causes and cures for very slow integration are:

1. The integration algorithm is not appropriate. It may happen that the problem is very stiff and a non-stiff algorithm (such as RKF45) is used for integration. Try to change the integration algorithm to a stiff method (such as STIFFBS or STIFF). If the error message persists, check for other potential causes.
2. The ratio between the integration interval and the error tolerance is too large. The integration algorithm will attempt to achieve precise solution as dictated by the error tolerance; therefore, very small step sizes are used. The resulting calculations take too many steps and thus too much time to cover the entire interval using such small steps. Try to increase the error tolerance if a less accurate solution is acceptable (in settings) or reduce the interval (final value) for integration.
3. There are errors in the problem setup. If the error message still persists after reducing the integration interval and changing integration method, there are probably errors in the problem setup or input. Check and verify that the basic equations of the problem are correct. Double check the entered problem, the numerical values used, and the units of the equations and various constants.

Application of Numerical Problem Solving in Chemical Engineering Coursework

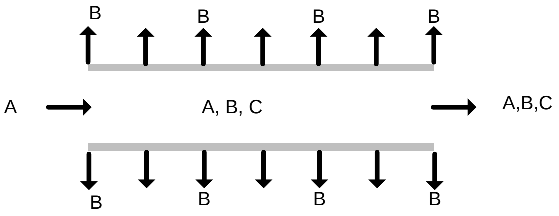
Presenters: Robert P. Hesketh, Rowan University; Michael B. Cutlip, University of Connecticut

Polymath **Software** problem-solving capabilities include

- **Differential Equations** - up to **300** simultaneous ordinary differential and 300 additional explicit algebraic equations
- **Nonlinear Equations** - up to **300** simultaneous nonlinear and 300 additional explicit algebraic equations
- **Data analysis and Regression** - up to 200 variables with up to 1000 data points for each, with capabilities for **linear, multiple linear, and nonlinear regressions** with extensive statistics plus polynomial and spline fitting with interpolation and graphing capabilities
- **Linear Equations** - up to **264** simultaneous equations

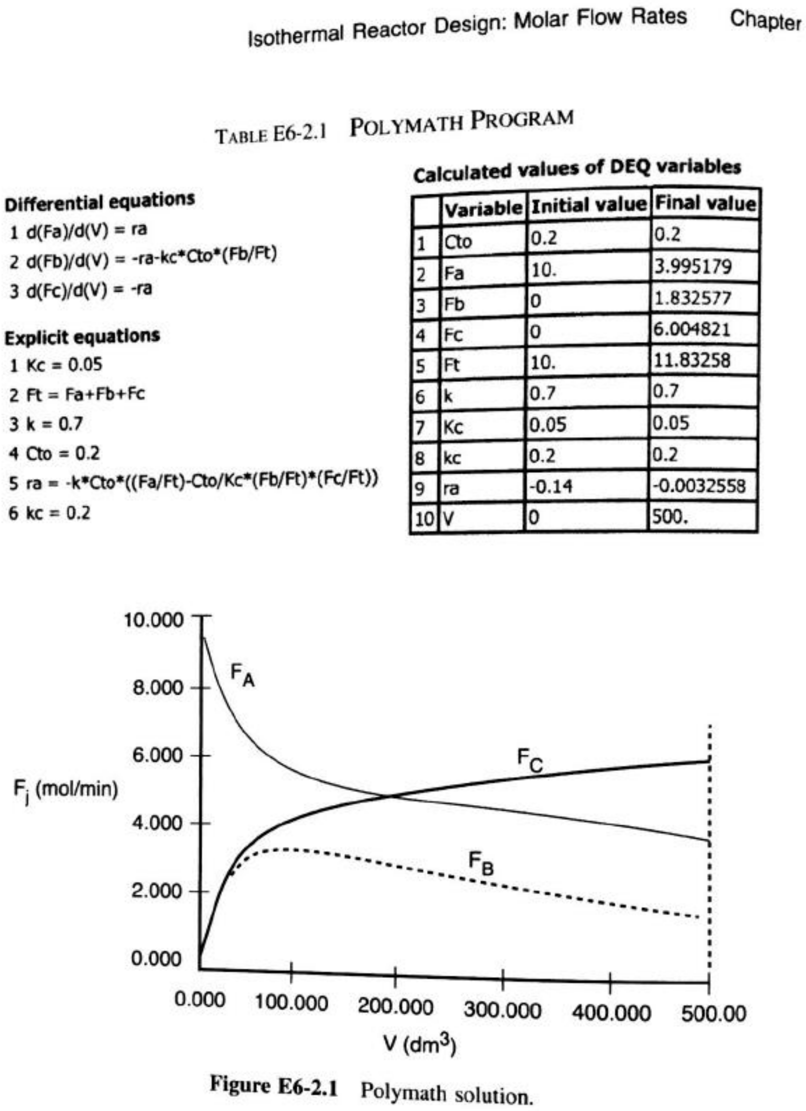
Integration of POLYMATH with Fogler’s chemical reaction engineering textbook

$$A \rightarrow B + C$$



Example 6-2 Membrane Reactor

$$\frac{dF_A}{dV} = r_A$$
$$\frac{dF_B}{dV} = -r_A - k_c C_B$$
$$\frac{dF_C}{dV} = r_C$$
$$r_A = -k\left(C_A - \frac{C_B C_C}{K_{eq}}\right)$$



Isothermal Reactor Design: Molar Flow Rates Chapter 6

TABLE E6-2.1 POLYMATH PROGRAM

Differential equations

1 d(Fa)/d(V) = ra
2 d(Fb)/d(V) = -ra-kc*Cto*(Fb/Ft)
3 d(Fc)/d(V) = -ra

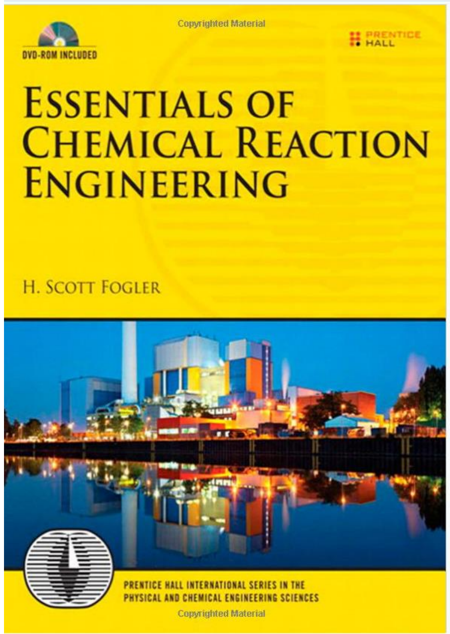
Explicit equations

1 Kc = 0.05
2 Ft = Fa+Fb+Fc
3 k = 0.7
4 Cto = 0.2
5 ra = -k*Cto*((Fa/Ft)-Cto/Kc*(Fb/Ft)*(Fc/Ft))
6 kc = 0.2

Calculated values of DEQ variables

	Variable	Initial value	Final value
1	Cto	0.2	0.2
2	Fa	10.	3.995179
3	Fb	0	1.832577
4	Fc	0	6.004821
5	Ft	10.	11.83258
6	k	0.7	0.7
7	Kc	0.05	0.05
8	kc	0.2	0.2
9	ra	-0.14	-0.0032558
10	V	0	500.

Figure E6-2.1 Polymath solution.

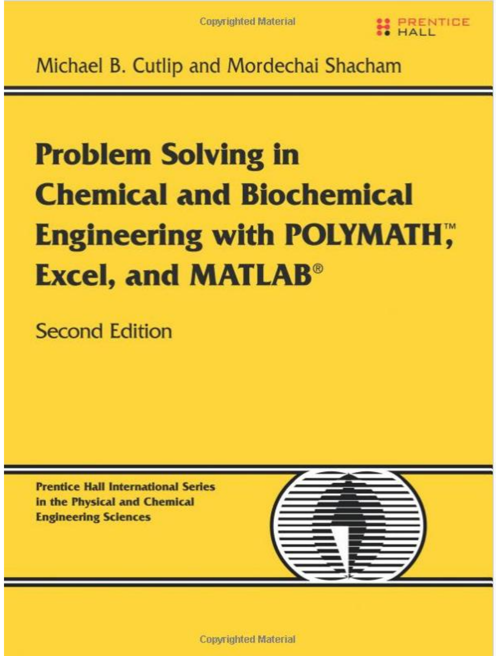


Integration already accomplished! No extra work required.

How do you integrate POLYMATH into your course?

Start with examples from POLYMATH Text

- Thermodynamics,
- Fluid Mechanics,
- Heat Transfer,
- Mass Transfer,
- Chemical Reaction Engineering,
- Phase Equilibria and Distillation,
- Process Dynamics and Control,
- Biochemical Engineering



Copyrighted Material

PRENTICE HALL

Michael B. Cutlip and Mordechai Shacham

Problem Solving in Chemical and Biochemical Engineering with POLYMATH™, Excel, and MATLAB®

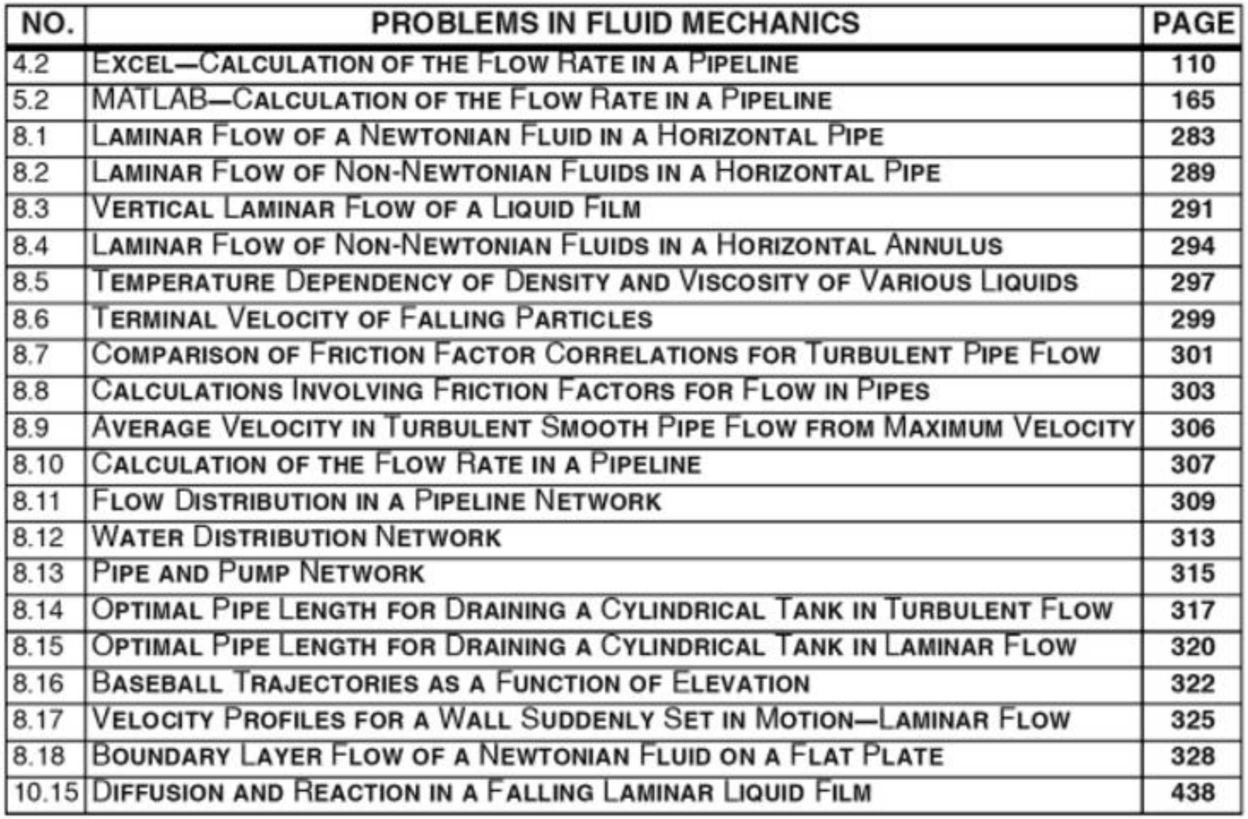
Second Edition

Prentice Hall International Series in the Physical and Chemical Engineering Sciences

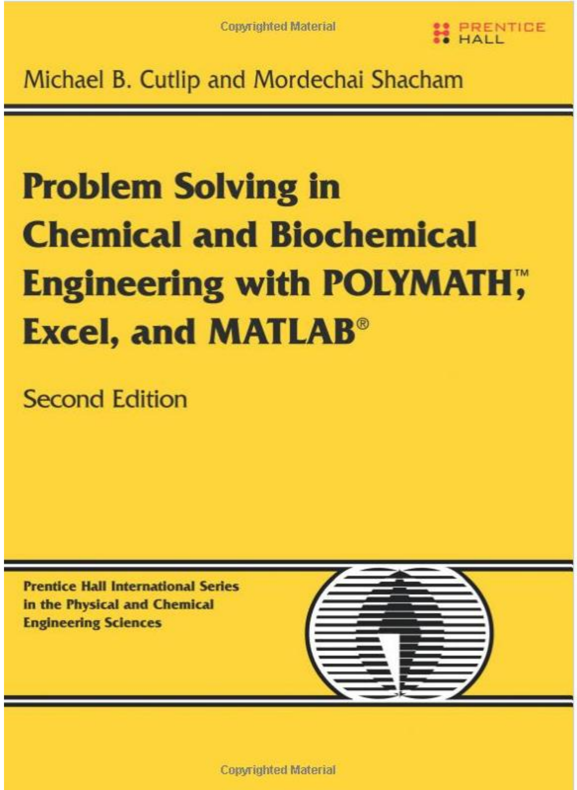
Copyrighted Material

Polymath Text: Fluids Course

Table I-3 Problems in Fluid Mechanics



NO.	PROBLEMS IN FLUID MECHANICS	PAGE
4.2	Excel—Calculation of the Flow Rate in a Pipeline	110
5.2	MATLAB—Calculation of the Flow Rate in a Pipeline	165
8.1	Laminar Flow of a Newtonian Fluid in a Horizontal Pipe	283
8.2	Laminar Flow of Non-Newtonian Fluids in a Horizontal Pipe	289
8.3	Vertical Laminar Flow of a Liquid Film	291
8.4	Laminar Flow of Non-Newtonian Fluids in a Horizontal Annulus	294
8.5	Temperature Dependency of Density and Viscosity of Various Liquids	297
8.6	Terminal Velocity of Falling Particles	299
8.7	Comparison of Friction Factor Correlations for Turbulent Pipe Flow	301
8.8	Calculations Involving Friction Factors for Flow in Pipes	303
8.9	Average Velocity in Turbulent Smooth Pipe Flow from Maximum Velocity	306
8.10	Calculation of the Flow Rate in a Pipeline	307
8.11	Flow Distribution in a Pipeline Network	309
8.12	Water Distribution Network	313
8.13	Pipe and Pump Network	315
8.14	Optimal Pipe Length for Draining a Cylindrical Tank in Turbulent Flow	317
8.15	Optimal Pipe Length for Draining a Cylindrical Tank in Laminar Flow	320
8.16	Baseball Trajectories as a Function of Elevation	322
8.17	Velocity Profiles for a Wall Suddenly Set in Motion—Laminar Flow	325
8.18	Boundary Layer Flow of a Newtonian Fluid on a Flat Plate	328
10.15	Diffusion and Reaction in a Falling Laminar Liquid Film	438



Copyrighted Material

PRENTICE HALL

Michael B. Cutlip and Mordechai Shacham

Problem Solving in Chemical and Biochemical Engineering with POLYMATH™, Excel, and MATLAB®

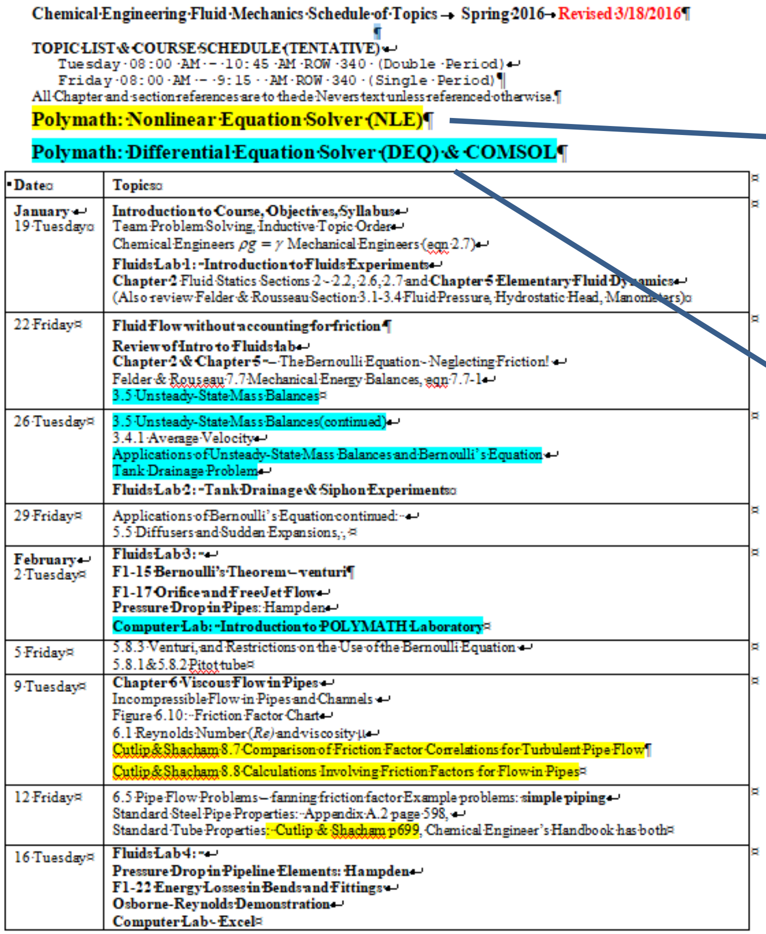
Second Edition

Prentice Hall International Series in the Physical and Chemical Engineering Sciences

Copyrighted Material

21 Problems in fluids

Example Schedule of Topics for ChE Fluids (2 credit hour)



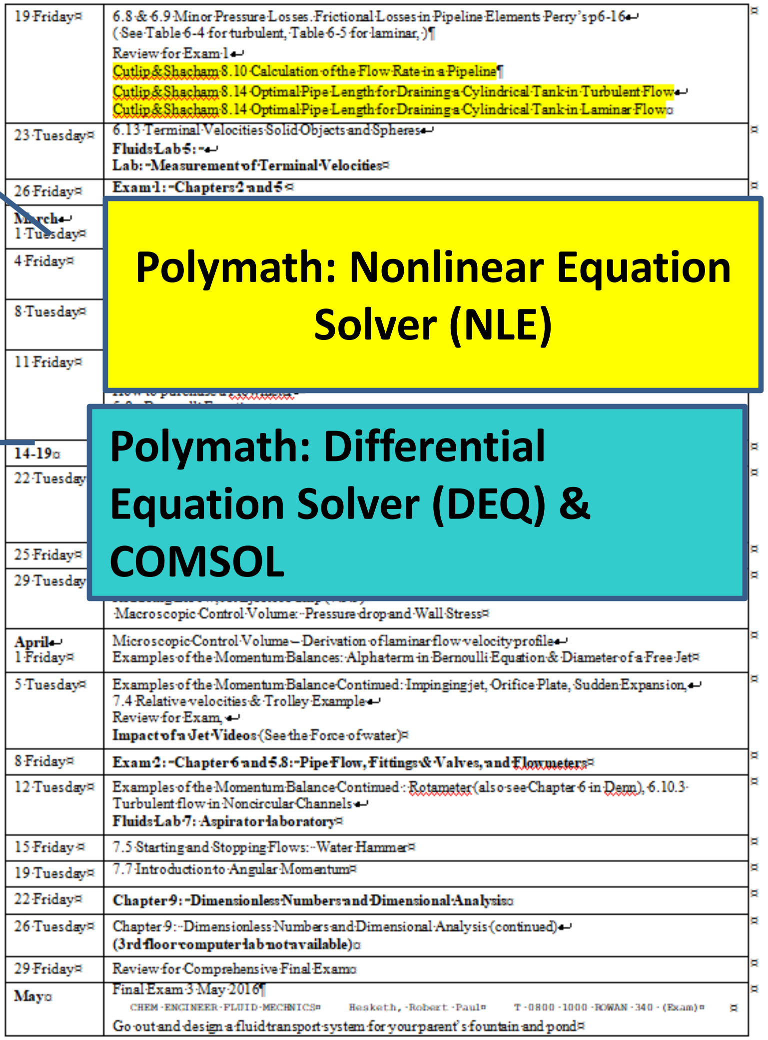
Chemical Engineering Fluid Mechanics Schedule of Topics → Spring 2016 → Revised 3/18/2016

TOPIC LIST & COURSE SCHEDULE (TENTATIVE)
Tuesday 08:00 AM – 10:45 AM ROW 340 (Double Period)
Friday 08:00 AM – 9:15 AM ROW 340 (Single Period)
All Chapter and section references are to the de Nevers text unless referenced otherwise.

Polymath: Nonlinear Equation Solver (NLE)

Polymath: Differential Equation Solver (DEQ) & COMSOL

Date	**Topics**
January 19 Tuesday	**Introduction to Course, Objectives, Syllabus** Team Problem Solving, Inductive Topic Order Chemical Engineers $\rho g = \gamma$ Mechanical Engineers (eqn 2.7) **Fluids Lab 1: Introduction to Fluids Experiments** **Chapter 2** Fluid Statics Sections 2 - 2.2, 2.6, 2.7 and **Chapter 5 Elementary Fluid Dynamics** (Also review Felder & Rousseau Section 3.1-3.4 Fluid Pressure, Hydrostatic Head, Manometers)
22 Friday	**Fluid Flow without accounting for friction** **Review of Intro to Fluids lab** **Chapter 2 & Chapter 5** – The Bernoulli Equation - Neglecting Friction! Felder & Rouseau 7.7 Mechanical Energy Balances, eqn 7.7-1 3.5 Unsteady-State Mass Balances
26 Tuesday	3.5 Unsteady-State Mass Balances (continued) 3.4.1 Average Velocity Applications of Unsteady-State Mass Balances and Bernoulli's Equation Tank Drainage Problem **Fluids Lab 2: Tank Drainage & Siphon Experiments**
29 Friday	Applications of Bernoulli's Equation continued: 5.5 Diffusers and Sudden Expansions, ;
February 2 Tuesday	**Fluids Lab 3:** **F1-15 Bernoulli's Theorem – venturi** **F1-17 Orifice and Free Jet Flow** **Pressure Drop in Pipes**: Hampden **Computer Lab: Introduction to POLYMATH Laboratory**
5 Friday	5.8.3 Venturi, and Restrictions on the Use of the Bernoulli Equation 5.8.1 & 5.8.2 Pitot tube
9 Tuesday	**Chapter 6 Viscous Flow in Pipes** Incompressible Flow in Pipes and Channels Figure 6.10: Friction Factor Chart 6.1 Reynolds Number (*Re*) and viscosity µ Cutlip & Shacham 8.7 Comparison of Friction Factor Correlations for Turbulent Pipe Flow Cutlip & Shacham 8.8 Calculations Involving Friction Factors for Flow in Pipes
12 Friday	6.5 Pipe Flow Problems – fanning friction factor Example problems: **simple piping** Standard Steel Pipe Properties: Appendix A.2 page 598, Standard Tube Properties: Cutlip & Shacham p699, Chemical Engineer's Handbook has both
16 Tuesday	**Fluids Lab 4:** **Pressure Drop in Pipeline Elements: Hampden** **F1-22 Energy Losses in Bends and Fittings** **Osborne-Reynolds Demonstration** **Computer Lab - Excel**

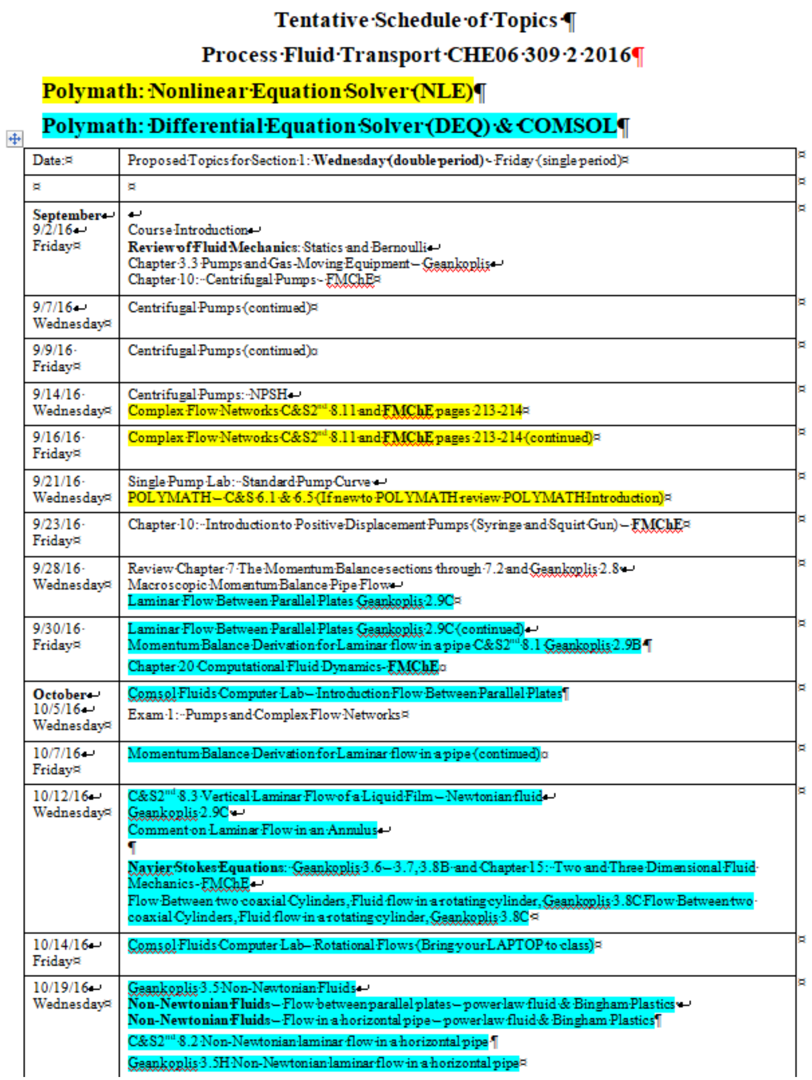


19 Friday	6.8 & 6.9 Minor Pressure Losses. Frictional Losses in Pipeline Elements Perry's p6-16 (See Table 6-4 for turbulent, Table 6-5 for laminar,) Review for Exam 1 Cutlip & Shacham 8.10 Calculation of the Flow Rate in a Pipeline Cutlip & Shacham 8.14 Optimal Pipe Length for Draining a Cylindrical Tank in Turbulent Flow Cutlip & Shacham 8.14 Optimal Pipe Length for Draining a Cylindrical Tank in Laminar Flow
23 Tuesday	6.13 Terminal Velocities Solid Objects and Spheres **Fluids Lab 5:** **Lab: Measurement of Terminal Velocities**
26 Friday	**Exam 1: Chapters 2 and 5**
March 1 Tuesday	
4 Friday	
8 Tuesday	
11 Friday	
14-19	
22 Tuesday	
25 Friday	
29 Tuesday	Macroscopic Control Volume: Pressure drop and Wall Stress
April 1 Friday	Microscopic Control Volume – Derivation of laminar flow velocity profile Examples of the Momentum Balances: Alpha term in Bernoulli Equation & Diameter of a Free Jet
5 Tuesday	Examples of the Momentum Balance Continued: Impinging jet, Orifice Plate, Sudden Expansion, 7.4 Relative velocities & Trolley Example Review for Exam, **Impact of a Jet Videos** (See the Force of water)
8 Friday	**Exam 2: Chapter 6 and 5.8: Pipe Flow, Fittings & Valves, and Flowmeters**
12 Tuesday	Examples of the Momentum Balance Continued : Rotameter (also see Chapter 6 in Denn), 6.10.3 Turbulent flow in Noncircular Channels **Fluids Lab 7: Aspirator laboratory**
15 Friday	7.5 Starting and Stopping Flows: Water Hammer
19 Tuesday	7.7 Introduction to Angular Momentum
22 Friday	**Chapter 9: Dimensionless Numbers and Dimensional Analysis**
26 Tuesday	Chapter 9: Dimensionless Numbers and Dimensional Analysis (continued) **(3rd floor computer lab not available)**
29 Friday	Review for Comprehensive Final Exam
May	Final Exam 3 May 2016 CHEM ENGINEER FLUID MECHNICS Hesketh, Robert Paul T 0800 1000 ROWAN 340 (Exam) Go out and design a fluid transport system for your parent's fountain and pond

Polymath: Nonlinear Equation Solver (NLE)

Polymath: Differential Equation Solver (DEQ) & COMSOL

Adv. ChE Fluids (2 cr)



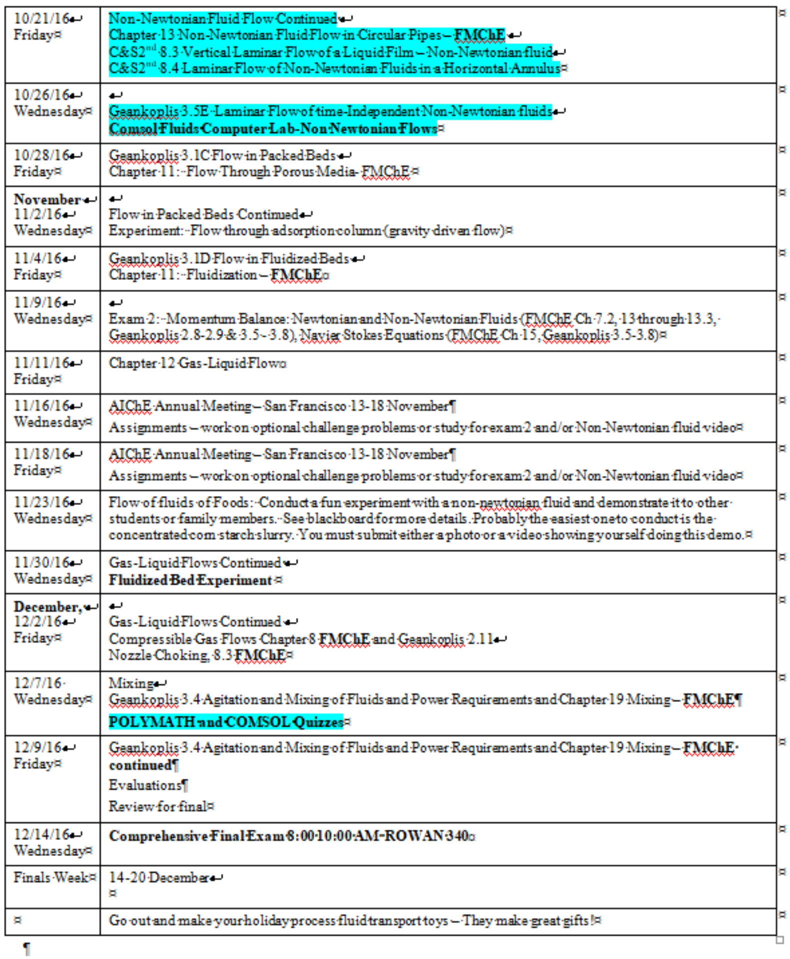
Tentative Schedule of Topics

Process Fluid Transport CHE06 309 2 2016

Polymath: Nonlinear Equation Solver (NLE)

Polymath: Differential Equation Solver (DEQ) & COMSOL

Date:	Proposed Topics for Section 1: **Wednesday (double period)** - Friday (single period)
September 9/2/16 Friday	Course Introduction **Review of Fluid Mechanics**: Statics and Bernoulli Chapter 3.3 Pumps and Gas-Moving Equipment – Geankoplis Chapter 10: Centrifugal Pumps - FMChE
9/7/16 Wednesday	Centrifugal Pumps (continued)
9/9/16 Friday	Centrifugal Pumps (continued)
9/14/16 Wednesday	Centrifugal Pumps: NPSH Complex Flow Networks C&S2nd 8.11 and **FMChE** pages 213-214
9/16/16 Friday	Complex Flow Networks C&S2nd 8.11 and **FMChE** pages 213-214 (continued)
9/21/16 Wednesday	Single Pump Lab: Standard Pump Curve POLYMATH – C&S 6.1 & 6.5 (If new to POLYMATH review POLYMATH Introduction)
9/23/16 Friday	Chapter 10: Introduction to Positive Displacement Pumps (Syringe and Squirt Gun) – **FMChE**
9/28/16 Wednesday	Review Chapter 7 The Momentum Balance sections through 7.2 and Geankoplis 2.8 Macroscopic Momentum Balance Pipe Flow Laminar Flow Between Parallel Plates Geankoplis 2.9C
9/30/16 Friday	Laminar Flow Between Parallel Plates Geankoplis 2.9C (continued) Momentum Balance Derivation for Laminar flow in a pipe C&S2nd 8.1 Geankoplis 2.9B Chapter 20 Computational Fluid Dynamics-**FMChE**
October 10/5/16 Wednesday	Comsol Fluids Computer Lab – Introduction Flow Between Parallel Plates Exam 1: Pumps and Complex Flow Networks
10/7/16 Friday	Momentum Balance Derivation for Laminar flow in a pipe (continued)
10/12/16 Wednesday	C&S2nd 8.3 Vertical Laminar Flow of a Liquid Film – Newtonian fluid Geankoplis 2.9C Comment on Laminar Flow in an Annulus **Navier Stokes Equations**: Geankoplis 3.6 – 3.7, 3.8B and Chapter 15: Two and Three Dimensional Fluid Mechanics - FMChE Flow Between two coaxial Cylinders, Fluid flow in a rotating cylinder, Geankoplis 3.8C Flow Between two coaxial Cylinders, Fluid flow in a rotating cylinder, Geankoplis 3.8C
10/14/16 Friday	Comsol Fluids Computer Lab – Rotational Flows (Bring your LAPTOP to class)
10/19/16 Wednesday	Geankoplis 3.5 Non-Newtonian Fluids **Non-Newtonian Fluids** – Flow between parallel plates – power law fluid & Bingham Plastics **Non-Newtonian Fluids** – Flow in a horizontal pipe – power law fluid & Bingham Plastics C&S2nd 8.2 Non-Newtonian laminar flow in a horizontal pipe Geankoplis 3.5H Non-Newtonian laminar flow in a horizontal pipe



10/21/16 Friday	Non-Newtonian Fluid Flow Continued Chapter 13 Non-Newtonian Fluid Flow in Circular Pipes – **FMChE** C&S2nd 8.3 Vertical Laminar Flow of a Liquid Film – Non-Newtonian fluid C&S2nd 8.4 Laminar Flow of Non-Newtonian Fluids in a Horizontal Annulus
10/26/16 Wednesday	Geankoplis 3.5E Laminar Flow of time-Independent Non-Newtonian fluids **Comsol Fluids Computer Lab-Non Newtonian Flows**
10/28/16 Friday	Geankoplis 3.1C Flow in Packed Beds Chapter 11: Flow Through Porous Media- FMChE
November 11/2/16 Wednesday	Flow in Packed Beds Continued Experiment: Flow through adsorption column (gravity driven flow)
11/4/16 Friday	Geankoplis 3.1D Flow in Fluidized Beds Chapter 11: Fluidization – **FMChE**
11/9/16 Wednesday	Exam 2: Momentum Balance: Newtonian and Non-Newtonian Fluids (FMChE Ch 7.2, 13 through 13.3, Geankoplis 2.8-2.9 & 3.5 - 3.8), Navier Stokes Equations (FMChE Ch 15, Geankoplis 3.5-3.8)
11/11/16 Friday	Chapter 12 Gas-Liquid Flow
11/16/16 Wednesday	AIChE Annual Meeting – San Francisco 13-18 November Assignments – work on optional challenge problems or study for exam 2 and/or Non-Newtonian fluid video
11/18/16 Friday	AIChE Annual Meeting – San Francisco 13-18 November Assignments – work on optional challenge problems or study for exam 2 and/or Non-Newtonian fluid video
11/23/16 Wednesday	Flow of fluids of Foods: Conduct a fun experiment with a non-newtonian fluid and demonstrate it to other students or family members. See blackboard for more details. Probably the easiest one to conduct is the concentrated corn starch slurry. You must submit either a photo or a video showing yourself doing this demo.
11/30/16 Wednesday	Gas-Liquid Flows Continued **Fluidized Bed Experiment**
December, 12/2/16 Friday	Gas-Liquid Flows Continued Compressible Gas Flows Chapter 8 **FMChE** and Geankoplis 2.11 Nozzle Choking, 8.3 **FMChE**
12/7/16 Wednesday	Mixing Geankoplis 3.4 Agitation and Mixing of Fluids and Power Requirements and Chapter 19 Mixing – **FMChE** **POLYMATH and COMSOL Quizzes**
12/9/16 Friday	Geankoplis 3.4 Agitation and Mixing of Fluids and Power Requirements and Chapter 19 Mixing – **FMChE continued** Evaluations Review for final
12/14/16 Wednesday	**Comprehensive Final Exam 8:00-10:00 AM - ROWAN 340**
Finals Week	14-20 December
	Go out and make your holiday process fluid transport toys – They make great gifts!

Typical Fluids Problems

ChE Course	**Problem Name**	**Numerical Method Illustrated**	**Equations**
Fluids	Calculations involving Friction Factors for Flow in Pipes (POLYMATH Text 8.7) and *pipeflow homework* *frictionfactorcalcsoln.pdf* *Excel Tutorial Solver Add-Ins rev4.pdf*	Solution of a system of simultaneous nonlinear algebraic equations (NLE)	$\frac{\Delta P}{\Delta L} = 2f_F\frac{\rho v^2}{D}$ $f_F = f(\varepsilon/D\,, Re)$ $Re = \rho v D/\mu$
	Unsteady-state tank drainage using a siphon tube (similar to POLYMATH text 8.14) *C&S8-14soln.pdf*	Solution of an first order ordinary differential equation (DEQ)	$\frac{dh_T}{dt} = v_{out}\frac{A_{out}}{A_{tank}}$ $v_{out} = f(h_T)$
Advanced Fluids	NonNewtonian fluid flow through a pipe (POLYMATH Text 8.2c) *NonNewtonian C&S 8.2 solutions & comsol.pdf* NonNewtonian fluid flow through an annulus (POLYMATH Text 8.4) *NonNewtonian C&S8.4 polymath&comsol & 3.8-8 solutions 2017.pdf*	Solution of 2 simultaneous first order ordinary differential equations with split boundary value conditions and comparison with solution using COMSOL which is an advanced finite element program	$\frac{d(r\tau_{rx})}{dr} = -\frac{dP}{dx}r$ $\tau_{rx} = -K\left(\frac{dv_x}{dr}\right)\left(\left\|\frac{dv_x}{dr}\right\|\right)^{(n-1)}$

POLYMATH in a classroom

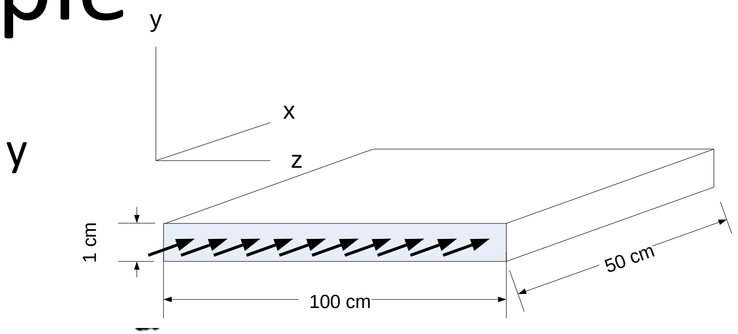
- Students need to understand the models that they use.
- Have students derive the model equations
- Then enter them into POLYMATH

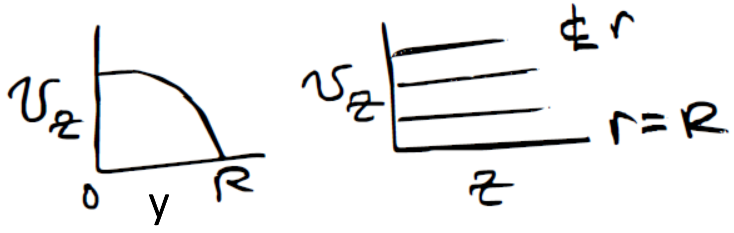
POLYMATH is not a “canned” program in which the equations are hidden such as in COMSOL and ASPEN

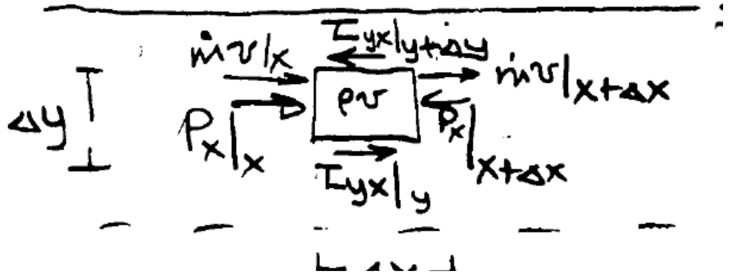
The next slides give an example of using POLYMATH with a problem in Fluids

Newtonian Fluid Flow Between Parallel Plates Example

- Figures showing flow
- Graphs with expected behavior
- Control Volume – shell balance
- Derivation
- Simplifications: steady-state etc.







$$\cancelto{0}{\frac{\partial(\rho v)}{\partial t}} = -\cancelto{0}{\frac{\partial(\rho v v)}{\partial x}} - \frac{\partial P_x}{\partial x} - \frac{\partial(\tau_{yx})}{\partial y}$$

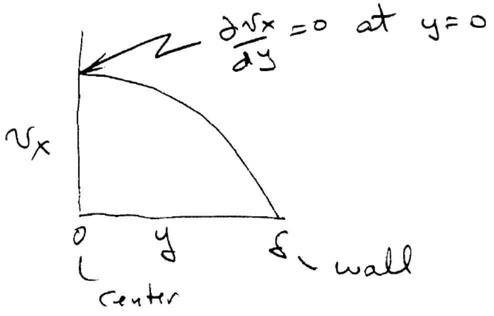
Analytical Solution

- Newtonian Fluid

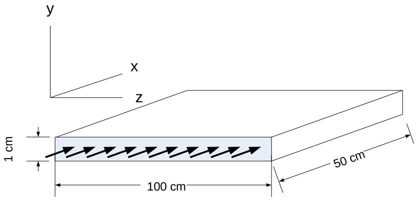
$$\tau_{yx} = -\mu \frac{\partial v_x}{\partial y}$$

$$\frac{\partial(\tau_{yx})}{\partial y} = -\underbrace{\frac{\partial P_x}{\partial x}}_{\text{this is a constant}}$$

- Boundary Conditions
  - $y = 0 \quad v_x = max \quad \tau_{yx} = 0$
  - $y = wall \; v_x = 0 \quad \tau_{yx} = \max$
- Integrate Twice: Analytical Solution
  - $v_x = -\frac{dP}{dx}\left(\frac{\delta^2}{2\mu}\right)\left[1 - \left(\frac{y}{\delta}\right)^2\right]$



Numerical Solution



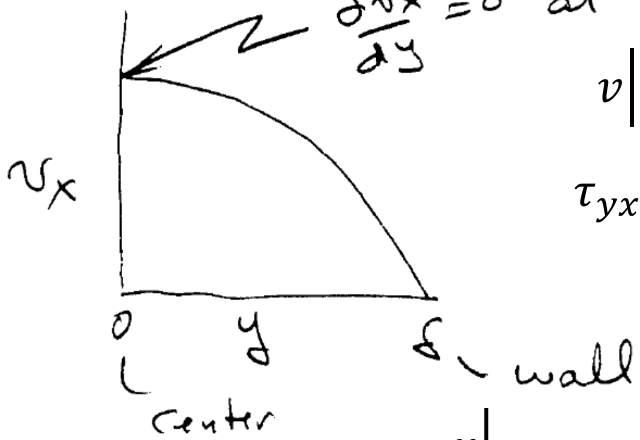
- Newtonian Fluid

$$\tau_{yx} = -\mu \frac{\partial v_x}{\partial y}$$

$$\frac{\partial (\tau_{yx})}{\partial y} = \underbrace{-\frac{\partial P_x}{\partial x}}_{\text{this is a constant}}$$

- Two coupled ODE's Split Boundary Condition
- Then manipulate two ODE's so they can be solved using the POLYMATH Differential Equation Solver (DEQ)

$\frac{\partial v_x}{\partial y} = 0$ at $y = 0$



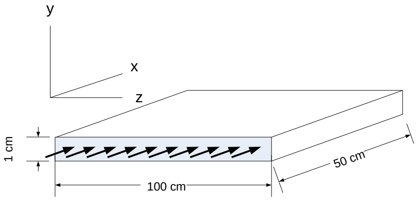
$$v\Big|_{y=0} = big$$

$$\tau_{yx}\Big|_{y=0} = 0$$

$$v\Big|_{y=wall} = 0$$

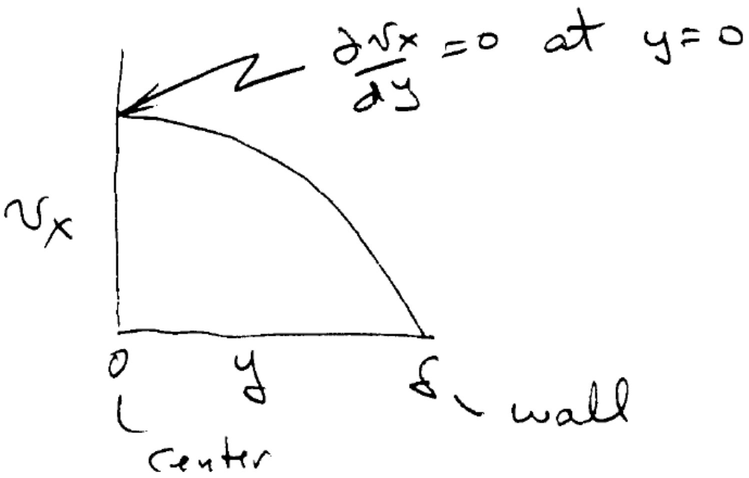
$$\tau_{yx}\Big|_{y=wall} = big$$

Required manipulation to solve 2 ODE's with split Boundary conditions



$$\tau_{yx} = -\mu\frac{\partial v_x}{\partial y} \Longrightarrow \left(\frac{\partial v_x}{\partial y}\right) = \frac{\tau_{yx}}{-\mu}$$

$$\frac{\partial(\tau_{yx})}{\partial y} = -\underbrace{\frac{\partial P_x}{\partial x}}_{\text{this is a constant}} \Longrightarrow \left(\frac{\partial \tau_{yx}}{\partial y}\right) = -\frac{dP}{dx}$$

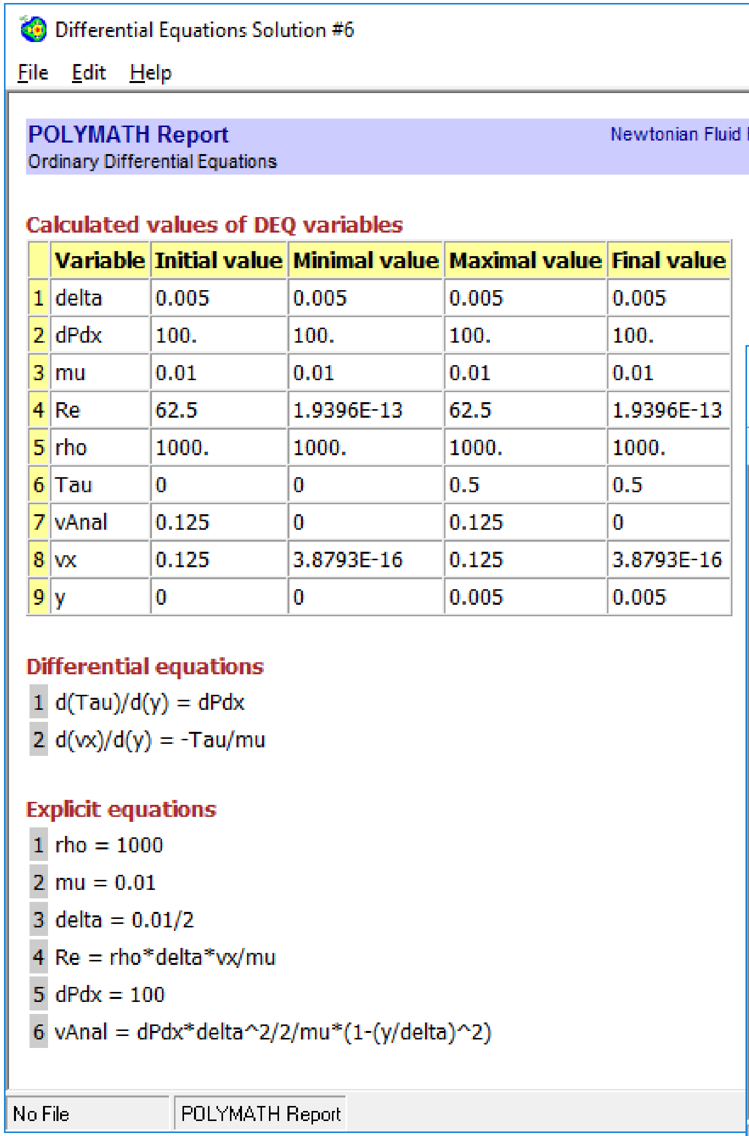


$y = 0 \quad v_x = \max \quad \tau_{yx} = 0$

$y = wall \; v_x = 0 \quad \tau_{yx} = \max$

Integration starts at y=0 and both initial conditions must be known! Solution is to guess v_x at $y = 0$ until at y =wall v_x=0

Solution



POLYMATH Report Newtonian Fluid Flow Between 2 parallel plates
Ordinary Differential Equations 30-Jun-2017

Calculated values of DEQ variables

	Variable	Initial value	Minimal value	Maximal value	Final value
1	delta	0.005	0.005	0.005	0.005
2	dPdx	100.	100.	100.	100.
3	mu	0.01	0.01	0.01	0.01
4	Re	62.5	1.9396E-13	62.5	1.9396E-13
5	rho	1000.	1000.	1000.	1000.
6	Tau	0	0	0.5	0.5
7	vAnal	0.125	0	0.125	0
8	vx	0.125	3.8793E-16	0.125	3.8793E-16
9	y	0	0	0.005	0.005

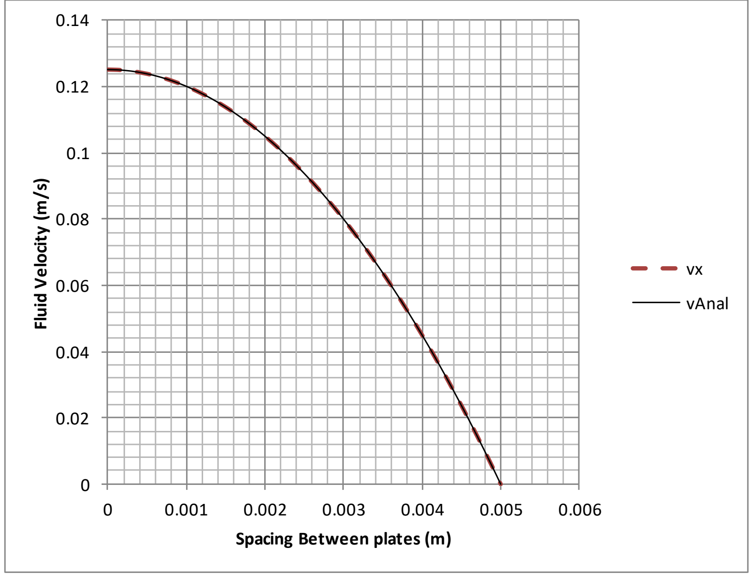
Differential equations

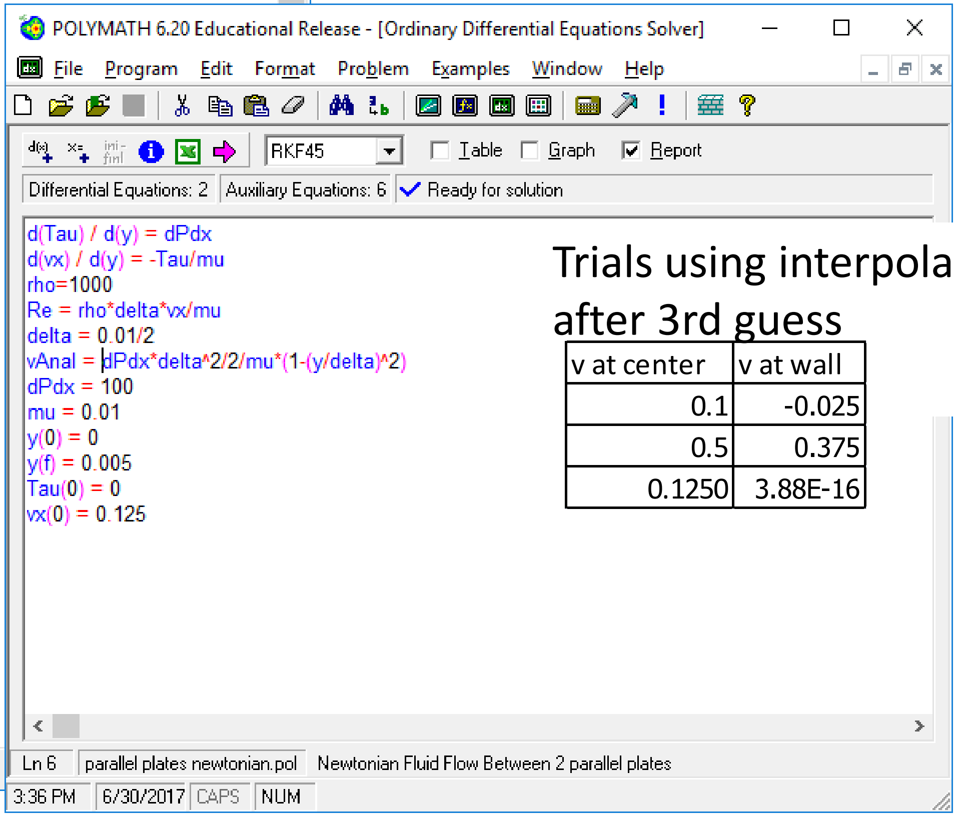
1 d(Tau)/d(y) = dPdx
2 d(vx)/d(y) = -Tau/mu

Explicit equations

1 rho = 1000
2 mu = 0.01
3 delta = 0.01/2
4 Re = rho*delta*vx/mu
5 dPdx = 100
6 vAnal = dPdx*delta^2/2/mu*(1-(y/delta)^2)

No File | POLYMATH Report





Differential Equations: 2 | Auxiliary Equations: 6 | ✓ Ready for solution

```
d(Tau) / d(y) = dPdx
d(vx) / d(y) = -Tau/mu
rho=1000
Re = rho*delta*vx/mu
delta = 0.01/2
vAnal = dPdx*delta^2/2/mu*(1-(y/delta)^2)
dPdx = 100
mu = 0.01
y(0) = 0
y(f) = 0.005
Tau(0) = 0
vx(0) = 0.125
```

Ln 6 | parallel plates newtonian.pol | Newtonian Fluid Flow Between 2 parallel plates
3:36 PM | 6/30/2017 | CAPS | NUM

Trials using interpolation after 3rd guess

v at center	v at wall
0.1	-0.025
0.5	0.375
0.1250	3.88E-16

Students are Confused

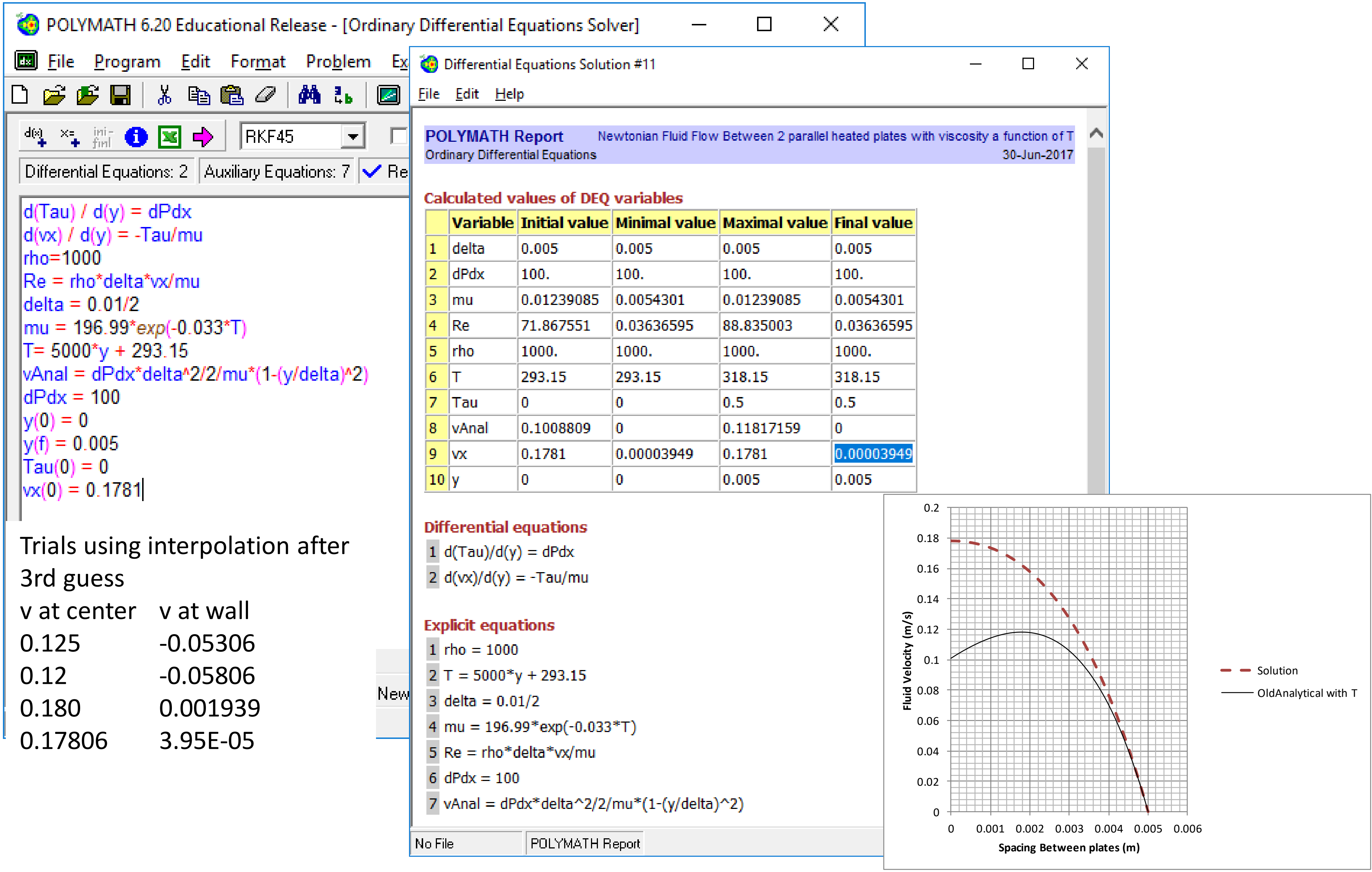
- **Question**
  Why do I have to do trial & error for the initial velocity? Why not just plug-in the maximum velocity from the analytical solution?
- **Answer**
  Your goal is always to compare a numerical solution to a simple analytical problem solution. This shows that the numerical solution method is correct.
- Then give students a more complex problem with one of the plates heated resulting in a temperature profile in the liquid. Now they must do the trial and error procedure.

Temperature Profile in liquid:

$$T = 5000\frac{K}{m}y + 293.15K$$

$$\mu = \frac{196.99\text{kg}}{m\ s}\exp\left(-\frac{0.033}{K}\mathrm{T}\right)$$

Heated Plates



```
d(Tau) / d(y) = dPdx
d(vx) / d(y) = -Tau/mu
rho=1000
Re = rho*delta*vx/mu
delta = 0.01/2
mu = 196.99*exp(-0.033*T)
T= 5000*y + 293.15
vAnal = dPdx*delta^2/2/mu*(1-(y/delta)^2)
dPdx = 100
y(0) = 0
y(f) = 0.005
Tau(0) = 0
vx(0) = 0.1781
```

POLYMATH Report Newtonian Fluid Flow Between 2 parallel heated plates with viscosity a function of T

Ordinary Differential Equations 30-Jun-2017

Calculated values of DEQ variables

	Variable	Initial value	Minimal value	Maximal value	Final value
1	delta	0.005	0.005	0.005	0.005
2	dPdx	100.	100.	100.	100.
3	mu	0.01239085	0.0054301	0.01239085	0.0054301
4	Re	71.867551	0.03636595	88.835003	0.03636595
5	rho	1000.	1000.	1000.	1000.
6	T	293.15	293.15	318.15	318.15
7	Tau	0	0	0.5	0.5
8	vAnal	0.1008809	0	0.11817159	0
9	vx	0.1781	0.00003949	0.1781	0.00003949
10	y	0	0	0.005	0.005

Differential equations

1 d(Tau)/d(y) = dPdx
2 d(vx)/d(y) = -Tau/mu

Explicit equations

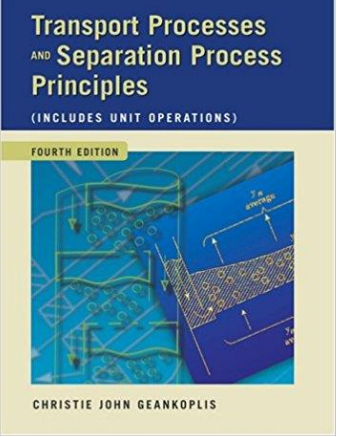
1 rho = 1000
2 T = 5000*y + 293.15
3 delta = 0.01/2
4 mu = 196.99*exp(-0.033*T)
5 Re = rho*delta*vx/mu
6 dPdx = 100
7 vAnal = dPdx*delta^2/2/mu*(1-(y/delta)^2)

Trials using interpolation after 3rd guess

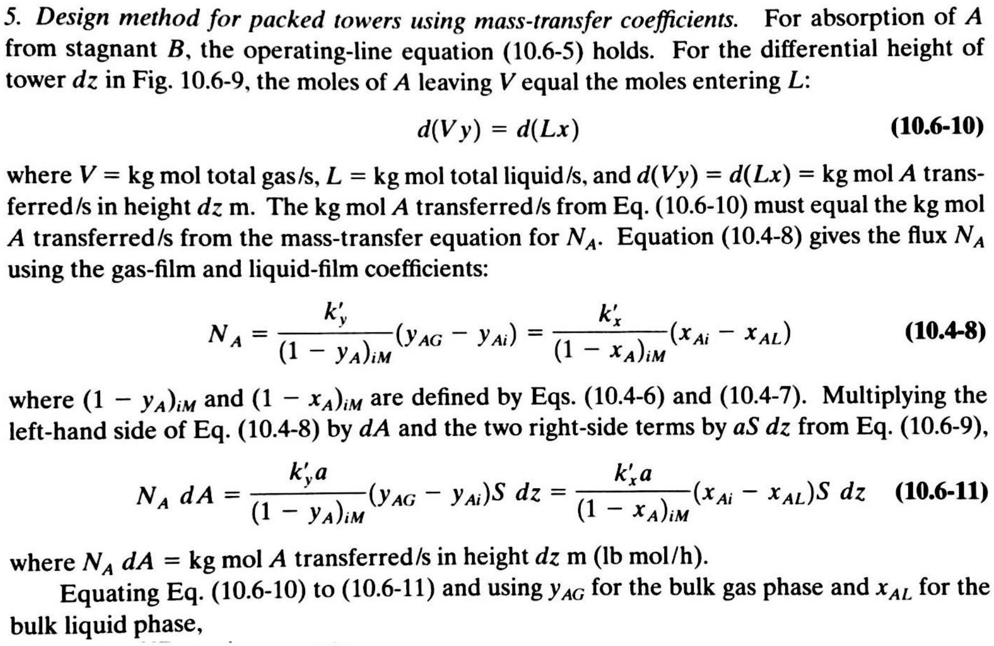
v at center	v at wall
0.125	-0.05306
0.12	-0.05806
0.180	0.001939
0.17806	3.95E-05

What about models that are formulated as integrals?

- Previous *state of the art* numerical methods where based on evaluating integrals
  - Trapezoidal rule
  - Simpson’s Rule
- Many textbooks present models only as integrals



Packed Towers: Gas Absorption Traditional Approach using integrals



5. *Design method for packed towers using mass-transfer coefficients.* For absorption of A from stagnant B, the operating-line equation (10.6-5) holds. For the differential height of tower dz in Fig. 10.6-9, the moles of A leaving V equal the moles entering L:

$$d(Vy) = d(Lx) \qquad \textbf{(10.6-10)}$$

where V = kg mol total gas/s, L = kg mol total liquid/s, and $d(Vy) = d(Lx)$ = kg mol A transferred/s in height dz m. The kg mol A transferred/s from Eq. (10.6-10) must equal the kg mol A transferred/s from the mass-transfer equation for N_A. Equation (10.4-8) gives the flux N_A using the gas-film and liquid-film coefficients:

$$N_A = \frac{k'_y}{(1 - y_A)_{iM}}(y_{AG} - y_{Ai}) = \frac{k'_x}{(1 - x_A)_{iM}}(x_{Ai} - x_{AL}) \qquad \textbf{(10.4-8)}$$

where $(1 - y_A)_{iM}$ and $(1 - x_A)_{iM}$ are defined by Eqs. (10.4-6) and (10.4-7). Multiplying the left-hand side of Eq. (10.4-8) by dA and the two right-side terms by $aS\,dz$ from Eq. (10.6-9),

$$N_A\,dA = \frac{k'_y a}{(1 - y_A)_{iM}}(y_{AG} - y_{Ai})S\,dz = \frac{k'_x a}{(1 - x_A)_{iM}}(x_{Ai} - x_{AL})S\,dz \qquad \textbf{(10.6-11)}$$

where $N_A\,dA$ = kg mol A transferred/s in height dz m (lb mol/h).

Equating Eq. (10.6-10) to (10.6-11) and using y_{AG} for the bulk gas phase and x_{AL} for the bulk liquid phase,

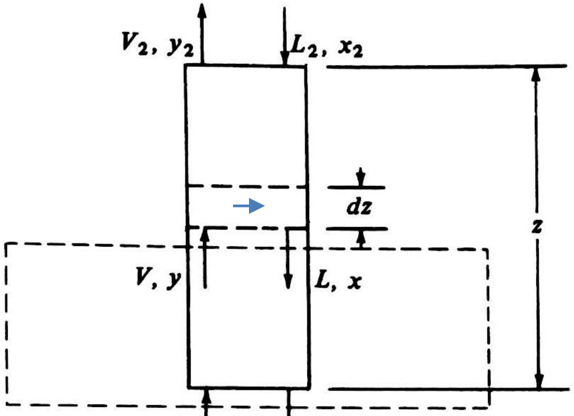
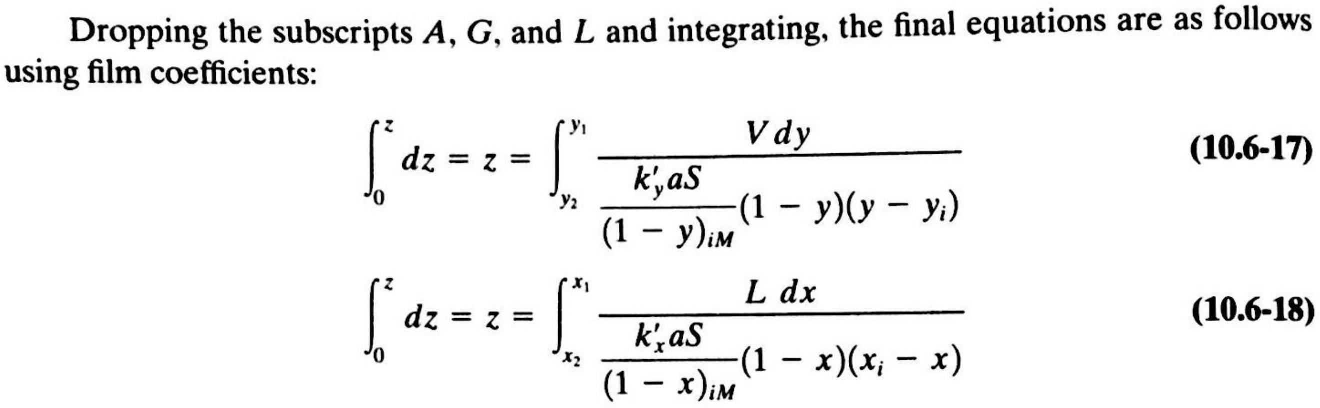


FIGURE 10.6-9. *Material balance for a countercurrent packed absorption tower.*



Dropping the subscripts A, G, and L and integrating, the final equations are as follows using film coefficients:

$$\int_0^z dz = z = \int_{y_2}^{y_1} \frac{V\,dy}{\dfrac{k'_y aS}{(1 - y)_{iM}}(1 - y)(y - y_i)} \qquad \textbf{(10.6-17)}$$

$$\int_0^z dz = z = \int_{x_2}^{x_1} \frac{L\,dx}{\dfrac{k'_x aS}{(1 - x)_{iM}}(1 - x)(x_i - x)} \qquad \textbf{(10.6-18)}$$

Derive model using Plug Flow Assumption: Create Differential Equations

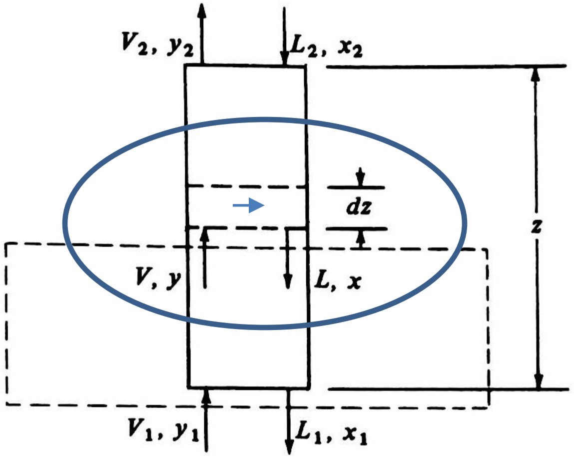
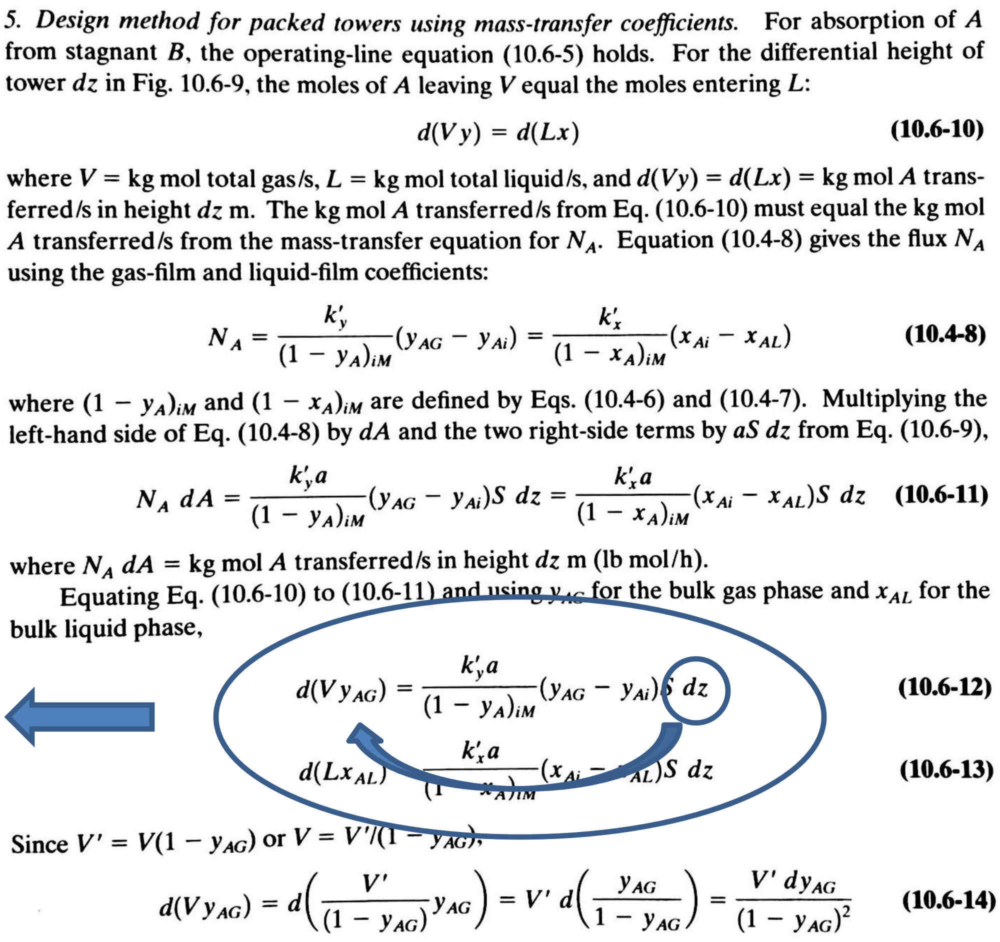


FIGURE 10.6-9. *Material balance for a countercurrent packed absorption tower.*



5. *Design method for packed towers using mass-transfer coefficients.* For absorption of A from stagnant B, the operating-line equation (10.6-5) holds. For the differential height of tower dz in Fig. 10.6-9, the moles of A leaving V equal the moles entering L:

$$d(Vy) = d(Lx) \qquad \textbf{(10.6-10)}$$

where V = kg mol total gas/s, L = kg mol total liquid/s, and $d(Vy) = d(Lx)$ = kg mol A transferred/s in height dz m. The kg mol A transferred/s from Eq. (10.6-10) must equal the kg mol A transferred/s from the mass-transfer equation for N_A. Equation (10.4-8) gives the flux N_A using the gas-film and liquid-film coefficients:

$$N_A = \frac{k'_y}{(1-y_A)_{iM}}(y_{AG} - y_{Ai}) = \frac{k'_x}{(1-x_A)_{iM}}(x_{Ai} - x_{AL}) \qquad \textbf{(10.4-8)}$$

where $(1-y_A)_{iM}$ and $(1-x_A)_{iM}$ are defined by Eqs. (10.4-6) and (10.4-7). Multiplying the left-hand side of Eq. (10.4-8) by dA and the two right-side terms by $aS\,dz$ from Eq. (10.6-9),

$$N_A\,dA = \frac{k'_y a}{(1-y_A)_{iM}}(y_{AG} - y_{Ai})S\,dz = \frac{k'_x a}{(1-x_A)_{iM}}(x_{Ai} - x_{AL})S\,dz \qquad \textbf{(10.6-11)}$$

where $N_A\,dA$ = kg mol A transferred/s in height dz m (lb mol/h).

Equating Eq. (10.6-10) to (10.6-11) and using y_{AG} for the bulk gas phase and x_{AL} for the bulk liquid phase,

$$d(Vy_{AG}) = \frac{k'_y a}{(1-y_A)_{iM}}(y_{AG} - y_{Ai})S\,dz \qquad \textbf{(10.6-12)}$$

$$d(Lx_{AL}) = \frac{k'_x a}{(1-x_A)_{iM}}(x_{Ai} - x_{AL})S\,dz \qquad \textbf{(10.6-13)}$$

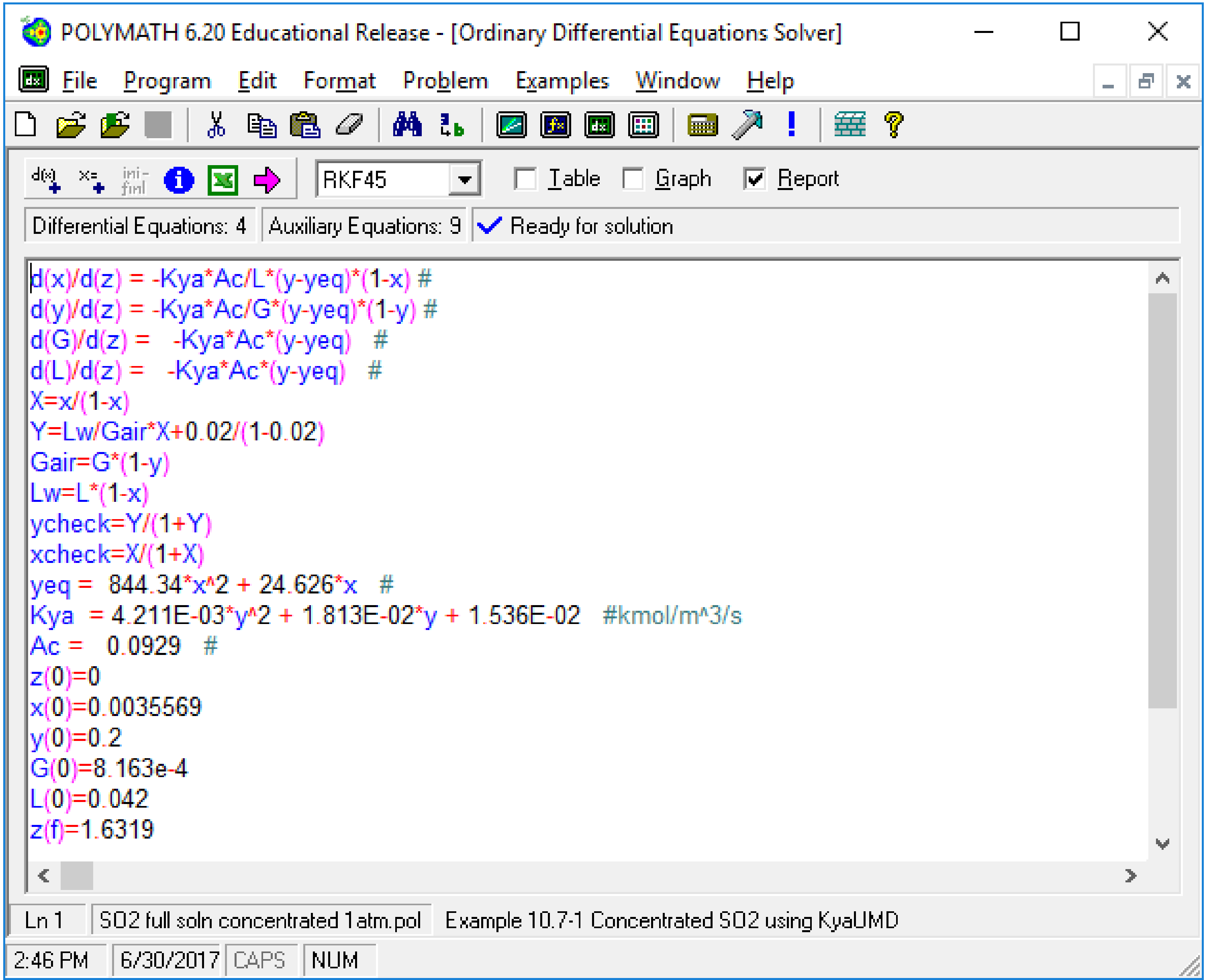
Since $V' = V(1-y_{AG})$ or $V = V'/(1-y_{AG})$,

$$d(Vy_{AG}) = d\left(\frac{V'}{(1-y_{AG})}y_{AG}\right) = V'\,d\left(\frac{y_{AG}}{1-y_{AG}}\right) = \frac{V'\,dy_{AG}}{(1-y_{AG})^2} \qquad \textbf{(10.6-14)}$$

$$\frac{d(Vy_{AG})}{dz} = -\frac{k'_y aS}{(1-y_A)_{iM}}(y_{AG} - y_{Ai})$$

$$\frac{d(Lx_{AG})}{dz} = -\frac{k'_x aS}{(1-x_A)_{iM}}(x_{Ai} - x_{AL})$$

Polymath Absorber Model



RKF45

☐ Table ☐ Graph ☑ Report

Differential Equations: 4 | Auxiliary Equations: 9 | ✓ Ready for solution

```
d(x)/d(z) = -Kya*Ac/L*(y-yeq)*(1-x) #
d(y)/d(z) = -Kya*Ac/G*(y-yeq)*(1-y) #
d(G)/d(z) =   -Kya*Ac*(y-yeq)   #
d(L)/d(z) =   -Kya*Ac*(y-yeq)   #
X=x/(1-x)
Y=Lw/Gair*X+0.02/(1-0.02)
Gair=G*(1-y)
Lw=L*(1-x)
ycheck=Y/(1+Y)
xcheck=X/(1+X)
yeq =  844.34*x^2 + 24.626*x   #
Kya  = 4.211E-03*y^2 + 1.813E-02*y + 1.536E-02   #kmol/m^3/s
Ac =   0.0929   #
z(0)=0
x(0)=0.0035569
y(0)=0.2
G(0)=8.163e-4
L(0)=0.042
z(f)=1.6319
```

Ln 1 | SO2 full soln concentrated 1atm.pol | Example 10.7-1 Concentrated SO2 using KyaUMD

2:46 PM | 6/30/2017 | CAPS | NUM

Introduction to Problem Solving with PolyMath, Excel and MATLAB

Robert Hesketh, Professor
Chemical Engineering
Rowan University
Glassboro, NJ 08028
hesketh@rowan.edu

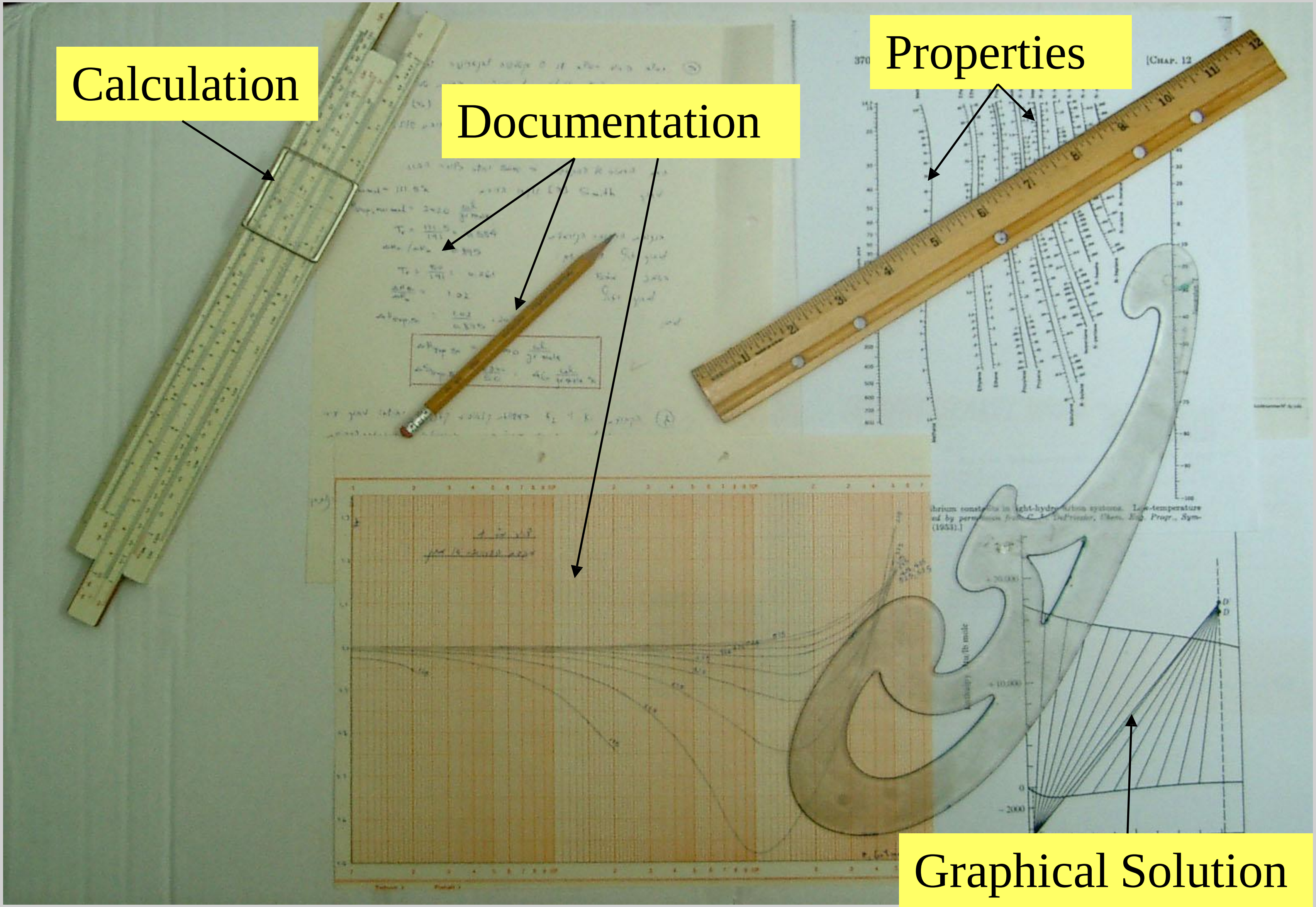
Michael B. Cutllip, Emeritus Professor
Dept. of Chemical & Biomolecular Engineering
University of Connecticut
Storrs, CT 06269-3222
michael.cutlip@uconn.edu

PLEASE READ BEFORE USING THIS PDF FILE

Users of this PDF file are recommended to utilize the latest version of Abobe Acrobat Reader DC. This the latest free Reader software that has enabled us to utilize many attachments. It also allows us to place icons in the text that will bring up and allow the reader to double click on an icon in the text to execute PolyMath and other software with the appropriate problem files. Please go to the the link below to upload and install the latest Acrobat Reader software. Also please install the latest PolyMath EDU Site 6.2 free software distributed with these ASEE Chemical Engineering Summer School materials.

https://get.adobe.com/reader/?promoid=8JD95JPQ&mv=other

Engineer's Tools - THEN in early 1960's

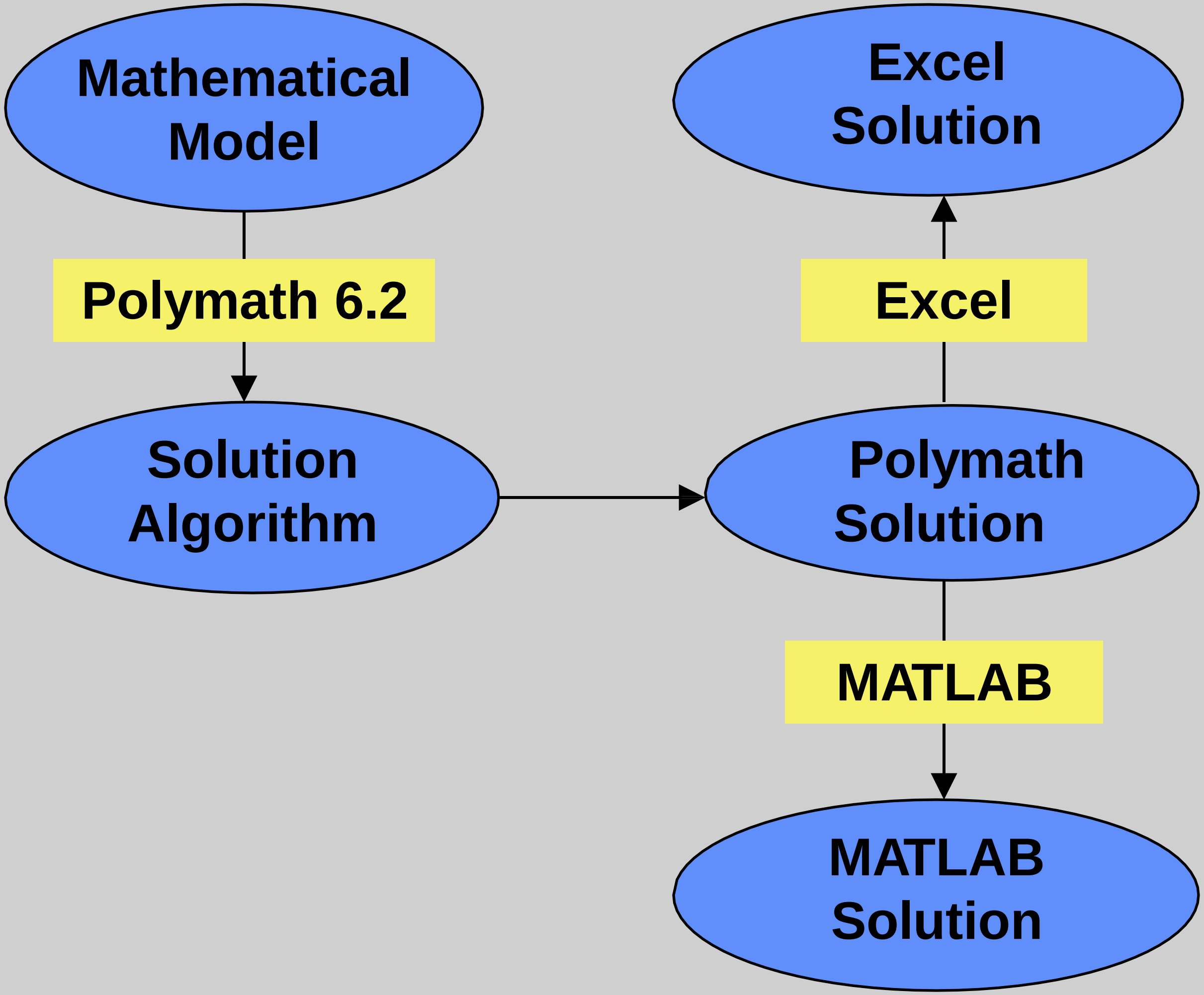


NOW! Increasing Problem Solving Efficiency and Capabilities with a Novel Combination of Software Tools

- **POLYMATH© (easy problem formulation)**
- **Excel™ (familiar spreadsheet environment)**
- **MATLAB™ (advanced problem solving)**

Students and Faculty at their personal computers or in computer labs can now effectively solve problems using all the above packages.

Desktop Problem Solving Involving Polymath, Excel, and MATLAB



Mathematical Model

Polymath 6.2

Solution Algorithm

Polymath Solution

Excel

Excel Solution

MATLAB

MATLAB Solution

POLYMATH Educational 6.2

Numerical Computation Package

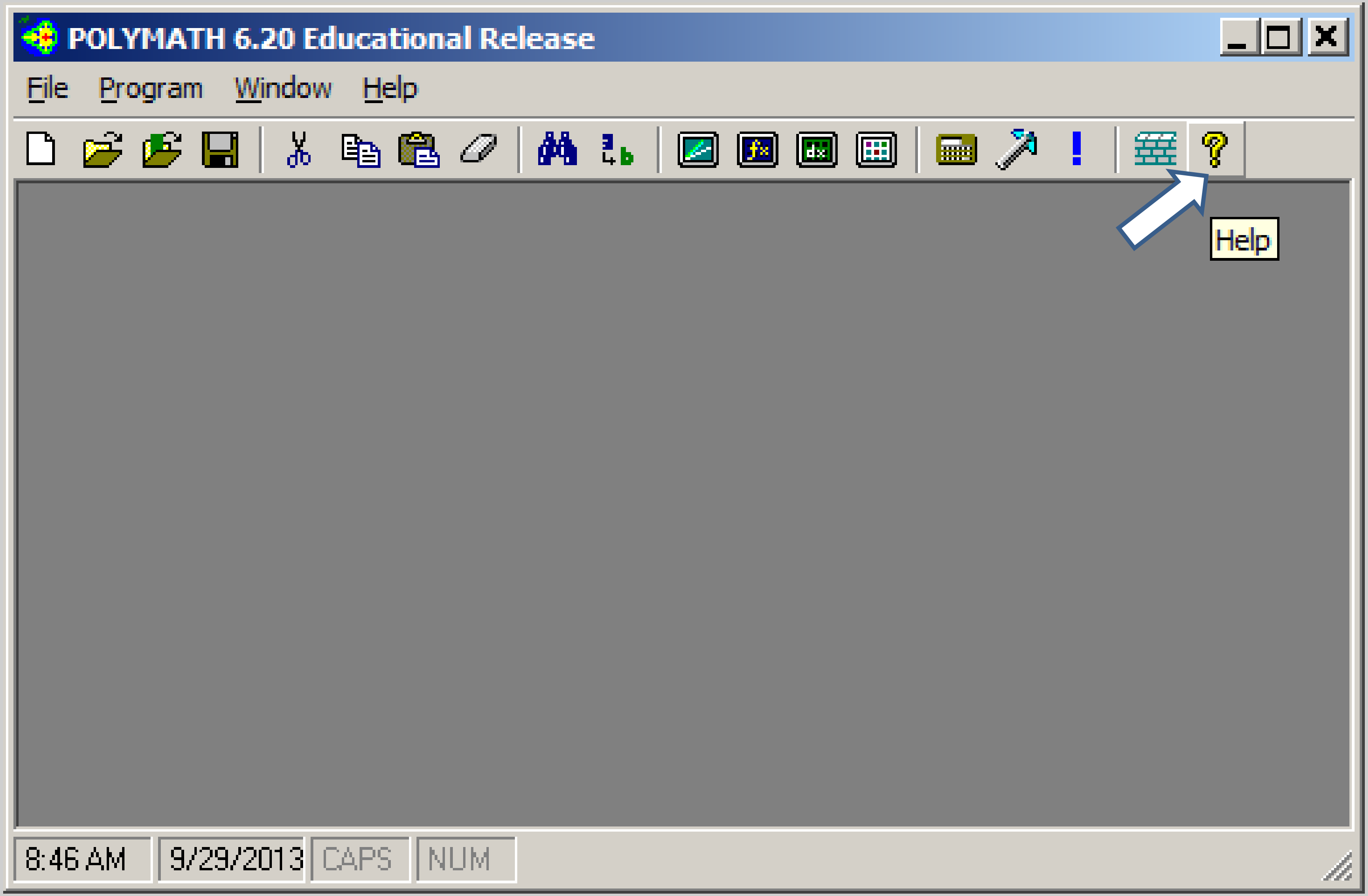
- Extremely Easy-to-Use
- Excellent Problem Solving Capabilities
  - Linear Equations – 100 (264) Professional Version
  - Nonlinear Equations – 30 (300)
  - Differential Equations – 30 (300)
  - Regressions (Linear, Polynomial, Multiple Linear, Nonlinear) - 301 data points (1001)

    - **Automated Export of Problems to Working Excel Spreadsheets Enabling Stand-Alone Excel Calculations (Provides Add-In for Excel that Solves ODEs.**
    - **Enables the Use MATLAB by Automatically Translating Problems to Code for Use in M-files.**

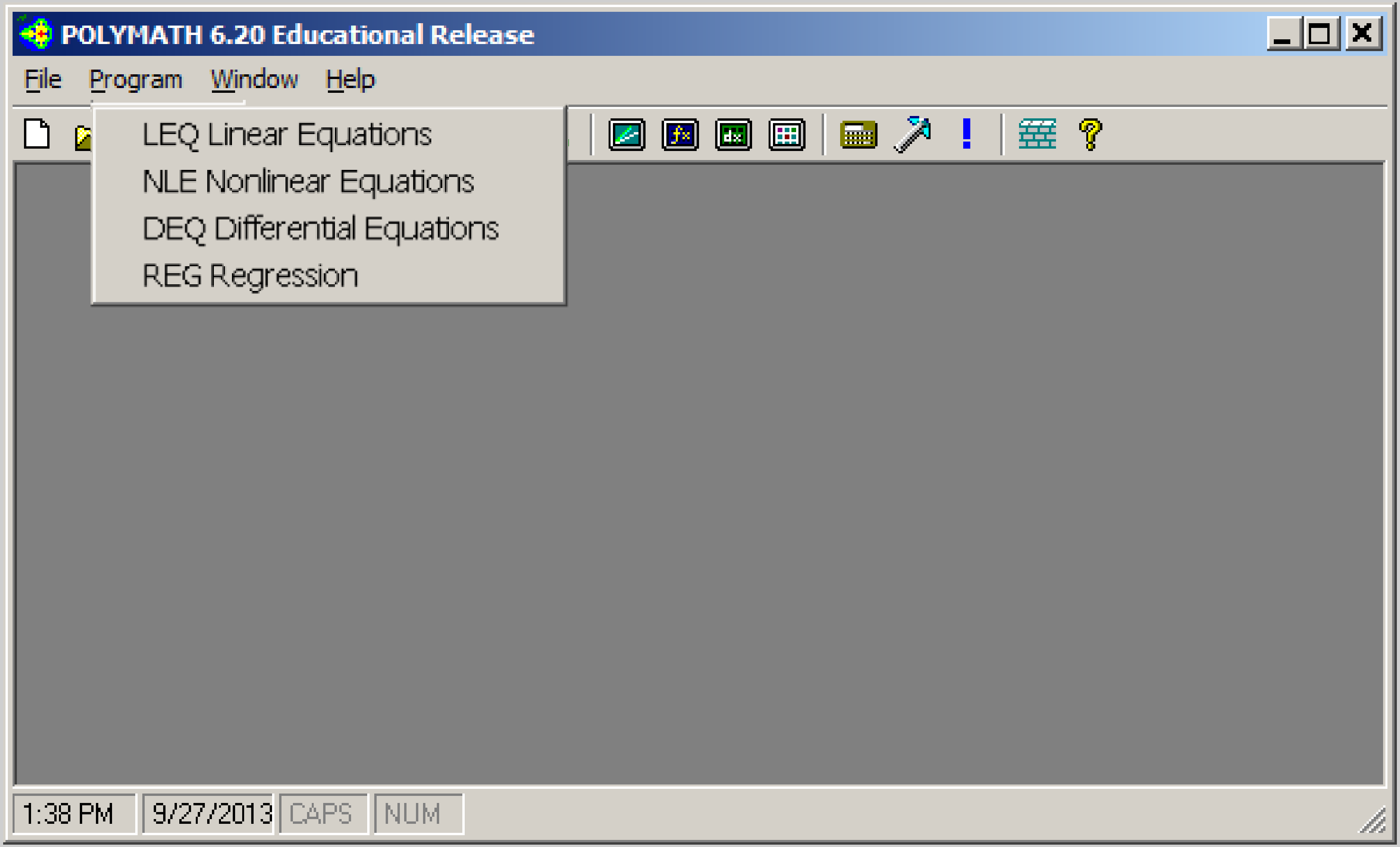
POLYMATH 6.2 features include:

- EASE OF USE WITHOUT ANY PROGRAMMING LANGUAGES OR CONTROL LANGUAGES TO REMEMBER
- STANDARD WINDOWS EDITING
- EXTENSIVE USER ALGORITHM SELECTION AND CONTROL
- EXECUTION WITH ALL 32-BIT AND 64-BIY WINDOWS OPERATING SYSTEMS INCLUDING WIN 8
- COMPATIBILITY WITH PREVIOUS VERSIONS
- THREE ON-BOARD UTILITIES: POWERFUL CALCULATOR, UNIT CONVERTER, AND EXTENSIVE ENGINEERING CONVERSION FACTORS
- EXTENSIVE ON-LINE DOCUMENTATION
- AUTOMATIC PROBLEM EXPORT TO EXCEL – EXCEL ADD-IN FOR DIFFERENTIAL EQUATIONS
- MATLAB OUTPUT GIVING ORDERED AND FORMATTED EQUATIONS

Initial Polymath Software Display with Help that Gives Detailed Information on the Software

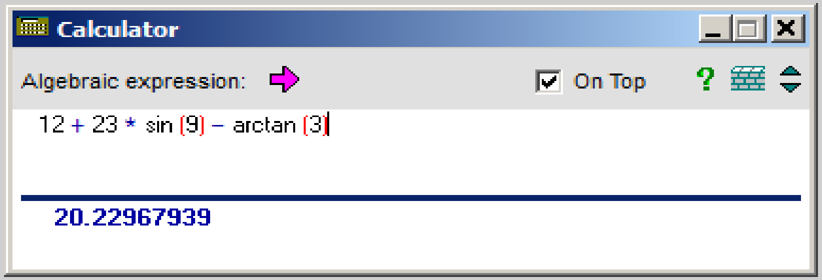


Polymath Software has Four Main Programs



Polymath Software also has Three Utilities:

- Calculator



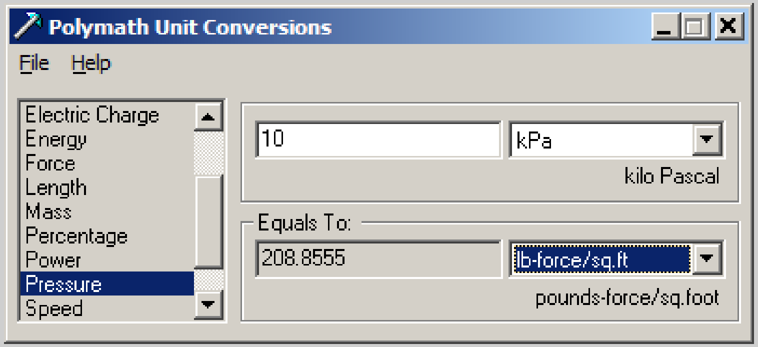
Algebraic expression:

☑ On Top

12 + 23 * sin (9) – arctan (3)

20.22967939

- Units Converter



10 kPa

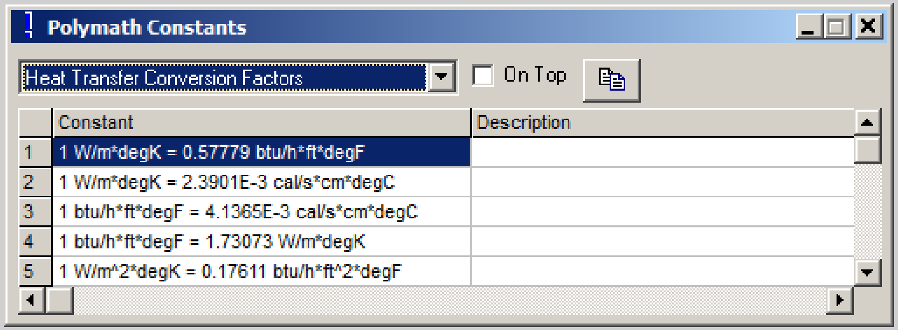
kilo Pascal

Equals To:

208.8555 lb-force/sq.ft

pounds-force/sq.foot

- Scientific Constants

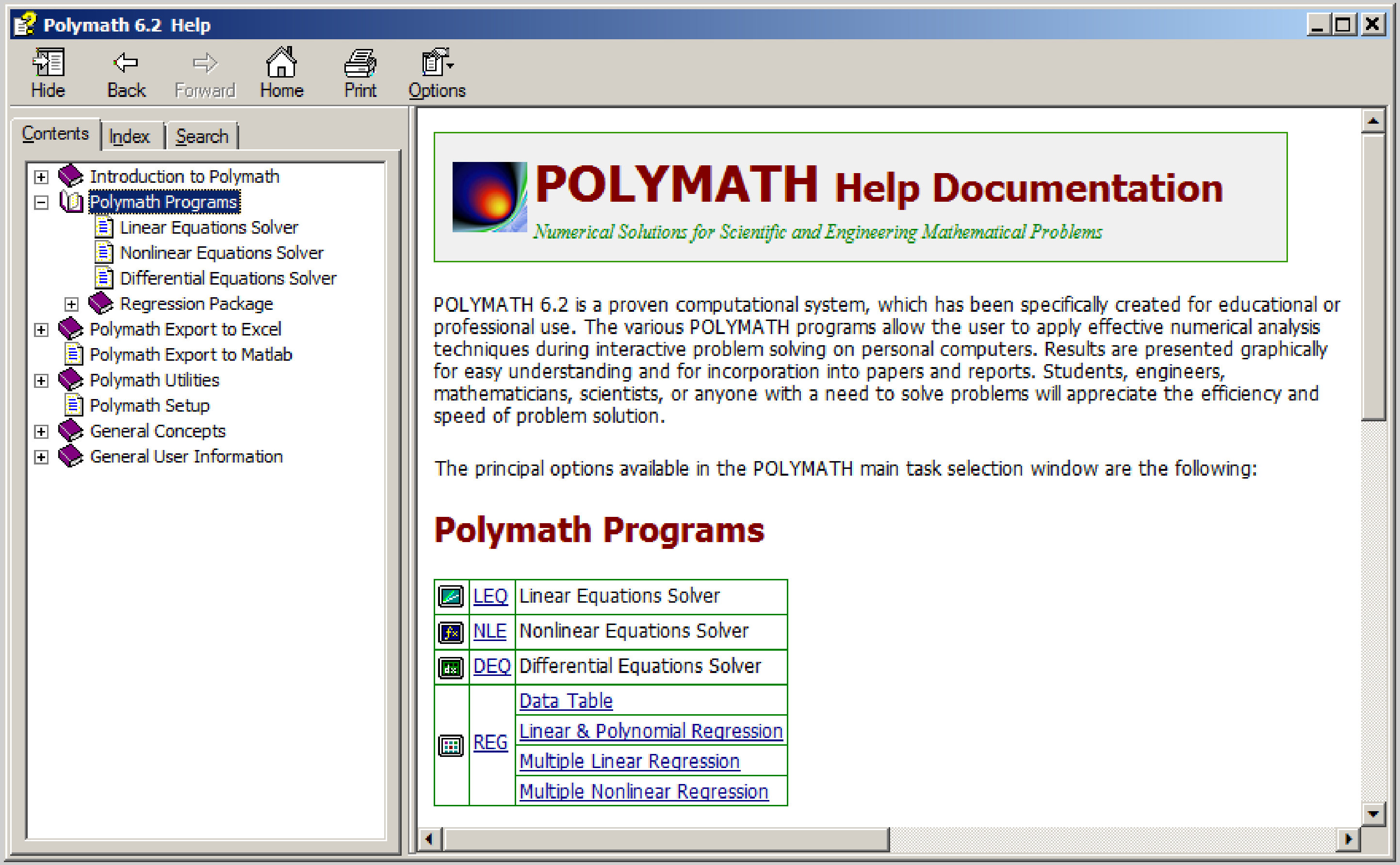


Heat Transfer Conversion Factors

☐ On Top

	Constant	Description
1	1 W/m*degK = 0.57779 btu/h*ft*degF	
2	1 W/m*degK = 2.3901E-3 cal/s*cm*degC	
3	1 btu/h*ft*degF = 4.1365E-3 cal/s*cm*degC	
4	1 btu/h*ft*degF = 1.73073 W/m*degK	
5	1 W/m^2*degK = 0.17611 btu/h*ft^2*degF	

Polymath Software has Extensive HELP



POLYMATH Help Documentation

Numerical Solutions for Scientific and Engineering Mathematical Problems

POLYMATH 6.2 is a proven computational system, which has been specifically created for educational or professional use. The various POLYMATH programs allow the user to apply effective numerical analysis techniques during interactive problem solving on personal computers. Results are presented graphically for easy understanding and for incorporation into papers and reports. Students, engineers, mathematicians, scientists, or anyone with a need to solve problems will appreciate the efficiency and speed of problem solution.

The principal options available in the POLYMATH main task selection window are the following:

Polymath Programs

LEQ	Linear Equations Solver
NLE	Nonlinear Equations Solver
DEQ	Differential Equations Solver
REG	Data Table Linear & Polynomial Regression Multiple Linear Regression Multiple Nonlinear Regression

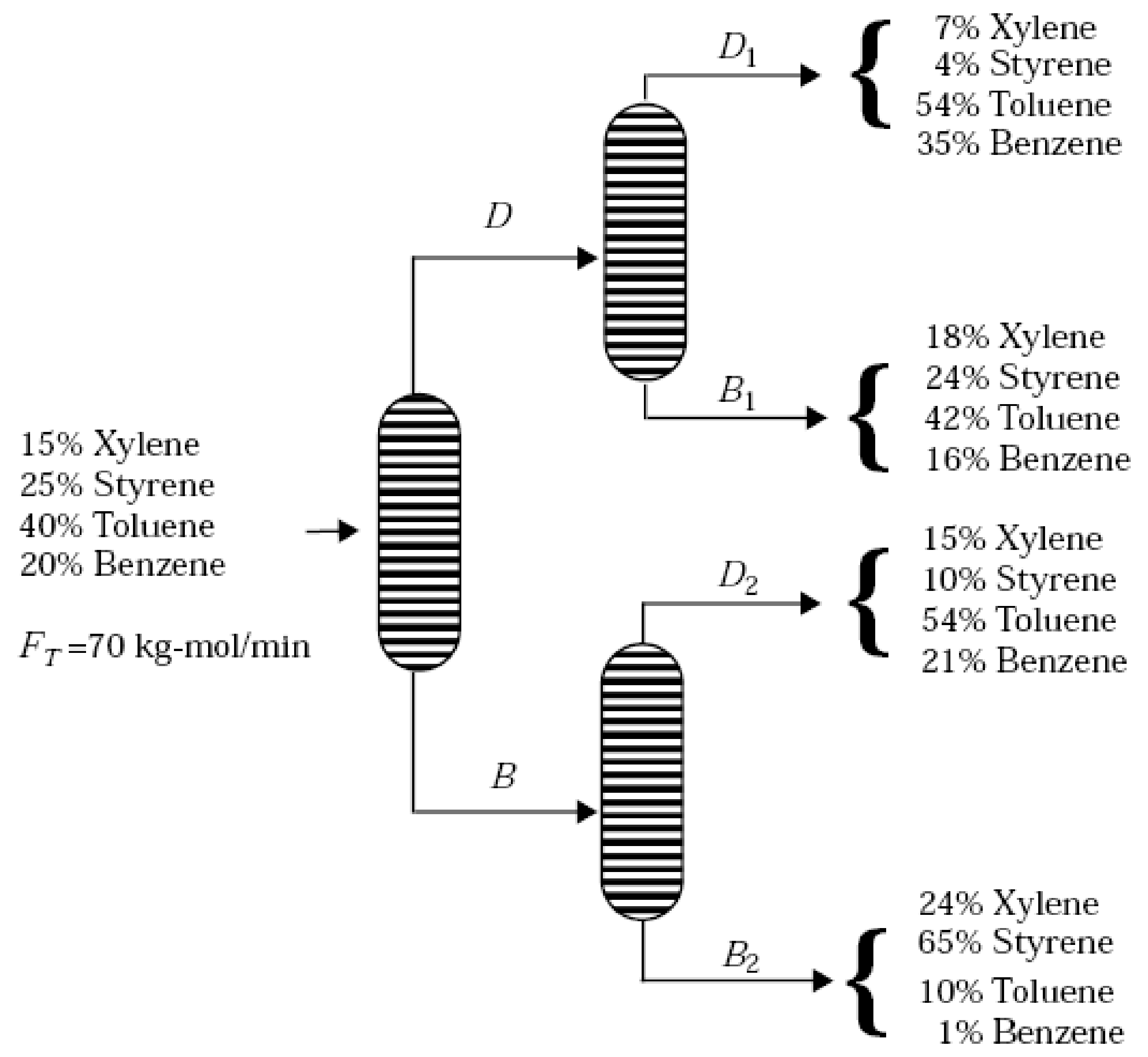
Open Polymath on your computer and look at the HELP to learn more about the program.

You can open Polymath HELP by clicking on the ? found at the top Polymath program menu. Alternately you can open a window containing HELP by double clicking on 📎 or by going to the Attachments list (click on the arrow icon on the left margin) and then by clicking on the paper clip and then double clicking on Polymath.pol followed by clicking ?.

Introductory Problems

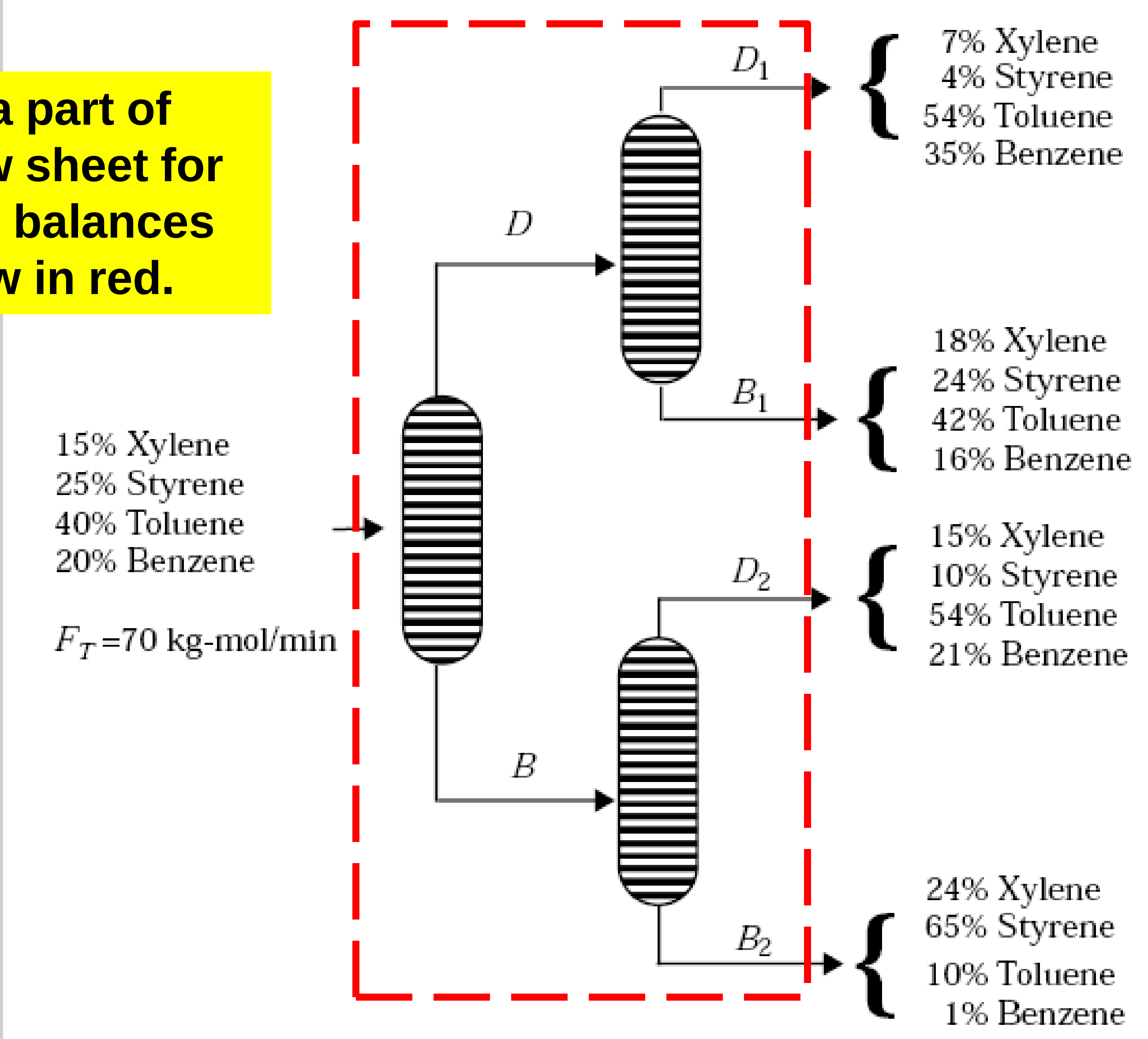
1. Linear Equations – Material Balances for Distillation Columns – Polymath
2. Explicit Calculations – Equation of State – Polymath and Excel
3. Nonlinear Equations – Pressure Drop for Pipe Flow – Polymath and Excel
4. Differential Equations – Series Reactions in a Batch Reactor - Polymath, Excel, and MATLAB
5. Regression – Hardening of Concrete (Multiple Linear Regression) - Polymath, Excel
6. Regressions – Vapor Pressure Data (Linear and Nonlinear) - Polymath, Excel

Problem 1 – Material Balances for Distillation Columns
Determine the Flow Rates B_1, D_1, B_2, and D_2



Linear Equations – Material Balance Problem
Determine the Flow Rates B_1, D_1, B_2, and D_2

Select a part of the flow sheet for making balances as show in red.



Linear Equations – Material Balance Problem to Determine the Flow Rates B_1, D_1, B_2, and D_2

$$\text{Xylene: } 0.07D_1 + 0.18B_1 + 0.15D_2 + 0.24B_2 = 0.15 \times 70$$

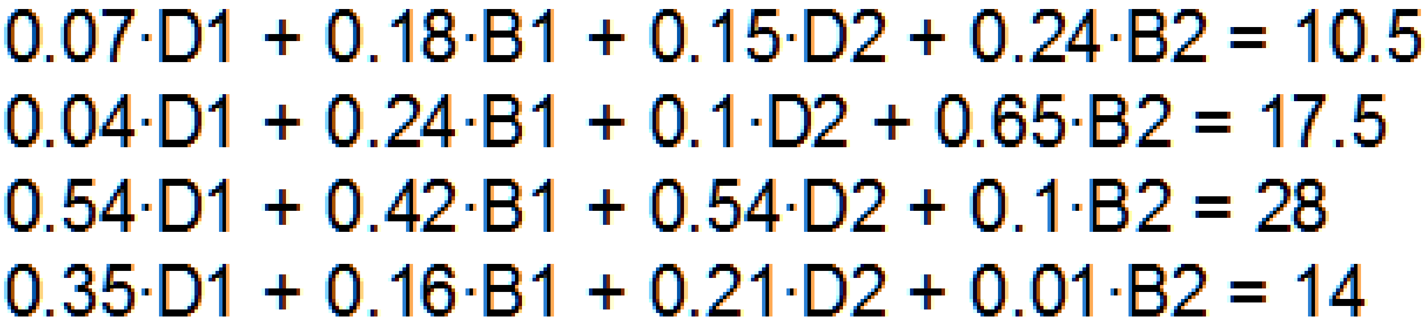
$$\text{Styrene: } 0.04D_1 + 0.24B_1 + 0.10D_2 + 0.65B_2 = 0.25 \times 70$$

$$\text{Toluene: } 0.54D_1 + 0.42B_1 + 0.54D_2 + 0.10B_2 = 0.40 \times 70$$

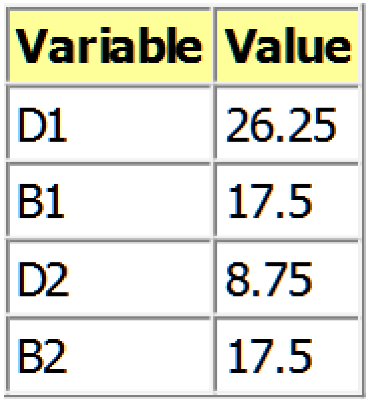
$$\text{Benzene: } 0.35D_1 + 0.16B_1 + 0.21D_2 + 0.01B_2 = 0.20 \times 70$$

Make Balances on Each Species:
Xylene
Styrene
Toluene
Benzene

Linear Equations – Material Balance Problem to Determine the Flow Rates B_1, D_1, B_2, and D_2



$$0.07 \cdot D1 + 0.18 \cdot B1 + 0.15 \cdot D2 + 0.24 \cdot B2 = 10.5$$
$$0.04 \cdot D1 + 0.24 \cdot B1 + 0.1 \cdot D2 + 0.65 \cdot B2 = 17.5$$
$$0.54 \cdot D1 + 0.42 \cdot B1 + 0.54 \cdot D2 + 0.1 \cdot B2 = 28$$
$$0.35 \cdot D1 + 0.16 \cdot B1 + 0.21 \cdot D2 + 0.01 \cdot B2 = 14$$



Variable	Value
D1	26.25
B1	17.5
D2	8.75
B2	17.5

Demonstration of the Actual Polymath Program

You can go to POLYMATH with the program ready for solution with a Double click on file name from attachments list. You may need to first click on the right arrow on left margin and then the paper clip.

OR

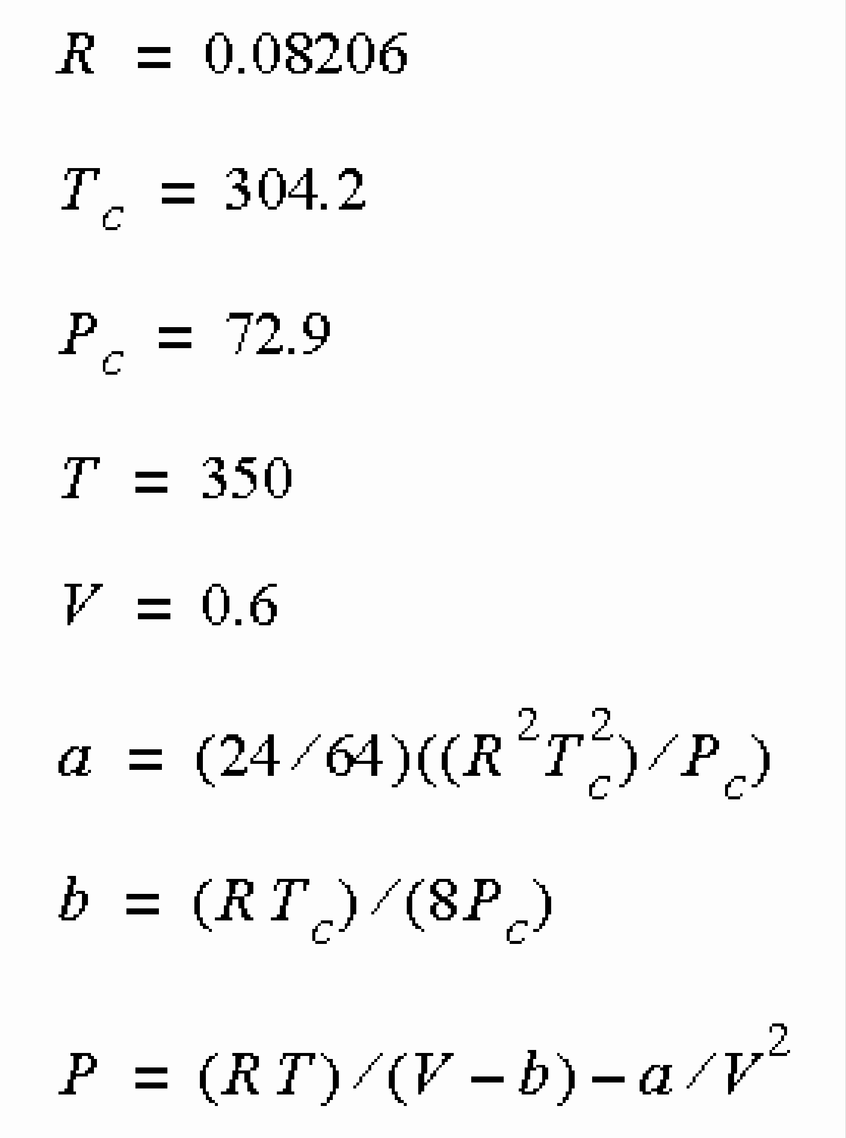
You can Double Click on the Link icon below to bring up the Polymath program with the problem already entered.

POLYMATH – This file attachment name is LinearEquations01.pol

Problem 2 - Explicit Calculations for an Equation of State

Calculate P when the other variables and parameters of the van der Waals equation of state are known.

Hint: Use POLYMATH Nonlinear Equations Option (even when all equations are explicit).



$$R = 0.08206$$

$$T_c = 304.2$$

$$P_c = 72.9$$

$$T = 350$$

$$V = 0.6$$

$$a = (24/64)((R^2 T_c^2)/P_c)$$

$$b = (R T_c)/(8 P_c)$$

$$P = (R T)/(V - b) - a/V^2$$

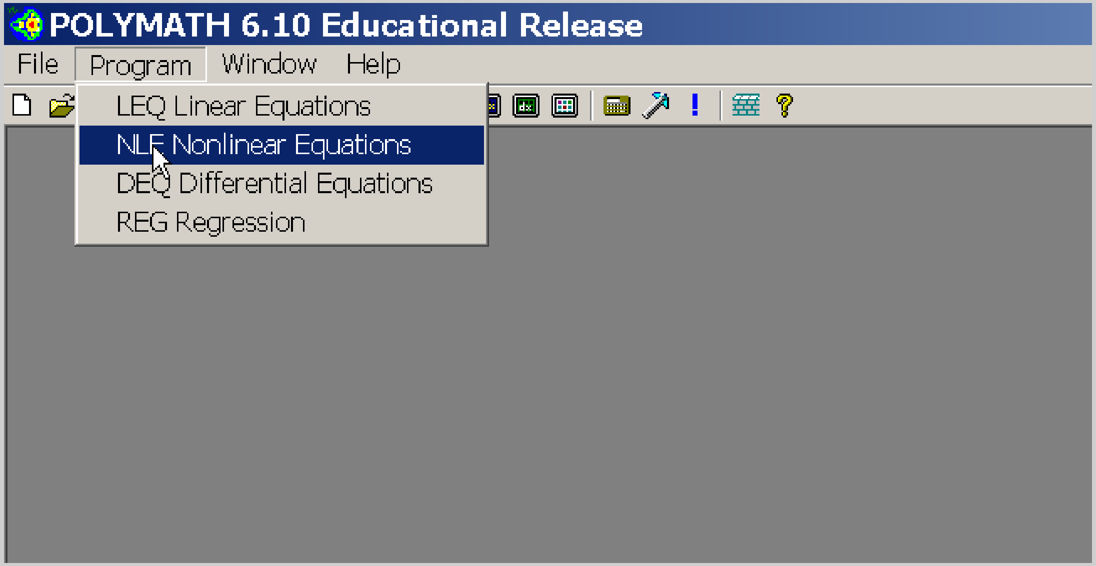
Problem 2 – Explicit Calculations for an Equation of State

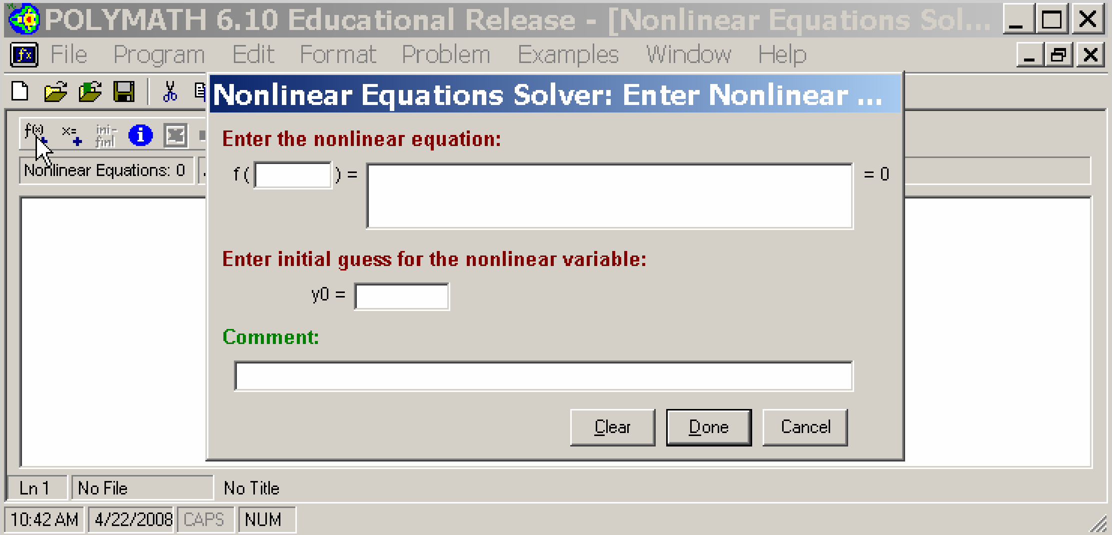
Polymath Solution Demonstration

Enter the equations into Polymath.

Note that the equations can be entered in any order. Polymath orders equations before solution.

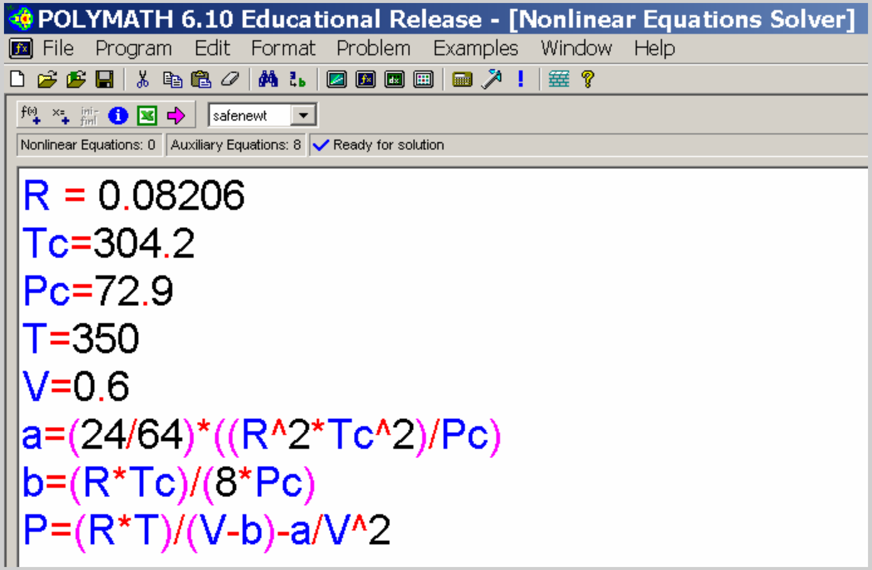
Use templates or full screen editor.





Problem 2 – Explicit Calculations for an Equation of State

Polymath Solution Exercise

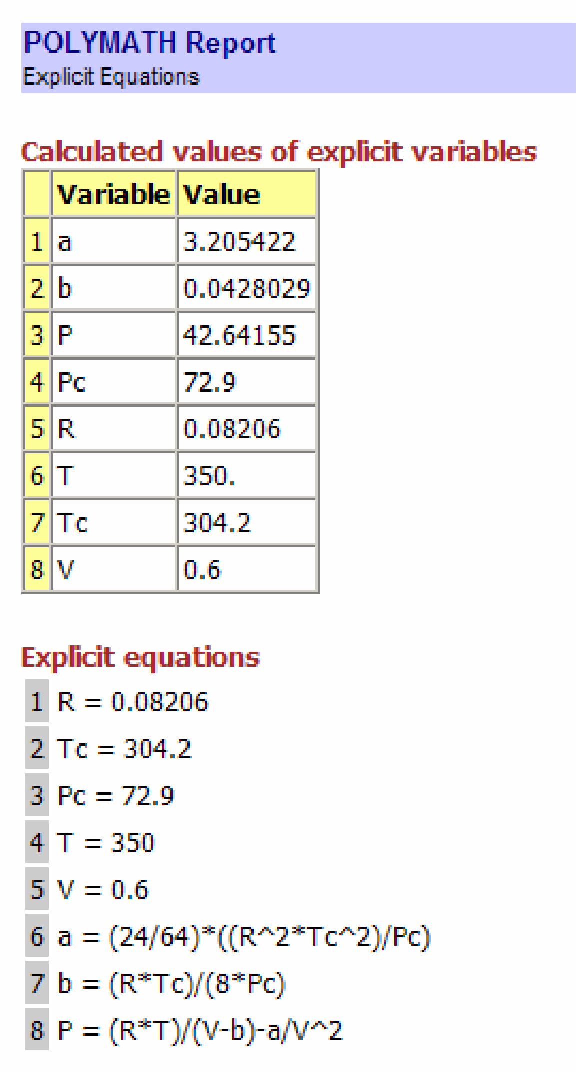


Nonlinear Equations: 0

Auxiliary Equations: 8

✓ Ready for solution

```
R = 0.08206
Tc=304.2
Pc=72.9
T=350
V=0.6
a=(24/64)*((R^2*Tc^2)/Pc)
b=(R*Tc)/(8*Pc)
P=(R*T)/(V-b)-a/V^2
```



POLYMATH Report

Explicit Equations

Calculated values of explicit variables

	Variable	Value
1	a	3.205422
2	b	0.0428029
3	P	42.64155
4	Pc	72.9
5	R	0.08206
6	T	350.
7	Tc	304.2
8	V	0.6

Explicit equations

1 R = 0.08206
2 Tc = 304.2
3 Pc = 72.9
4 T = 350
5 V = 0.6
6 a = (24/64)*((R^2*Tc^2)/Pc)
7 b = (R*Tc)/(8*Pc)
8 P = (R*T)/(V-b)-a/V^2

Use Polymath to enter and solve equations

OR

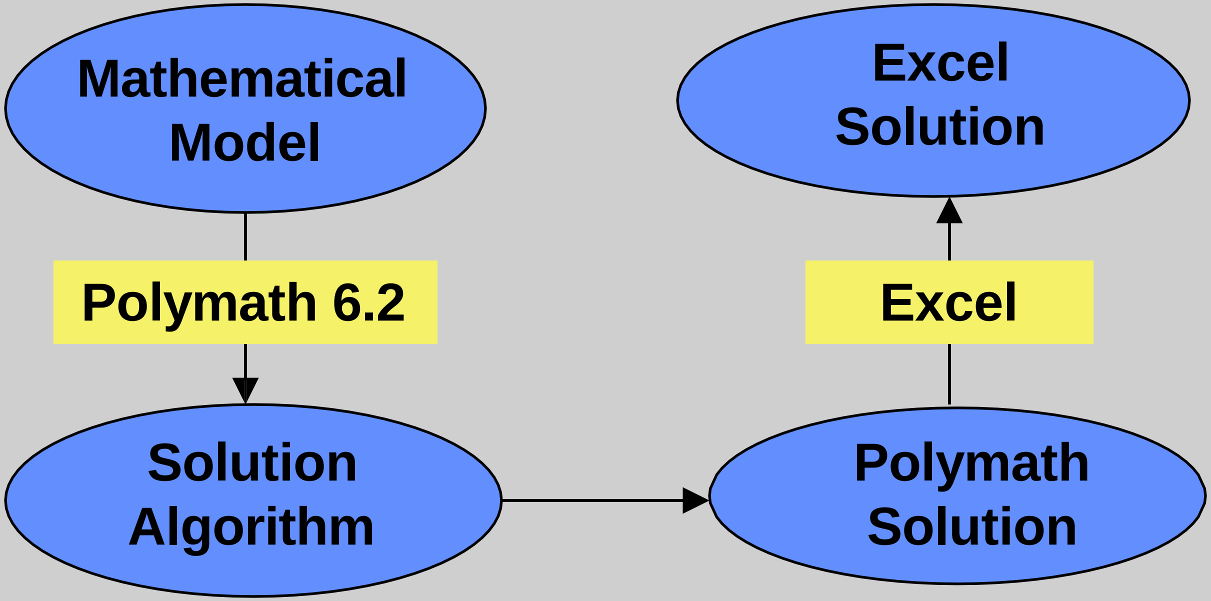
Execute this problem solution with Polymath and verify the given Polymath Report solution.

PolymathNonlinear.pol

NonlinearEquations01.pol

Problem 2 - Explicit Calculations for an Equation of State

Polymath Solution then Export to Excel for Solution



Mathematical Model

Polymath 6.2

Solution Algorithm

Polymath Solution

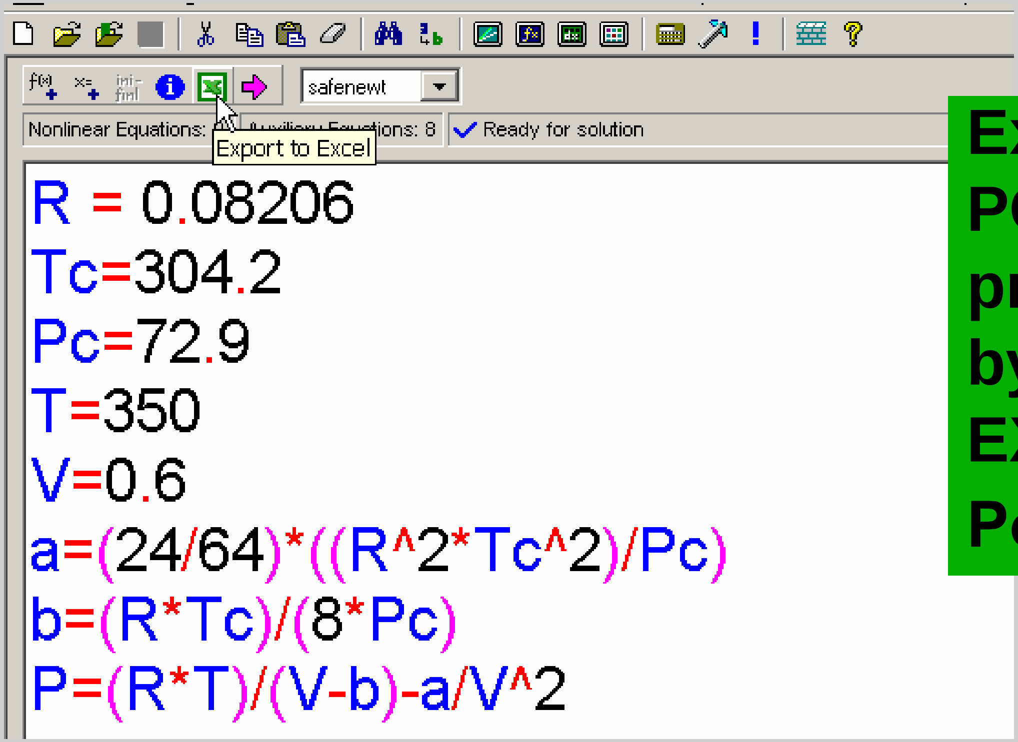
Excel

Excel Solution

Problem 2 – Explicit Calculations for an Equation of State

Polymath Solution then Export to Excel for Solution

Exercise



```
R = 0.08206
Tc=304.2
Pc=72.9
T=350
V=0.6
a=(24/64)*((R^2*Tc^2)/Pc)
b=(R*Tc)/(8*Pc)
P=(R*T)/(V-b)-a/V^2
```

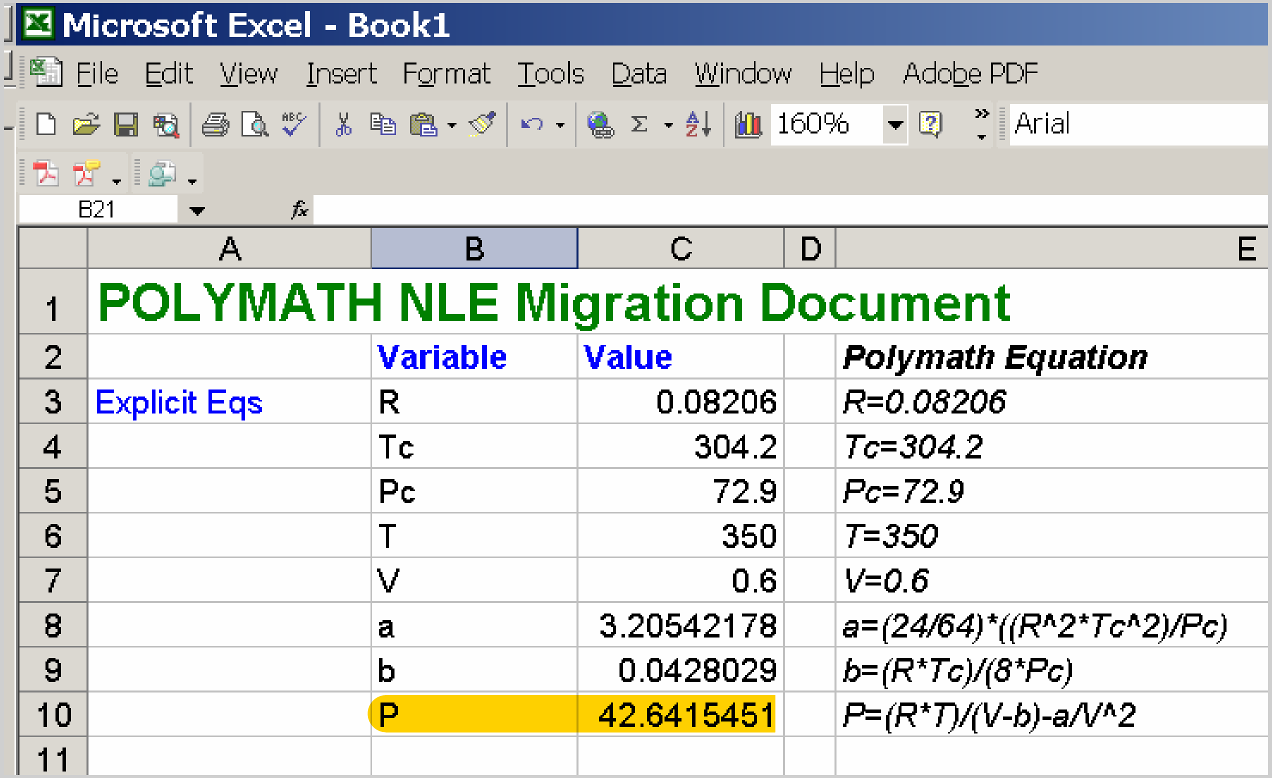
Export the POLYMATH problem to EXCEL by clicking the EXCEL icon in the Polymath Program.

Hint – Be sure to have an open Excel Spreadsheet running on your computer before exporting problem. Open Excel manually or click on attachment Excel.xls.

Problem 2 – Explicit Calculations for an Equation of State

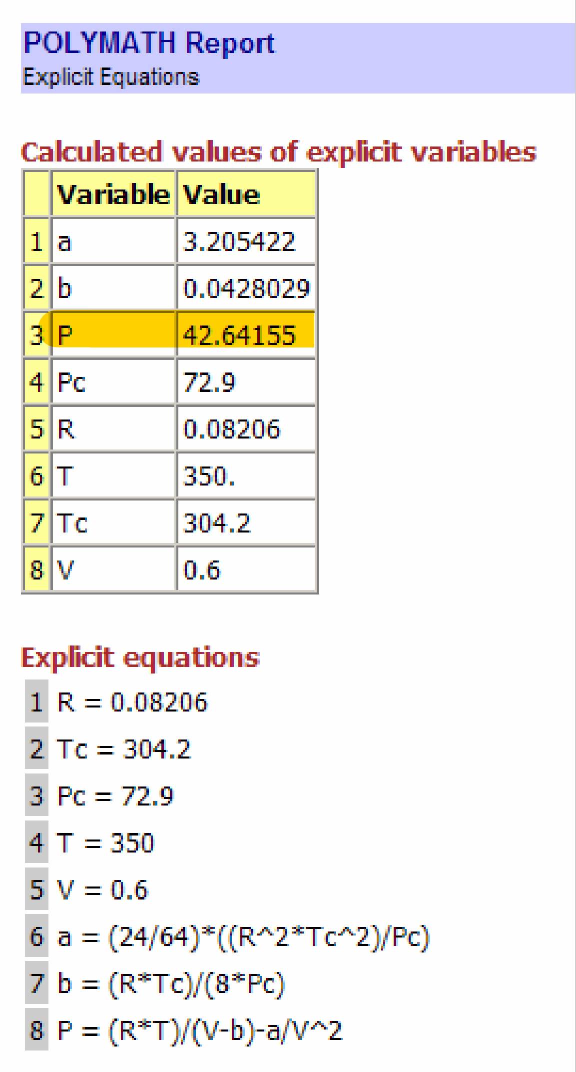
Polymath Solution then Export to Excel for Solution

Exercise



	A	B	C	D	E
1	POLYMATH NLE Migration Document				
2		Variable	Value		Polymath Equation
3	Explicit Eqs	R	0.08206		R=0.08206
4		Tc	304.2		Tc=304.2
5		Pc	72.9		Pc=72.9
6		T	350		T=350
7		V	0.6		V=0.6
8		a	3.20542178		a=(24/64)*((R^2*Tc^2)/Pc)
9		b	0.0428029		b=(R*Tc)/(8*Pc)
10		P	42.6415451		P=(R*T)/(V-b)-a/V^2
11					

Compare your EXCEL results to the POLYMATH results.



POLYMATH Report

Explicit Equations

Calculated values of explicit variables

	Variable	Value
1	a	3.205422
2	b	0.0428029
3	P	42.64155
4	Pc	72.9
5	R	0.08206
6	T	350.
7	Tc	304.2
8	V	0.6

Explicit equations

1 R = 0.08206
2 Tc = 304.2
3 Pc = 72.9
4 T = 350
5 V = 0.6
6 a = (24/64)*((R^2*Tc^2)/Pc)
7 b = (R*Tc)/(8*Pc)
8 P = (R*T)/(V-b)-a/V^2

Problem 3 – Pressure Drop Calculation for Pipe Flow

Polymath Solution for Two Nonlinear Equations – Simultaneous Solution with If… Then… Else… Statement

The second nonlinear equation uses the If… Then... Else Statement

Friction Factor Equation

fF = 16 / Re if Re < 2100

= 1 / (4 * *log*(Re * *sqrt*(fF)) - 0.4) ^ 2 if Re >= 2100

becomes in Polymath

f(fF) = **If** (Re < 2100) **Then** (fF - 16 / Re)

Else (fF - 1 / (4 * *log*(Re * *sqrt*(fF)) - 0.4) ^ 2)

Problem 3 – Pressure Drop Calculation for Pipe Flow

Polymath Solution for Two Nonlinear Equations – Simultaneous Solution with If… Then… Else… Statement

Pressure Drop Equation

dp = 2 * fF * rho * v * v * L / D

becomes in Polymath

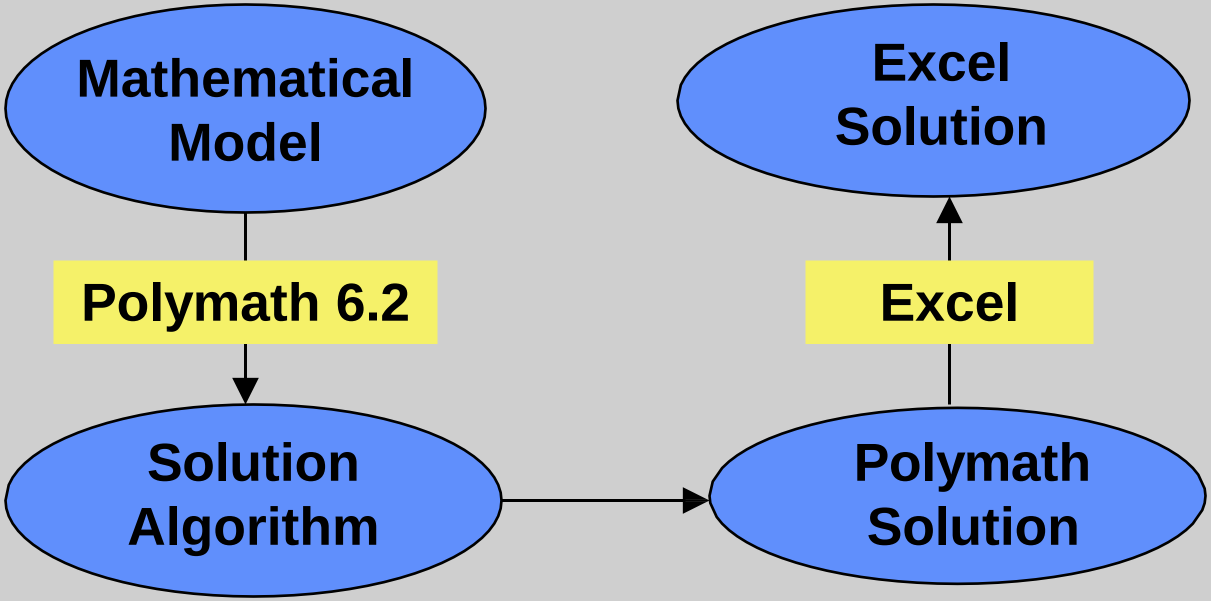
The nonlinear equation is rearranged to equal zero.

f(D) = dp - 2 * fF * rho * v * v * L / D

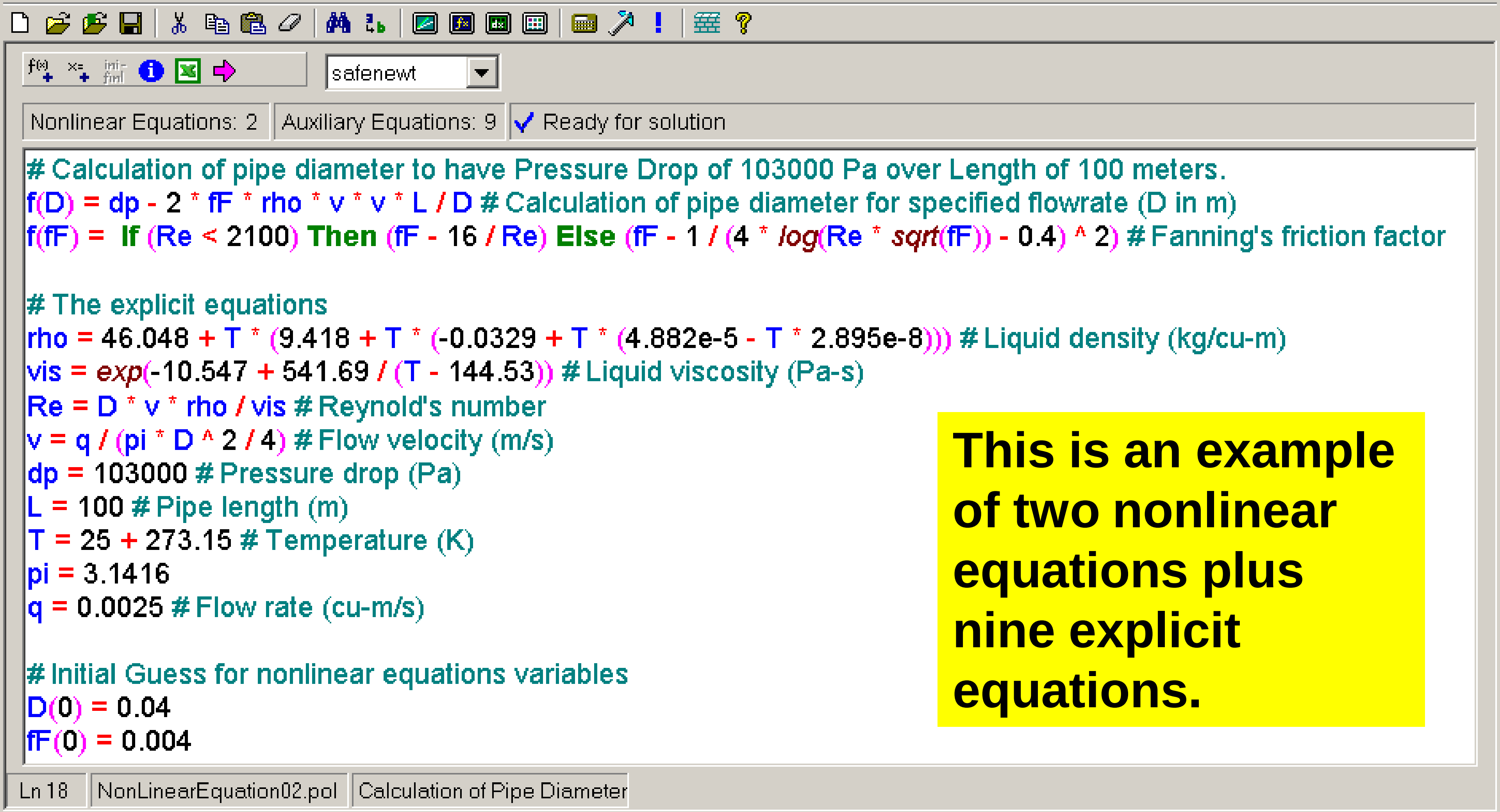
Problem 3 – Pressure Drop Calculation for Pipe Flow

Polymath Solution for Two Nonlinear Equations – Simultaneous Solution with If… Then… Else… Statement

Solution will be made in Polymath and Excel



Problem 3 – Pressure Drop Calculation for Pipe Flow Two Nonlinear Equations – Simultaneous Solution with If… Then… Else… Statement



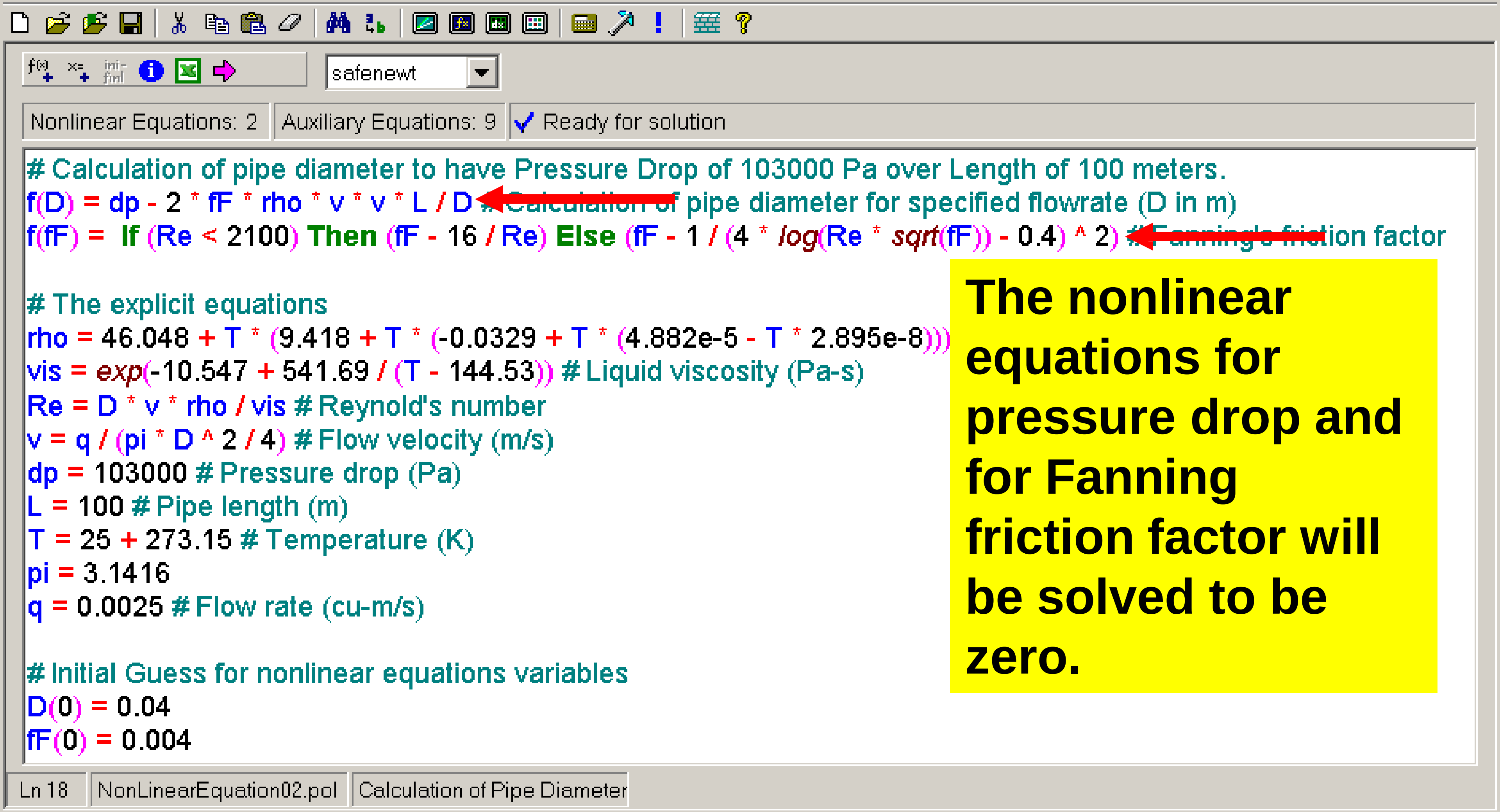
Nonlinear Equations: 2 | Auxiliary Equations: 9 | ✓ Ready for solution

```
# Calculation of pipe diameter to have Pressure Drop of 103000 Pa over Length of 100 meters.
f(D) = dp - 2 * fF * rho * v * v * L / D # Calculation of pipe diameter for specified flowrate (D in m)
f(fF) =  If (Re < 2100) Then (fF - 16 / Re) Else (fF - 1 / (4 * log(Re * sqrt(fF)) - 0.4) ^ 2) # Fanning's friction factor

# The explicit equations
rho = 46.048 + T * (9.418 + T * (-0.0329 + T * (4.882e-5 - T * 2.895e-8))) # Liquid density (kg/cu-m)
vis = exp(-10.547 + 541.69 / (T - 144.53)) # Liquid viscosity (Pa-s)
Re = D * v * rho / vis # Reynold's number
v = q / (pi * D ^ 2 / 4) # Flow velocity (m/s)
dp = 103000 # Pressure drop (Pa)
L = 100 # Pipe length (m)
T = 25 + 273.15 # Temperature (K)
pi = 3.1416
q = 0.0025 # Flow rate (cu-m/s)

# Initial Guess for nonlinear equations variables
D(0) = 0.04
fF(0) = 0.004
```

Ln 18 | NonLinearEquation02.pol | Calculation of Pipe Diameter

This is an example of two nonlinear equations plus nine explicit equations.

Problem 3 – Pressure Drop Calculation for Pipe Flow Two Nonlinear Equations – Simultaneous Solution with If… Then… Else… Statement



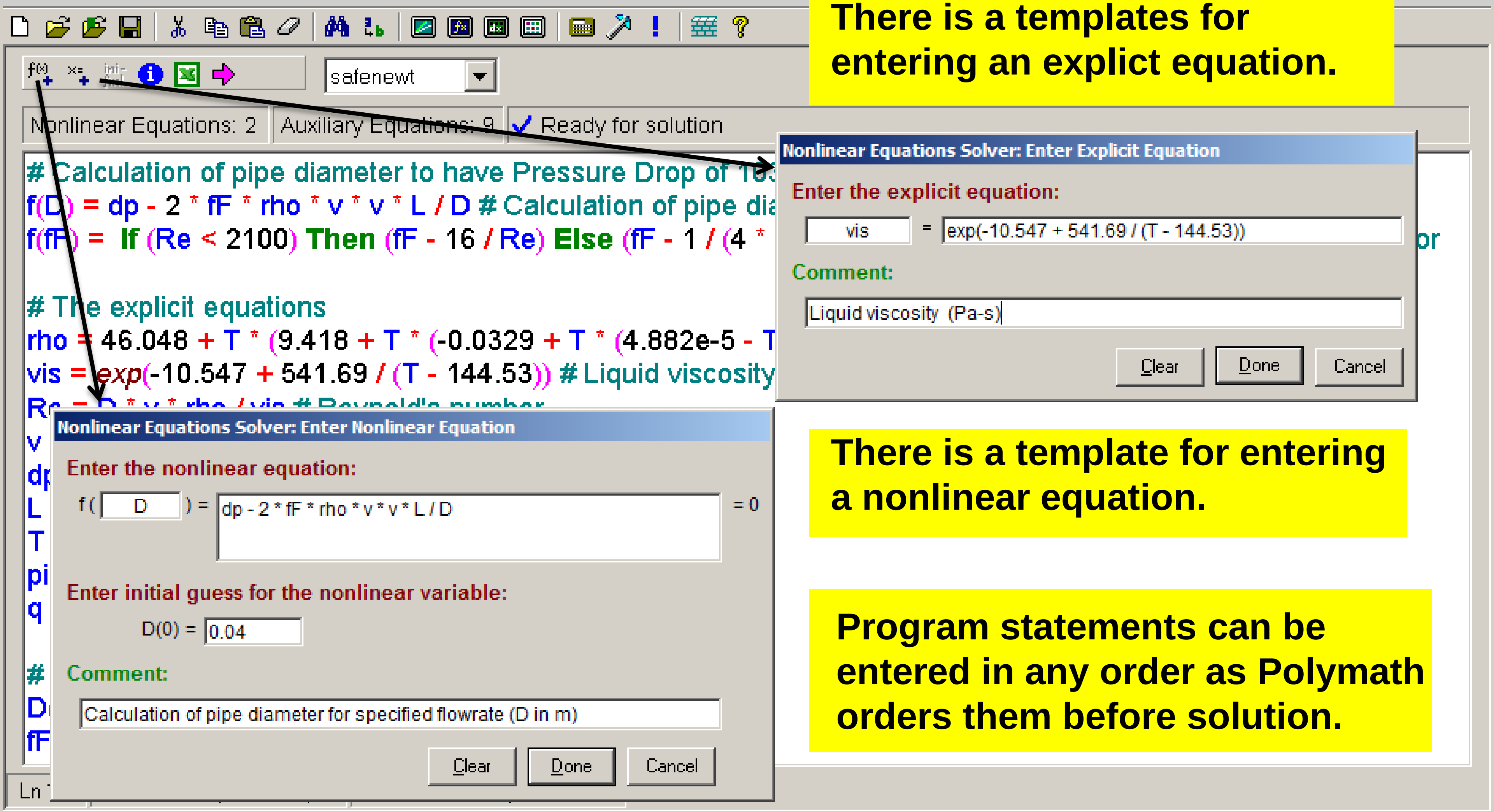
Nonlinear Equations: 2 | Auxiliary Equations: 9 | ✓ Ready for solution

```
# Calculation of pipe diameter to have Pressure Drop of 103000 Pa over Length of 100 meters.
f(D) = dp - 2 * fF * rho * v * v * L / D # Calculation of pipe diameter for specified flowrate (D in m)
f(fF) =  If (Re < 2100) Then (fF - 16 / Re) Else (fF - 1 / (4 * log(Re * sqrt(fF)) - 0.4) ^ 2) # Fanning's friction factor

# The explicit equations
rho = 46.048 + T * (9.418 + T * (-0.0329 + T * (4.882e-5 - T * 2.895e-8)))
vis = exp(-10.547 + 541.69 / (T - 144.53)) # Liquid viscosity (Pa-s)
Re = D * v * rho / vis # Reynold's number
v = q / (pi * D ^ 2 / 4) # Flow velocity (m/s)
dp = 103000 # Pressure drop (Pa)
L = 100 # Pipe length (m)
T = 25 + 273.15 # Temperature (K)
pi = 3.1416
q = 0.0025 # Flow rate (cu-m/s)

# Initial Guess for nonlinear equations variables
D(0) = 0.04
fF(0) = 0.004
```

Ln 18 | NonLinearEquation02.pol | Calculation of Pipe Diameter

The nonlinear equations for pressure drop and for Fanning friction factor will be solved to be zero.

Problem 3 – Pressure Drop Calculation for Pipe Flow Two Nonlinear Equations – Simultaneous Solution with If… Then… Else… Statement



safenewt

Nonlinear Equations: 2

Auxiliary Equations: 9

✓ Ready for solution

```
# Calculation of pipe diameter to have Pressure Drop of 103
f(D) = dp - 2 * fF * rho * v * v * L / D # Calculation of pipe dia
f(fF) =  If (Re < 2100) Then (fF - 16 / Re) Else (fF - 1 / (4 *
                                                                  or

# The explicit equations
rho = 46.048 + T * (9.418 + T * (-0.0329 + T * (4.882e-5 - T
vis = exp(-10.547 + 541.69 / (T - 144.53)) # Liquid viscosity
Re = D * v * rho / vis # Reynold's number
v
dp
L
T
pi
q

#
D
fF
```

Nonlinear Equations Solver: Enter Explicit Equation

Enter the explicit equation:

vis = exp(-10.547 + 541.69 / (T - 144.53))

Comment:

Liquid viscosity (Pa-s)

Nonlinear Equations Solver: Enter Nonlinear Equation

Enter the nonlinear equation:

f(D) = dp - 2 * fF * rho * v * v * L / D = 0

Enter initial guess for the nonlinear variable:

D(0) = 0.04

Comment:

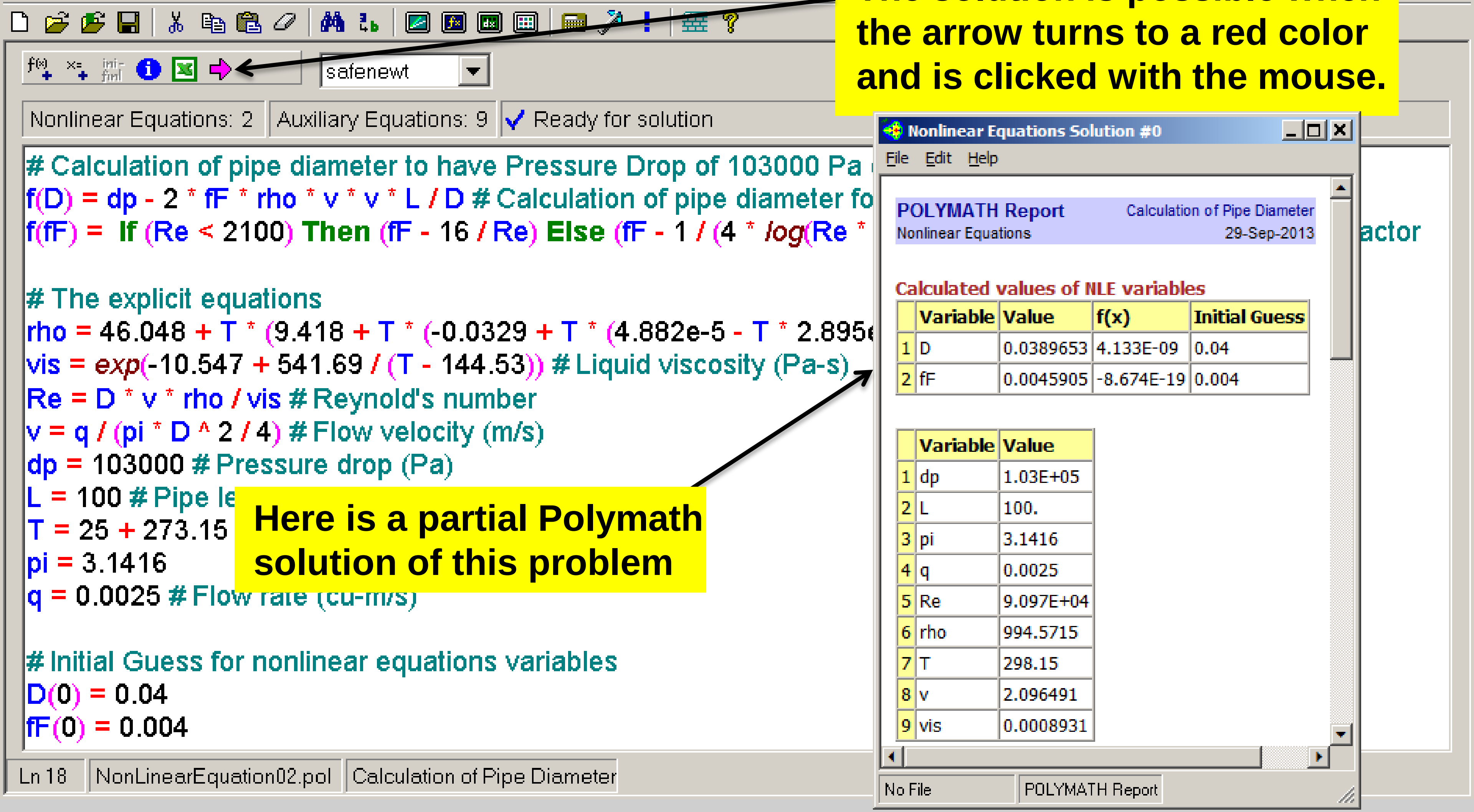
Calculation of pipe diameter for specified flowrate (D in m)

There is a templates for entering an explict equation.

There is a template for entering a nonlinear equation.

Program statements can be entered in any order as Polymath orders them before solution.

Problem 3 – Pressure Drop Calculation for Pipe Flow
Two Nonlinear Equations – Simultaneous Solution with If… Then… Else… Statement



The solution is possible when the arrow turns to a red color and is clicked with the mouse.

Nonlinear Equations: 2 | Auxiliary Equations: 9 | ✓ Ready for solution

```
# Calculation of pipe diameter to have Pressure Drop of 103000 Pa
f(D) = dp - 2 * fF * rho * v * v * L / D # Calculation of pipe diameter fo
f(fF) =  If (Re < 2100) Then (fF - 16 / Re) Else (fF - 1 / (4 * log(Re *                                         actor

# The explicit equations
rho = 46.048 + T * (9.418 + T * (-0.0329 + T * (4.882e-5 - T * 2.895
vis = exp(-10.547 + 541.69 / (T - 144.53)) # Liquid viscosity (Pa-s)
Re = D * v * rho / vis # Reynold's number
v = q / (pi * D ^ 2 / 4) # Flow velocity (m/s)
dp = 103000 # Pressure drop (Pa)
L = 100 # Pipe le
T = 25 + 273.15
pi = 3.1416
q = 0.0025 # Flow rate (cu-m/s)

# Initial Guess for nonlinear equations variables
D(0) = 0.04
fF(0) = 0.004
```

Ln 18 | NonLinearEquation02.pol | Calculation of Pipe Diameter

POLYMATH Report Calculation of Pipe Diameter
Nonlinear Equations 29-Sep-2013

Calculated values of NLE variables

	Variable	Value	f(x)	Initial Guess
1	D	0.0389653	4.133E-09	0.04
2	fF	0.0045905	-8.674E-19	0.004

	Variable	Value
1	dp	1.03E+05
2	L	100.
3	pi	3.1416
4	q	0.0025
5	Re	9.097E+04
6	rho	994.5715
7	T	298.15
8	v	2.096491
9	vis	0.0008931

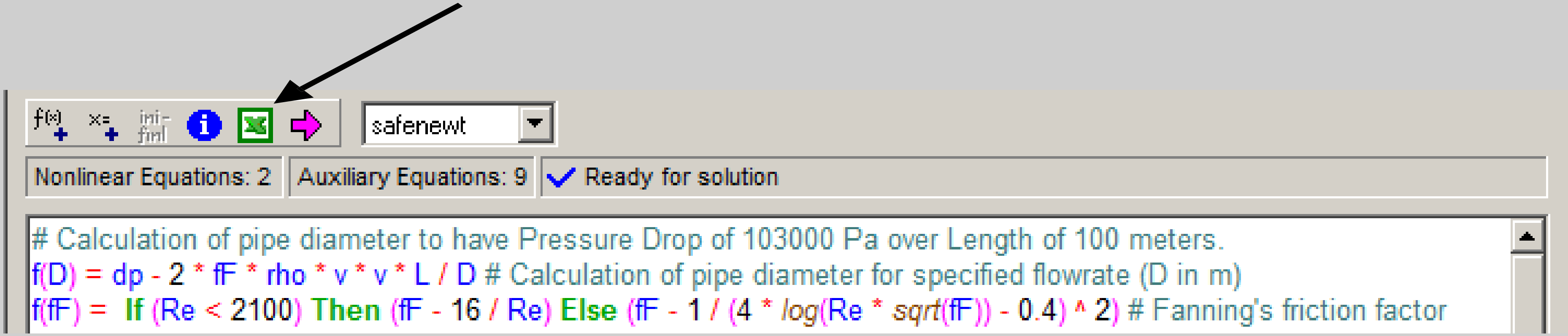
No File | POLYMATH Report

Here is a partial Polymath solution of this problem

Problem 3 – Pressure Drop Calculation for Pipe Flow

Excel Solution for Two Nonlinear Equations – Simultaneous Solution with If… Then… Else… Logic

Polymath Software has the option of automatically sending a problem to Excel by clicking on the Excel icon where the problem is ready to be solved. For Nonlinear Equations, you will use the Solver Add-In to obtain Excel solution. Excel must be open on your computer.



Nonlinear Equations: 2

Auxiliary Equations: 9

✓ Ready for solution

```
# Calculation of pipe diameter to have Pressure Drop of 103000 Pa over Length of 100 meters.
f(D) = dp - 2 * fF * rho * v * v * L / D # Calculation of pipe diameter for specified flowrate (D in m)
f(fF) =  If (Re < 2100) Then (fF - 16 / Re) Else (fF - 1 / (4 * log(Re * sqrt(fF)) - 0.4) ^ 2) # Fanning's friction factor
```

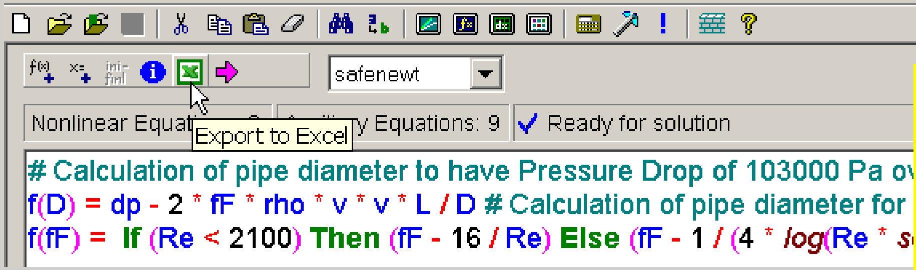
Problem 3 – Pressure Drop Calculation for Pipe Flow

From

Polymath

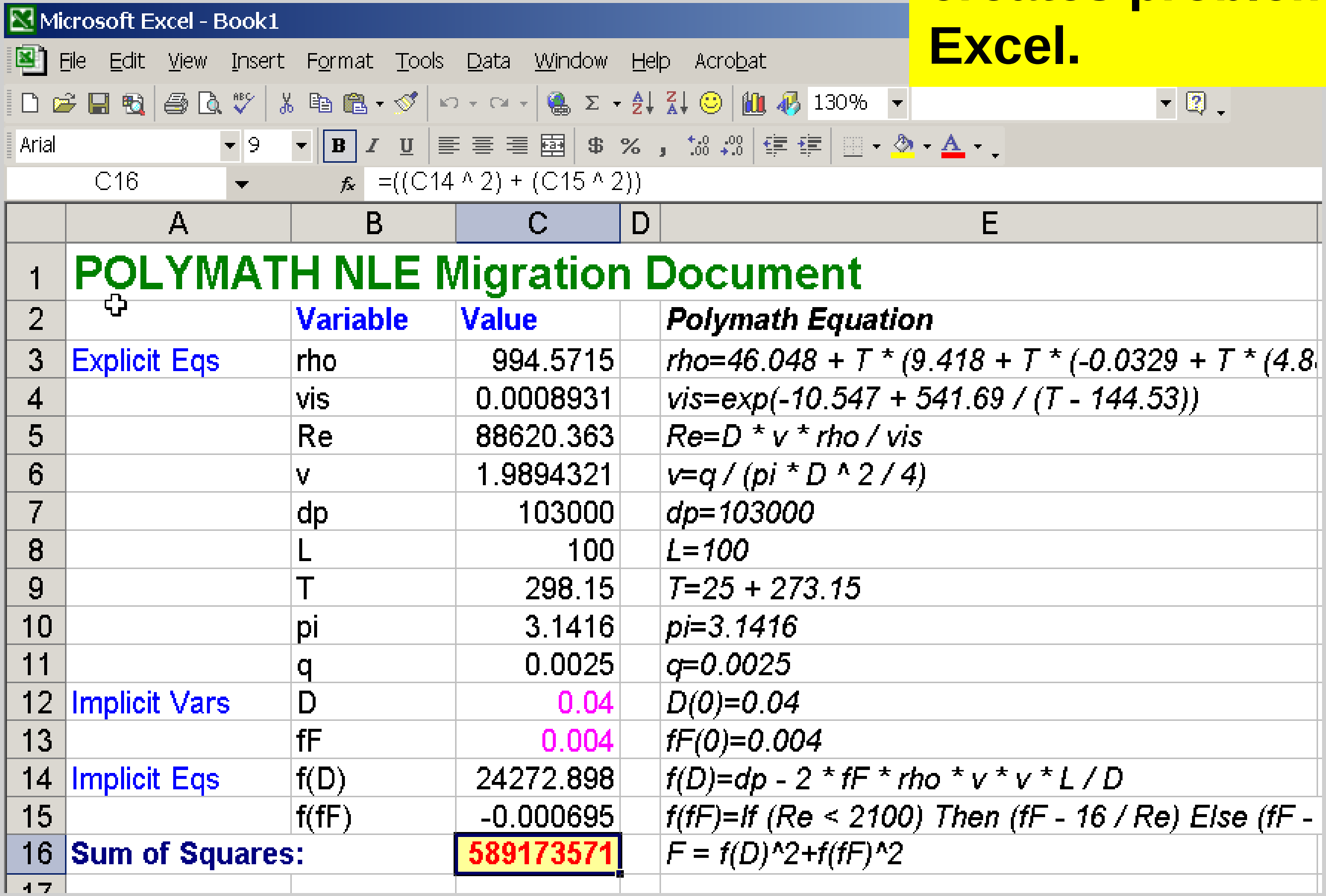
To

Excel



```
# Calculation of pipe diameter to have Pressure Drop of 103000 Pa o
f(D) = dp - 2 * fF * rho * v * v * L / D # Calculation of pipe diameter for
f(fF) =  If (Re < 2100) Then (fF - 16 / Re) Else (fF - 1 / (4 * log(Re * s
```

One Key Press on the Excel icon automatically creates problem in Excel.

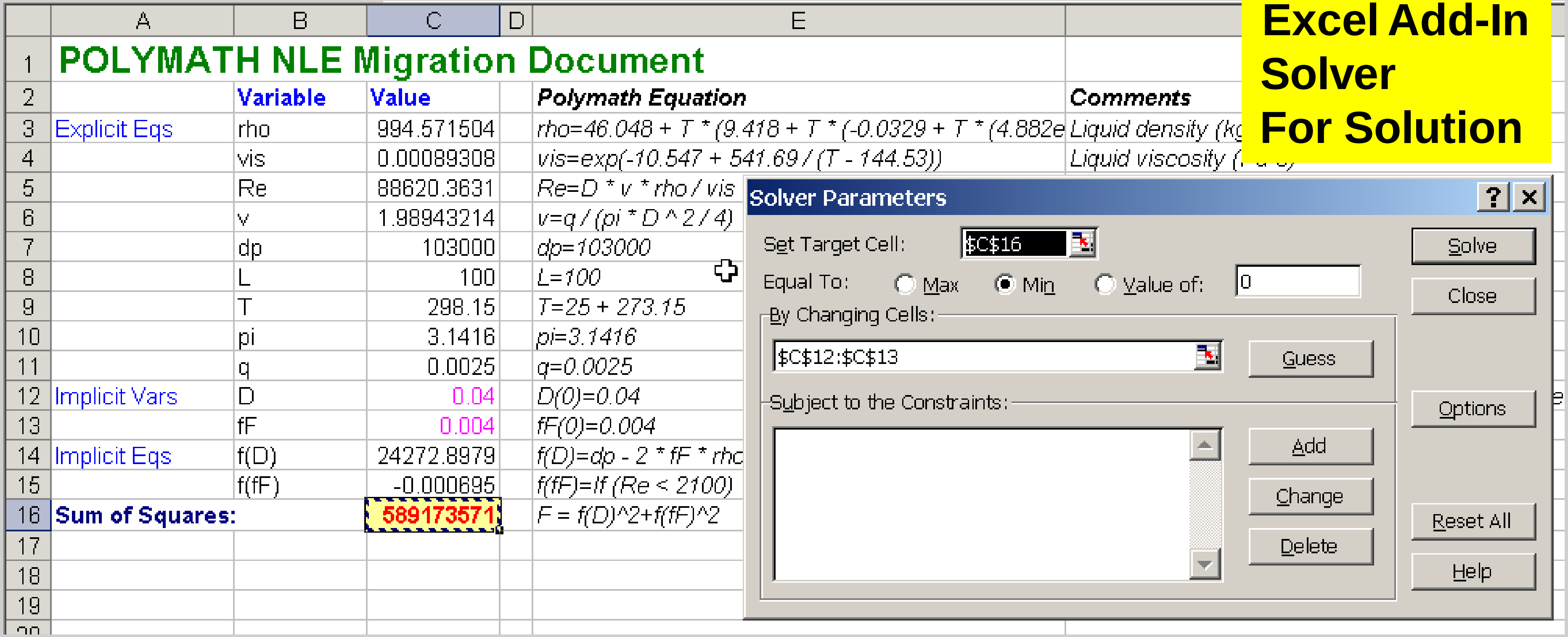


C16 =((C14 ^ 2) + (C15 ^ 2))

	A	B	C	D	E
1	POLYMATH NLE Migration Document				
2		Variable	Value		*Polymath Equation*
3	Explicit Eqs	rho	994.5715		*rho=46.048 + T * (9.418 + T * (-0.0329 + T * (4.8*
4		vis	0.0008931		*vis=exp(-10.547 + 541.69 / (T - 144.53))*
5		Re	88620.363		*Re=D * v * rho / vis*
6		v	1.9894321		*v=q / (pi * D ^ 2 / 4)*
7		dp	103000		*dp=103000*
8		L	100		*L=100*
9		T	298.15		*T=25 + 273.15*
10		pi	3.1416		*pi=3.1416*
11		q	0.0025		*q=0.0025*
12	Implicit Vars	D	0.04		*D(0)=0.04*
13		fF	0.004		*fF(0)=0.004*
14	Implicit Eqs	f(D)	24272.898		*f(D)=dp - 2 * fF * rho * v * v * L / D*
15		f(fF)	-0.000695		*f(fF)=If (Re < 2100) Then (fF - 16 / Re) Else (fF -*
16	**Sum of Squares:**		589173571		F = f(D)^2+f(fF)^2
17					

Two Nonlinear Equations – Simultaneous Solution with If… Then… Else… Statement

Use Excel Add-In Solver For Solution



	A	B	C	D	E	
1	**POLYMATH NLE Migration Document**					
2		**Variable**	**Value**		***Polymath Equation***	***Comments***
3	Explicit Eqs	rho	994.571504		*rho=46.048 + T * (9.418 + T * (-0.0329 + T * (4.882e*	*Liquid density (kg*
4		vis	0.00089308		*vis=exp(-10.547 + 541.69 / (T - 144.53))*	*Liquid viscosity (*
5		Re	88620.3631		*Re=D * v * rho / vis*	
6		v	1.98943214		*v=q / (pi * D ^ 2 / 4)*	
7		dp	103000		*dp=103000*	
8		L	100		*L=100*	
9		T	298.15		*T=25 + 273.15*	
10		pi	3.1416		*pi=3.1416*	
11		q	0.0025		*q=0.0025*	
12	Implicit Vars	D	0.04		*D(0)=0.04*	
13		fF	0.004		*fF(0)=0.004*	
14	Implicit Eqs	f(D)	24272.8979		*f(D)=dp - 2 * fF * rhc*	
15		f(fF)	-0.000695		*f(fF)=If (Re < 2100)*	
16	**Sum of Squares:**		**589173571**		*F = f(D)^2+f(fF)^2*	
17						
18						
19						

Solver Parameters

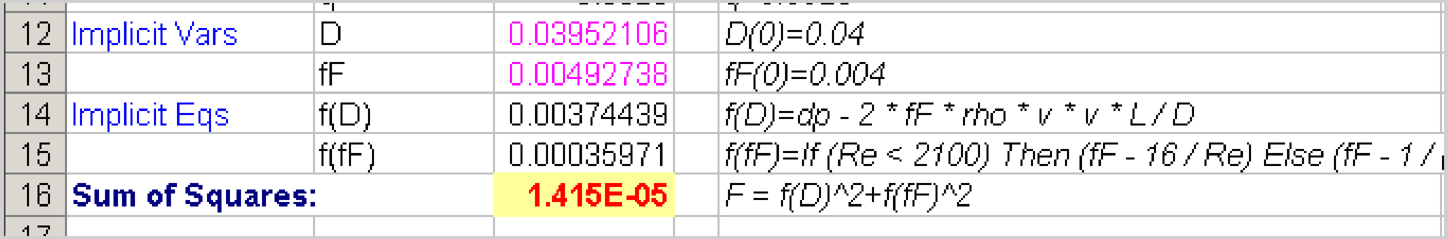
Set Target Cell: $C$16

Equal To: ○ Max ● Min ○ Value of: 0

By Changing Cells: $C$12:$C$13

Subject to the Constraints:

Excel Solution



12	Implicit Vars	D	0.03952106	*D(0)=0.04*
13		fF	0.00492738	*fF(0)=0.004*
14	Implicit Eqs	f(D)	0.00374439	*f(D)=dp - 2 * fF * rho * v * v * L / D*
15		f(fF)	0.00035971	*f(fF)=If (Re < 2100) Then (fF - 16 / Re) Else (fF - 1 /*
16	**Sum of Squares:**		**1.415E-05**	*F = f(D)^2+f(fF)^2*

Note - The Solver display, options, and results vary with the Excel version being used.

Problem 3 – Pressure Drop Calculation for Pipe Flow Two Nonlinear Equations – Simultaneous Solution with If… Then… Else… Statement

Refer to the previously presented Polymath Solution and the Export and Solution of Same Problem in Excel to solve this problem with Polymath and Excel.

First Let's Open an Excel Worksheet

Please open Excel and have Solver Add-In available. This file is Excel.xls

Let's Solve Polymath Problem, Export to Excel, and Solve in Excel

POLYMATH – This file is NonLinearEquations02.pol

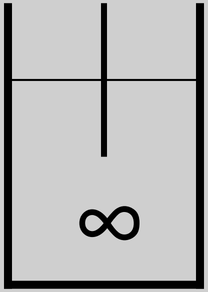
Excel – Ready for Solution file is NonLinearEquations02.xls

Problem 4 – Batch Reactor Kinetics

Differential Equations – Simultaneous ODEs

$$A \xrightarrow{k_1} B \xrightarrow{k_2} C$$

Consider a Batch Reactor that initially has only reactant A



$$\frac{dC_A}{dt} = -k_1 C_A \qquad \text{I. C. } C_A\big|_{t=0} = 1$$

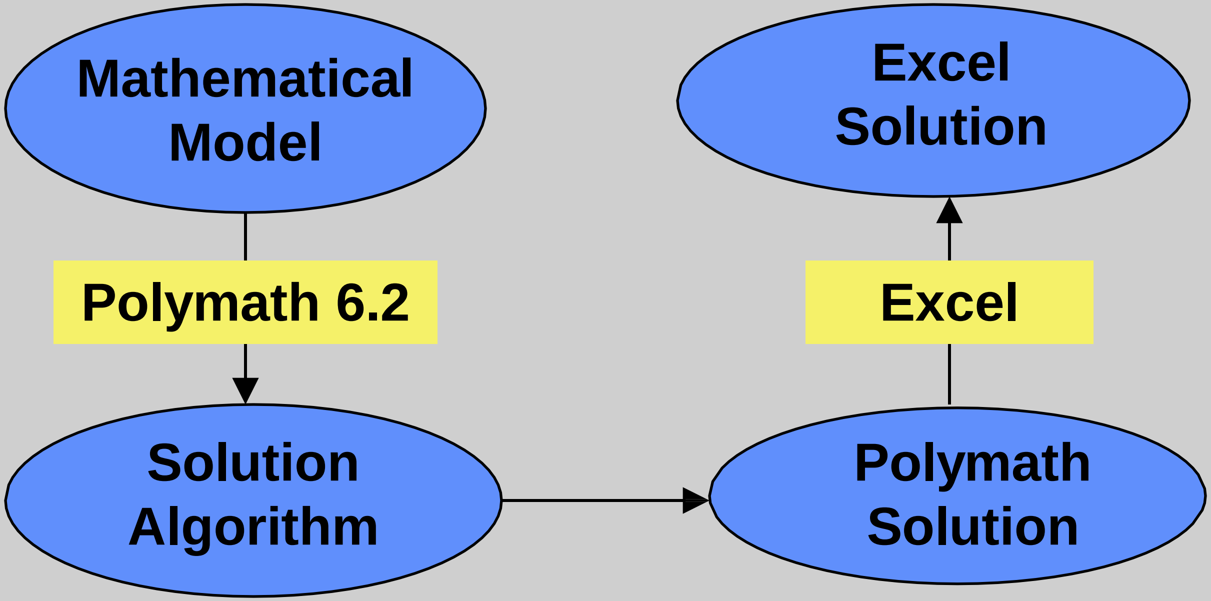
$$\frac{dC_B}{dt} = k_1 C_A - k_2 C_B \qquad \text{I. C. } C_B\big|_{t=0} = 0$$

$$\frac{dC_C}{dt} = k_2 C_B \qquad \text{I. C. } C_C\big|_{t=0} = 0$$

$$k_1 = 2 \qquad k_2 = 3$$

POLYMATH – This file is DifferentialEquations01.pol

Problem 4 – Batch Reactor Kinetics
Differential Equations – Simultaneous ODEs



Mathematical Model

Polymath 6.2

Solution Algorithm

Polymath Solution

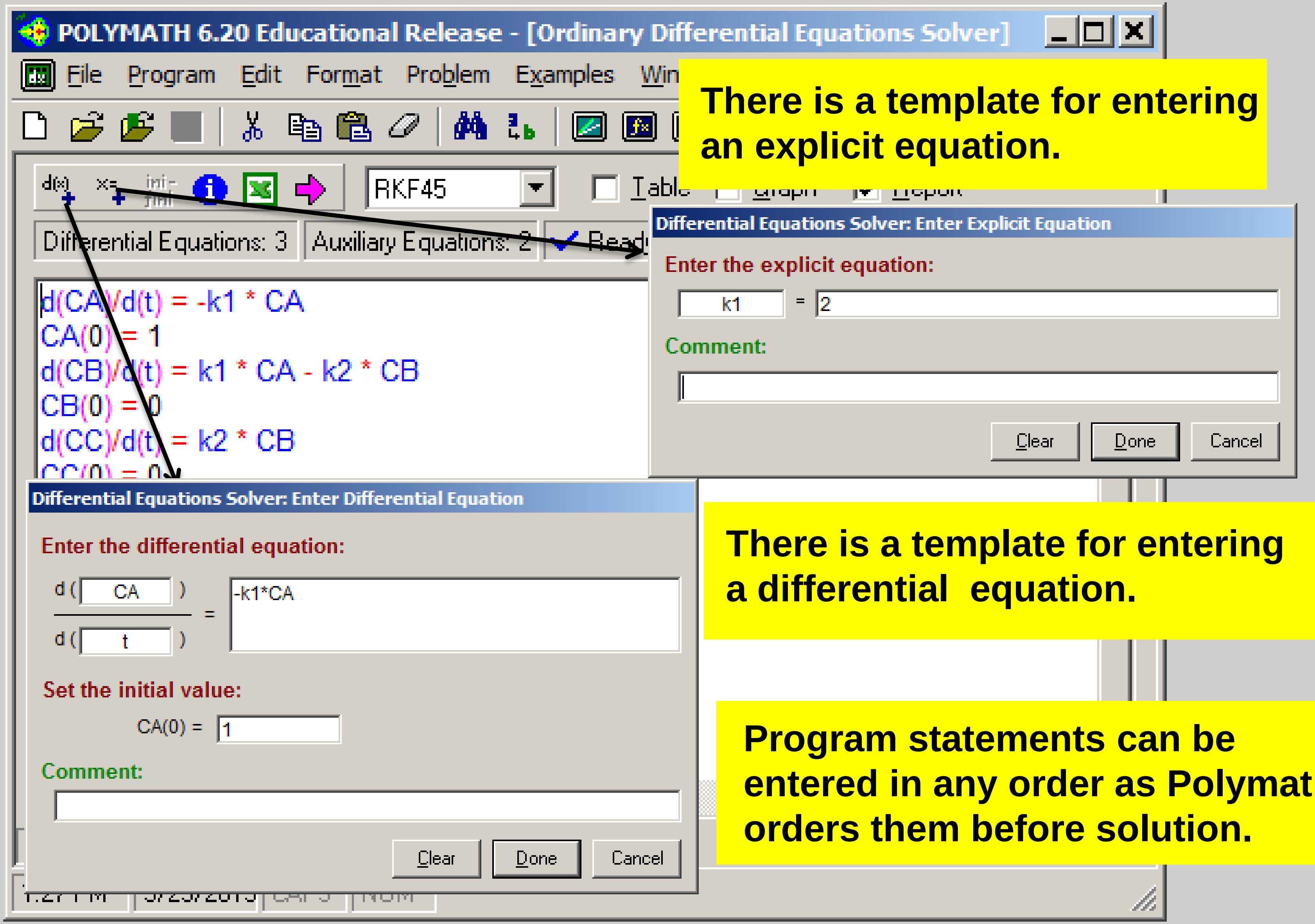
Excel

Excel Solution

Let's Enter and Solve this Problem in POLYMATH

POLYMATH – The solution file is DifferentialEquation1.pol. Please check for the Graph and the Report to be given during the solution.

Problem 4 – Batch Reactor Kinetics
Differential Equations – Simultaneous ODEs



```
d(CA)/d(t) = -k1 * CA
CA(0) = 1
d(CB)/d(t) = k1 * CA - k2 * CB
CB(0) = 0
d(CC)/d(t) = k2 * CB
CC(0) = 0
```

Differential Equations Solver: Enter Explicit Equation

Enter the explicit equation:

k1 = 2

Comment:

Differential Equations Solver: Enter Differential Equation

Enter the differential equation:

d(CA) / d(t) = -k1*CA

Set the initial value:

CA(0) = 1

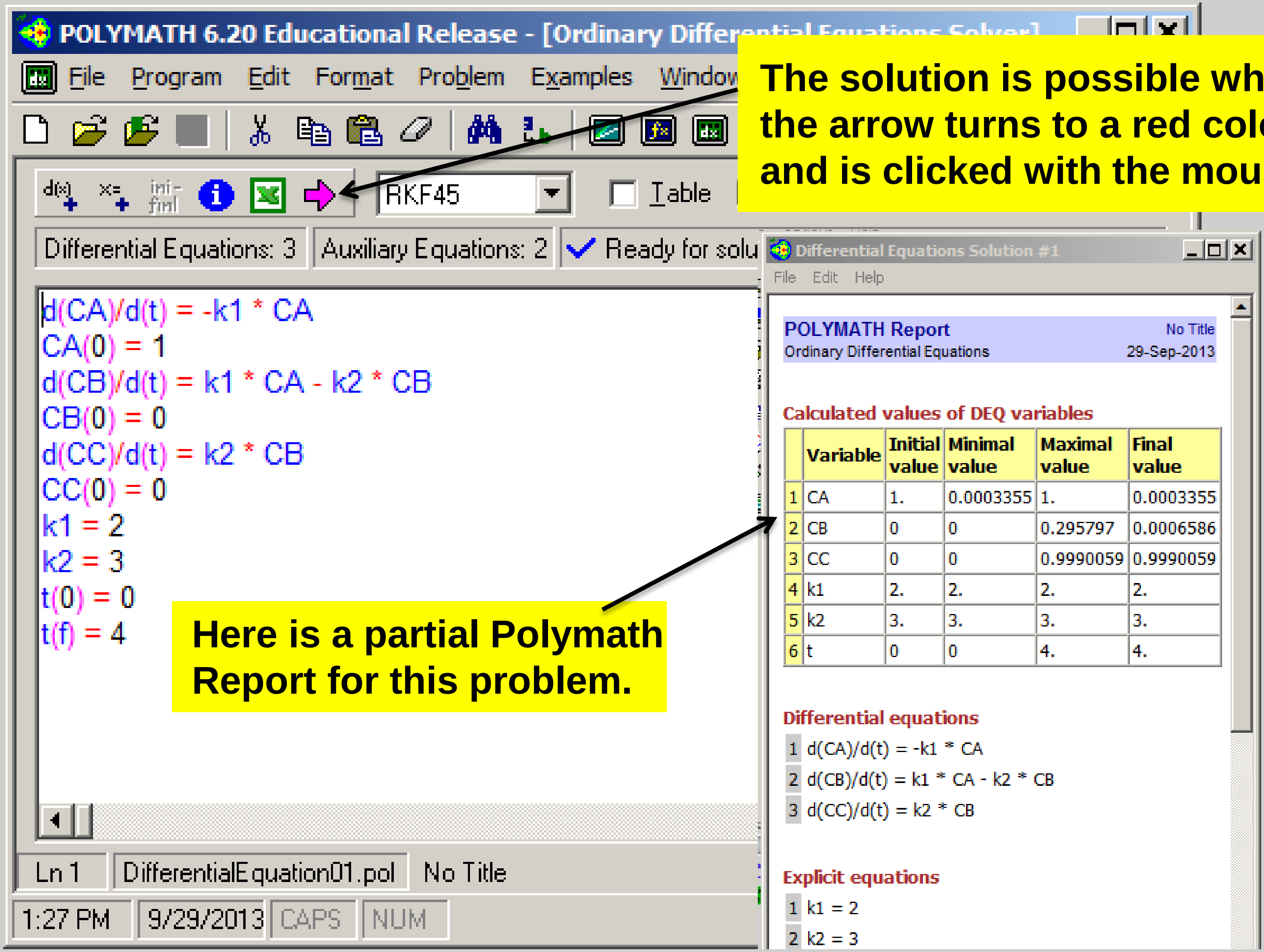
Comment:

There is a template for entering an explicit equation.

There is a template for entering a differential equation.

Program statements can be entered in any order as Polymath orders them before solution.

Problem 4 – Batch Reactor Kinetics
Differential Equations – Simultaneous ODEs



```
d(CA)/d(t) = -k1 * CA
CA(0) = 1
d(CB)/d(t) = k1 * CA - k2 * CB
CB(0) = 0
d(CC)/d(t) = k2 * CB
CC(0) = 0
k1 = 2
k2 = 3
t(0) = 0
t(f) = 4
```

POLYMATH Report No Title

Ordinary Differential Equations 29-Sep-2013

Calculated values of DEQ variables

	Variable	Initial value	Minimal value	Maximal value	Final value
1	CA	1.	0.0003355	1.	0.0003355
2	CB	0	0	0.295797	0.0006586
3	CC	0	0	0.9990059	0.9990059
4	k1	2.	2.	2.	2.
5	k2	3.	3.	3.	3.
6	t	0	0	4.	4.

Differential equations

1 d(CA)/d(t) = -k1 * CA
2 d(CB)/d(t) = k1 * CA - k2 * CB
3 d(CC)/d(t) = k2 * CB

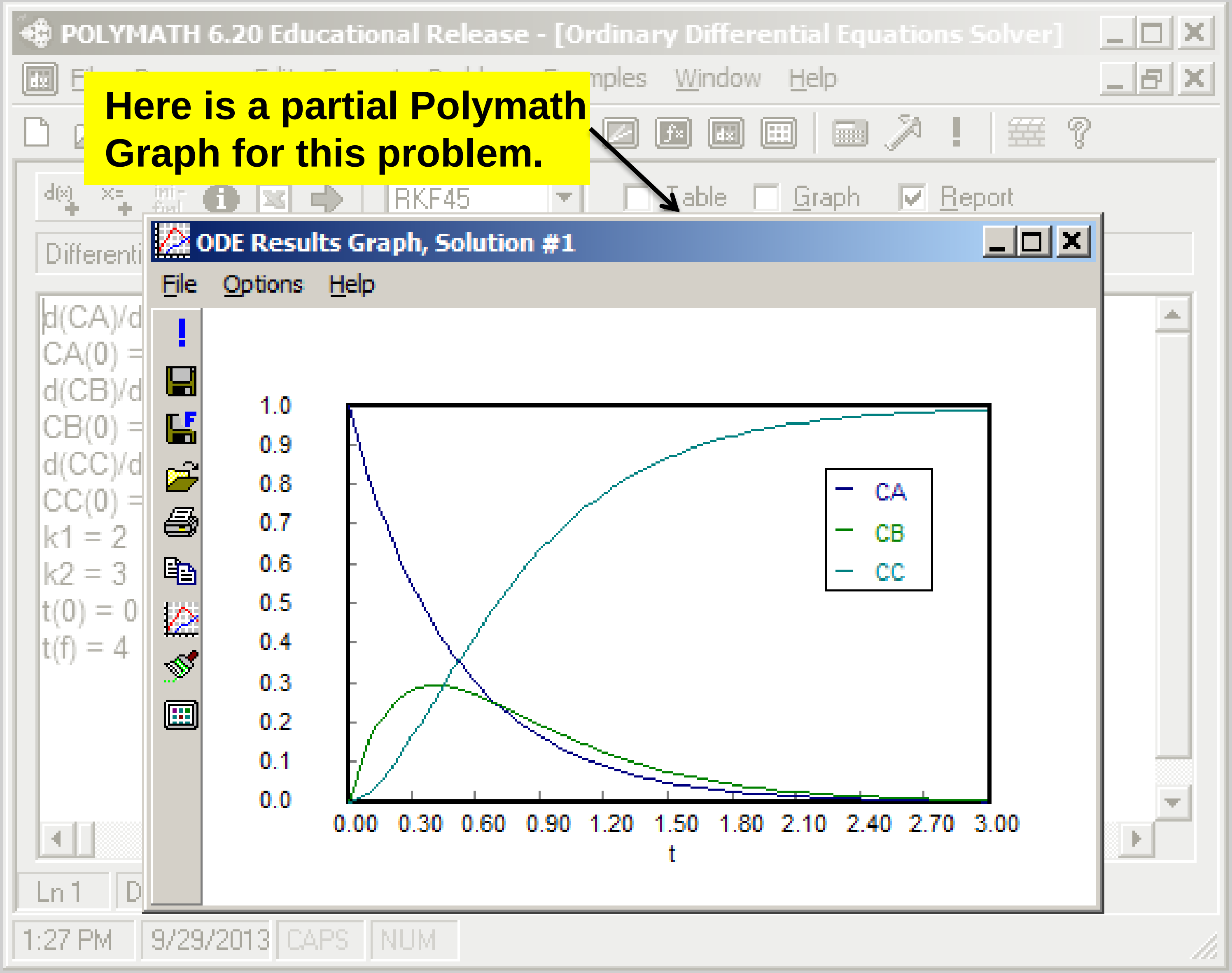
Explicit equations

1 k1 = 2
2 k2 = 3

The solution is possible when the arrow turns to a red color and is clicked with the mouse.

Here is a partial Polymath Report for this problem.

Problem 4 – Batch Reactor Kinetics Differential Equations – Simultaneous ODEs

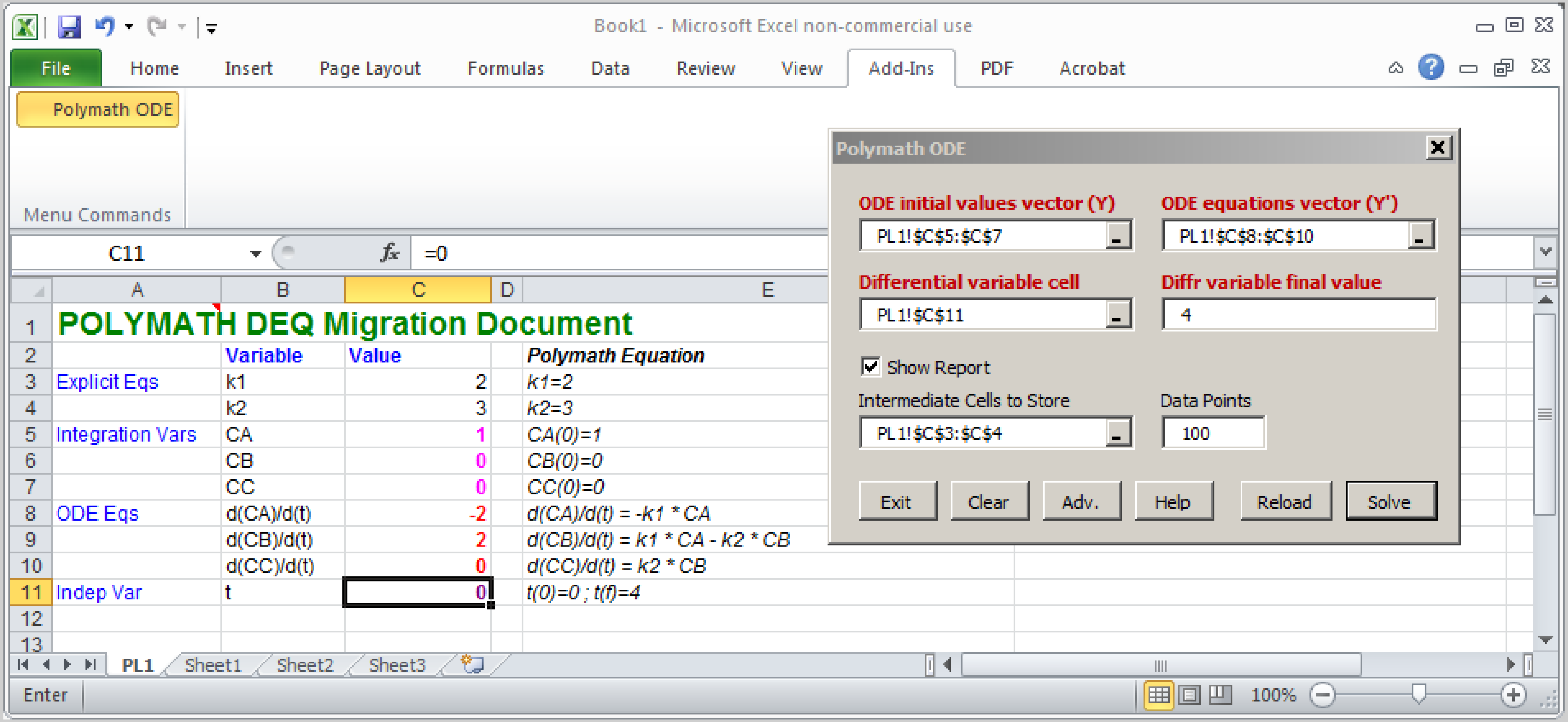


Here is a partial Polymath Graph for this problem.

d(CA)/d
CA(0) =
d(CB)/d
CB(0) =
d(CC)/d
CC(0) =
k1 = 2
k2 = 3
t(0) = 0
t(f) = 4

Problem 4 – Batch Reactor Kinetics
Differential Equations – Simultaneous ODEs

Let's Open Excel and Export Problem to Excel by pressing the Excel icon. Polymath ODE_Solver should be available on Add-Ins sheet at top left and press to bring up Polymath ODE control box.



C11 f_x =0

	A	B	C	D	E
1	POLYMATH DEQ Migration Document				
2		Variable	Value		*Polymath Equation*
3	Explicit Eqs	k1	2		*k1=2*
4		k2	3		*k2=3*
5	Integration Vars	CA	1		*CA(0)=1*
6		CB	0		*CB(0)=0*
7		CC	0		*CC(0)=0*
8	ODE Eqs	d(CA)/d(t)	-2		*d(CA)/d(t) = -k1 * CA*
9		d(CB)/d(t)	2		*d(CB)/d(t) = k1 * CA - k2 * CB*
10		d(CC)/d(t)	0		*d(CC)/d(t) = k2 * CB*
11	Indep Var	t	0		*t(0)=0 ; t(f)=4*
12					
13					

PL1 | Sheet1 | Sheet2 | Sheet3

Polymath ODE

ODE initial values vector (Y): PL1!$C$5:$C$7

ODE equations vector (Y'): PL1!$C$8:$C$10

Differential variable cell: PL1!$C$11

Diffr variable final value: 4

☑ Show Report

Intermediate Cells to Store: PL1!$C$3:$C$4

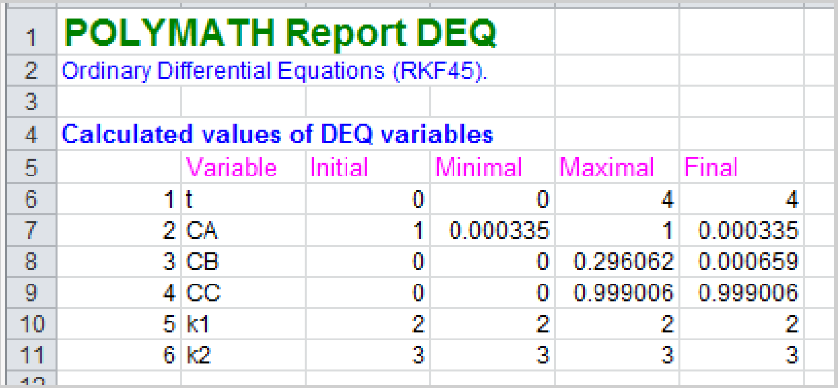
Data Points: 100

Exit | Clear | Adv. | Help | Reload | Solve

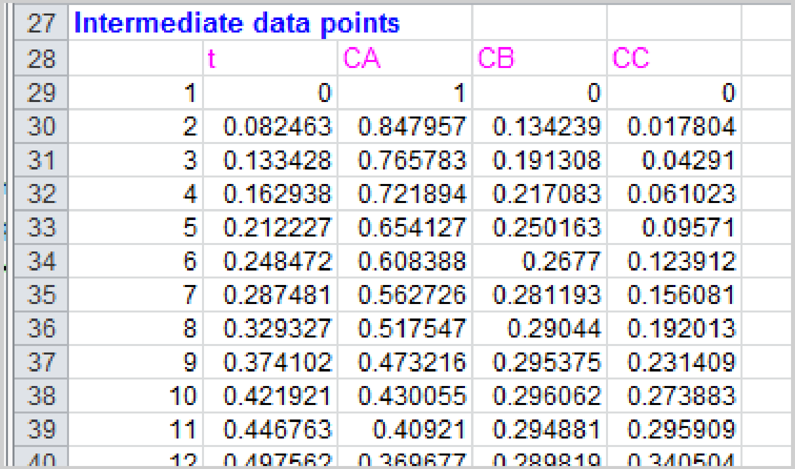
POLYMATH – The solution file is DifferentialEquation01.pol. Open Excel before export from Polymath or open DifferentialEquation01.xls for solution.

Problem 4 – Batch Reactor Kinetics Differential Equations – Simultaneous ODEs

Setup solution in Polymath and export to Excel. Solve with ODE_Solver AddIn in Excel and compare with Polymath solution.



1	**POLYMATH Report DEQ**					
2	Ordinary Differential Equations (RKF45).					
3						
4	**Calculated values of DEQ variables**					
5		Variable	Initial	Minimal	Maximal	Final
6	1	t	0	0	4	4
7	2	CA	1	0.000335	1	0.000335
8	3	CB	0	0	0.296062	0.000659
9	4	CC	0	0	0.999006	0.999006
10	5	k1	2	2	2	2
11	6	k2	3	3	3	3



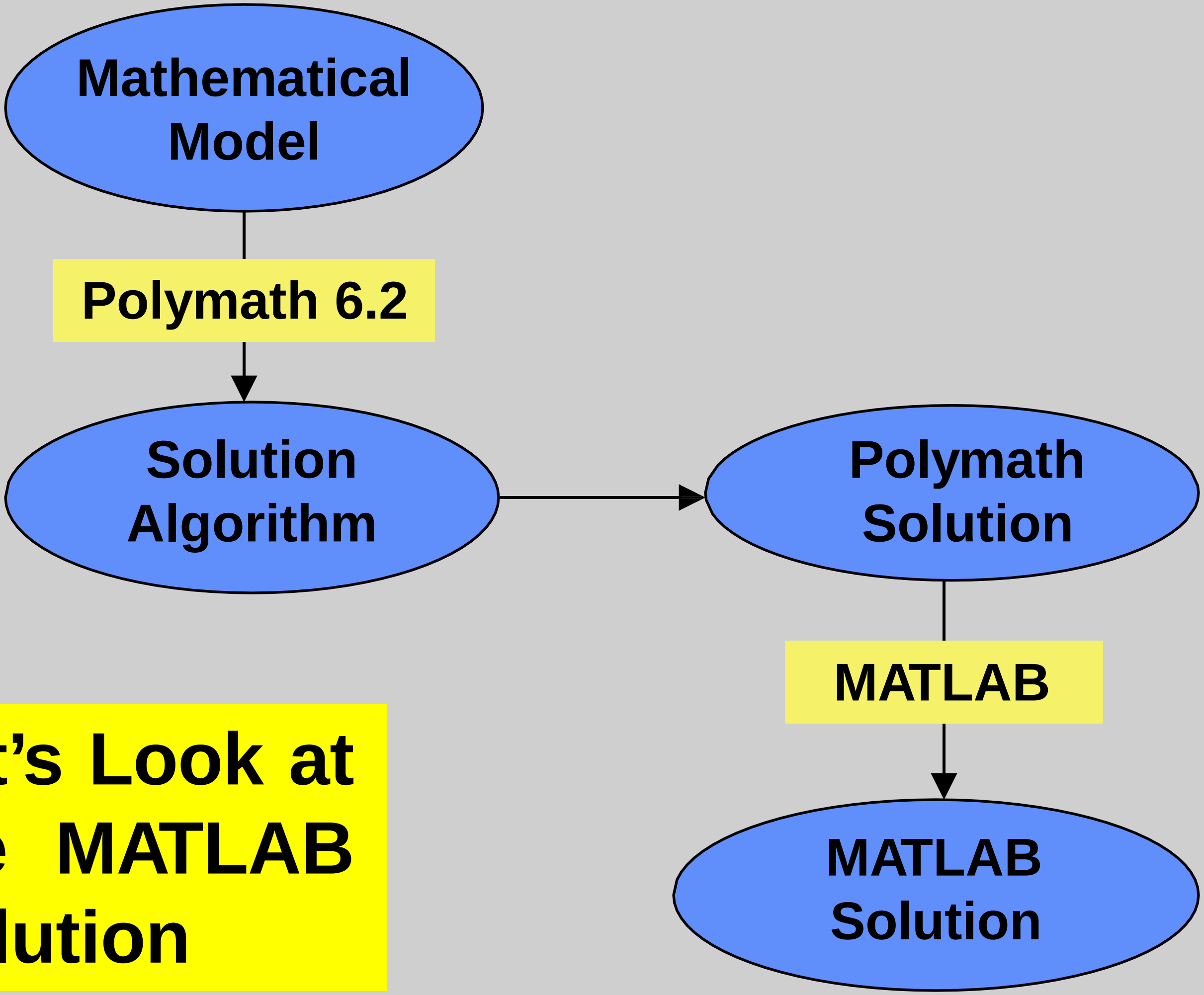
27	**Intermediate data points**				
28		t	CA	CB	CC
29	1	0	1	0	0
30	2	0.082463	0.847957	0.134239	0.017804
31	3	0.133428	0.765783	0.191308	0.04291
32	4	0.162938	0.721894	0.217083	0.061023
33	5	0.212227	0.654127	0.250163	0.09571
34	6	0.248472	0.608388	0.2677	0.123912
35	7	0.287481	0.562726	0.281193	0.156081
36	8	0.329327	0.517547	0.29044	0.192013
37	9	0.374102	0.473216	0.295375	0.231409
38	10	0.421921	0.430055	0.296062	0.273883
39	11	0.446763	0.40921	0.294881	0.295909
40	12	0.497562	0.369677	0.289819	0.340504

The problem variables names can be added to the results that is similar to the Polymath report.

The problem variables can be added to the Intermediate data points and plotted using Excel graphics.

Solution is on DifferentialEquations01.xls on sheet DEQ Solution (1).

Problem 4 – Batch Reactor Kinetics
Differential Equations – Simultaneous ODEs



Mathematical Model

Polymath 6.2

Solution Algorithm

Polymath Solution

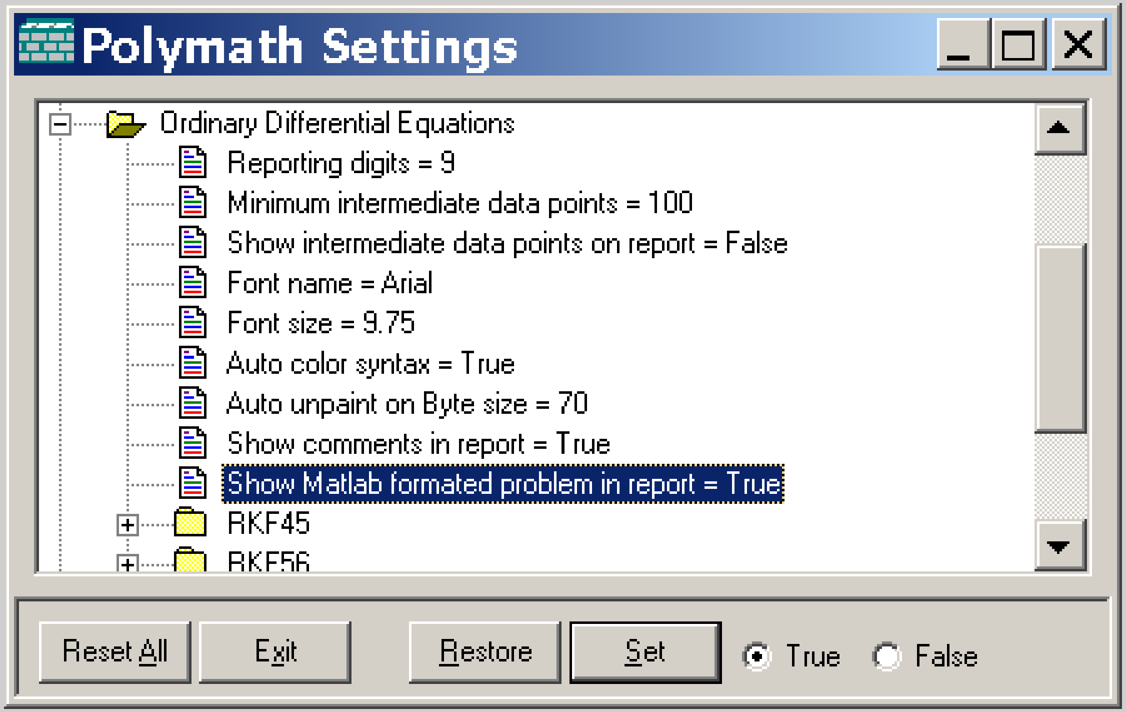
MATLAB

MATLAB Solution

Let's Look at the MATLAB Solution

Problem 4 – Batch Reactor Kinetics
Differential Equations – Simultaneous ODEs

MATLAB problem solution is obtained by first requesting MATLAB output in the Polymath Setting window found with the Settings Icon.



Ordinary Differential Equations
- Reporting digits = 9
- Minimum intermediate data points = 100
- Show intermediate data points on report = False
- Font name = Arial
- Font size = 9.75
- Auto color syntax = True
- Auto unpaint on Byte size = 70
- Show comments in report = True
- Show Matlab formated problem in report = True

RKF45

RKF56

Problem 4 – Batch Reactor Kinetics
Differential Equations – Simultaneous ODEs

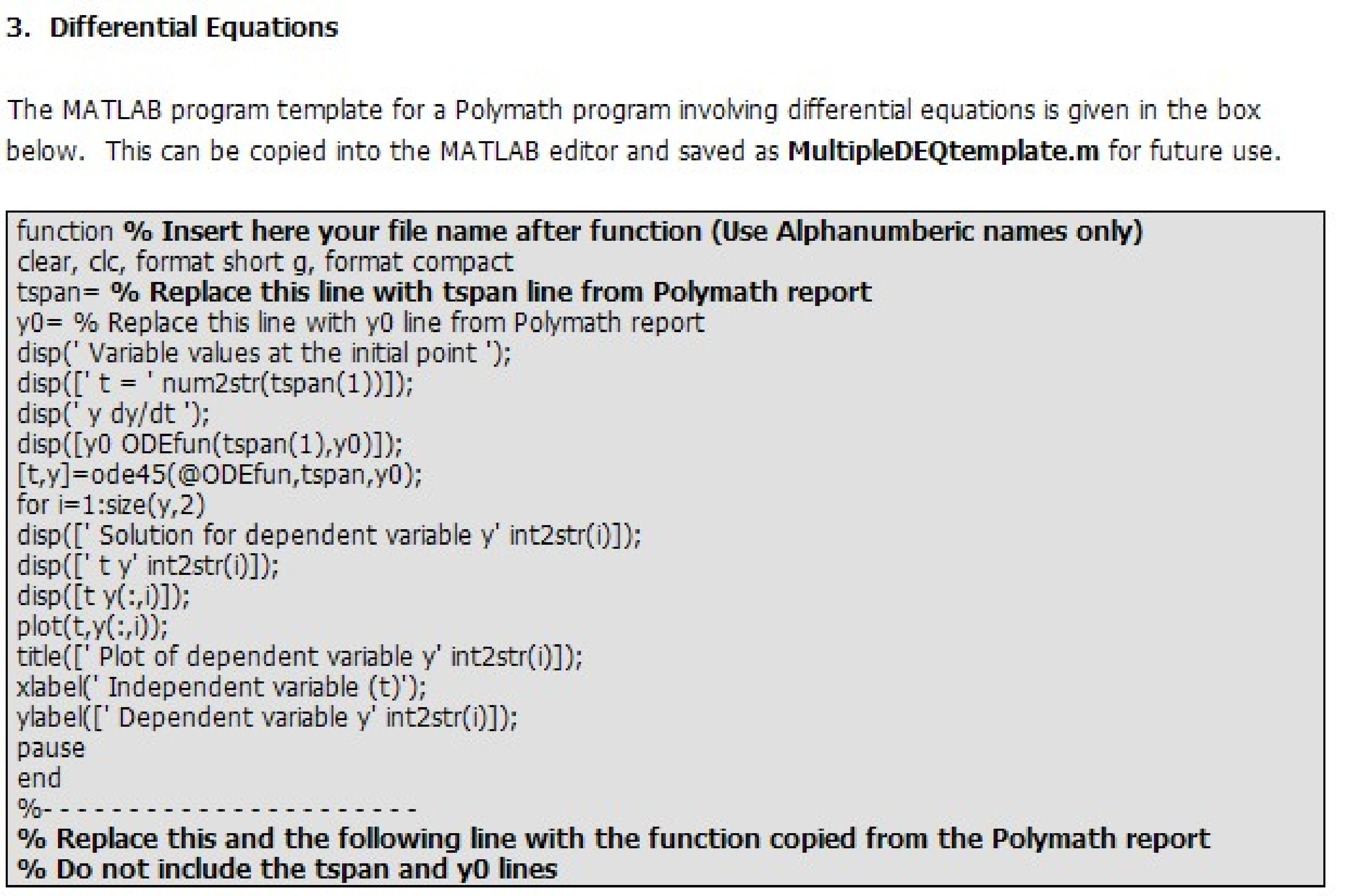
This option for MATLAB formatted output results in the MATLAB code to be generated automatically at the end of the POLYMATH report.

Matlab formatted problem

```
tspan = [0 4.]; % Range for the independent variable
y0 = [1.; 0; 0]; % Initial values for the dependent
variables
function dYfuncvecdt = ODEfun(t,Yfuncvec);
CA = Yfuncvec(1);
CB = Yfuncvec(2);
CC = Yfuncvec(3);
k1 = 2;
k2 = 3;
dCAdt = 0 - (k1 * CA);
dCBdt = k1 * CA - (k2 * CB);
dCCdt = k2 * CB;
dYfuncvecdt = [dCAdt; dCBdt; dCCdt];
```

Differential Equations – Simultaneous ODEs

The MATLAB formatted output is copied and pasted into the MATLAB template that is provided within the Polymath HELP materials.

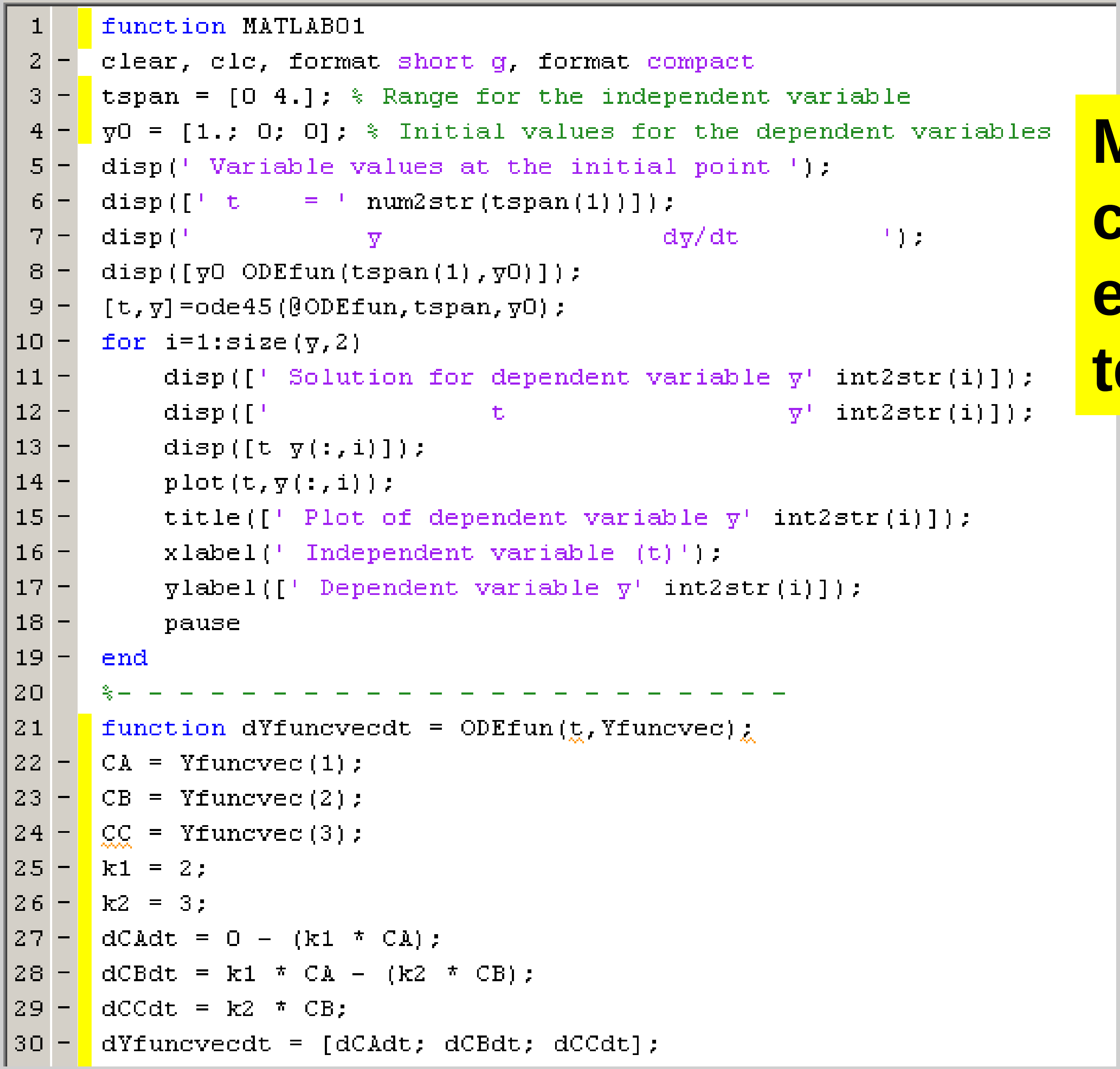


3. Differential Equations

The MATLAB program template for a Polymath program involving differential equations is given in the box below. This can be copied into the MATLAB editor and saved as **MultipleDEQtemplate.m** for future use.

```
function % Insert here your file name after function (Use Alphanumberic names only)
clear, clc, format short g, format compact
tspan= % Replace this line with tspan line from Polymath report
y0= % Replace this line with y0 line from Polymath report
disp(' Variable values at the initial point ');
disp([' t = ' num2str(tspan(1))]);
disp(' y dy/dt ');
disp([y0 ODEfun(tspan(1),y0)]);
[t,y]=ode45(@ODEfun,tspan,y0);
for i=1:size(y,2)
disp([' Solution for dependent variable y' int2str(i)]);
disp([' t y' int2str(i)]);
disp([t y(:,i)]);
plot(t,y(:,i));
title([' Plot of dependent variable y' int2str(i)]);
xlabel(' Independent variable (t)');
ylabel([' Dependent variable y' int2str(i)]);
pause
end
%- - - - - - - - - - - - - - - - - - - - - -
% Replace this and the following line with the function copied from the Polymath report
% Do not include the tspan and y0 lines
```

Differential Equations – Simultaneous ODEs

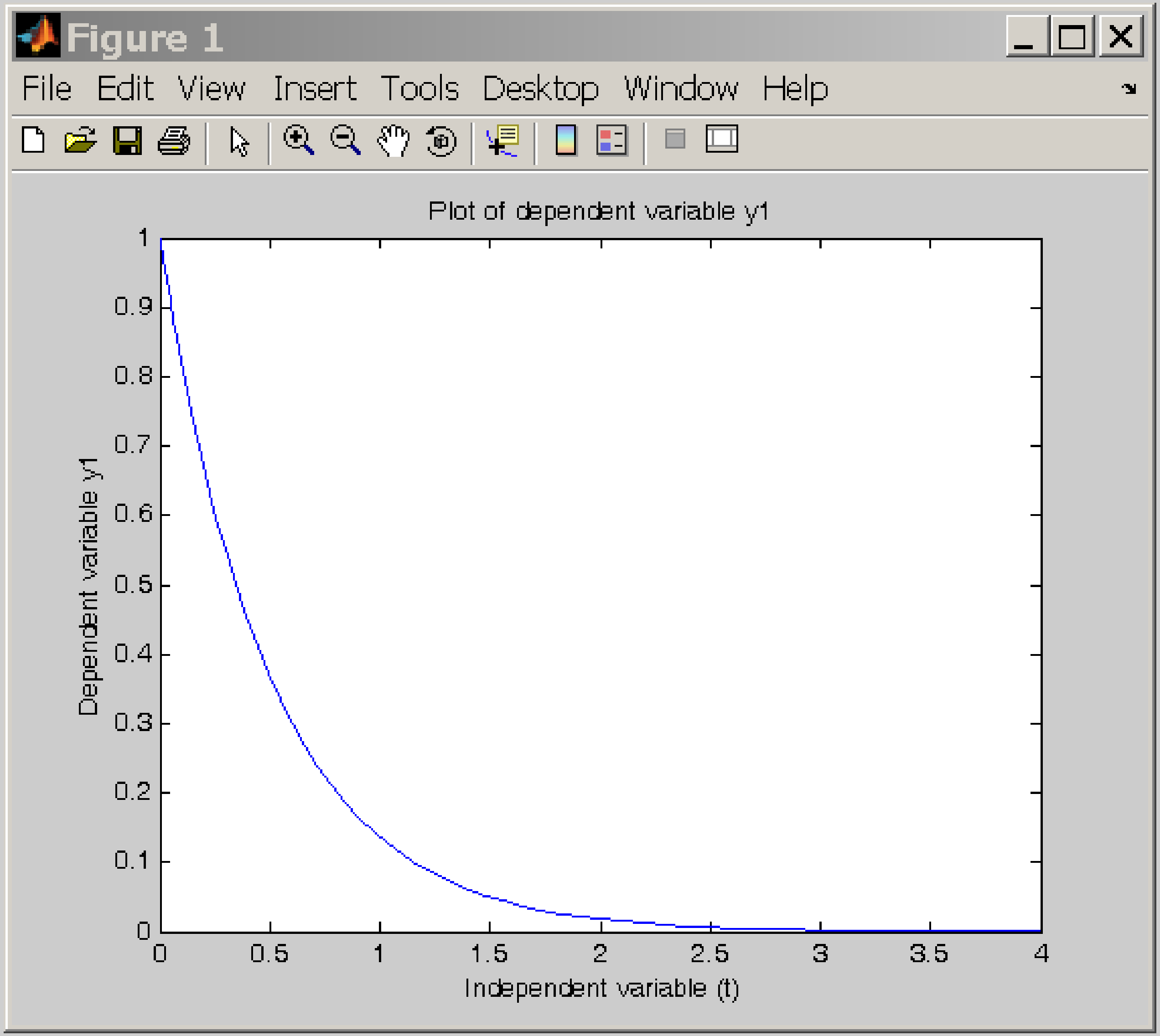


```
function MATLAB01
clear, clc, format short g, format compact
tspan = [0 4.]; % Range for the independent variable
y0 = [1.; 0; 0]; % Initial values for the dependent variables
disp(' Variable values at the initial point ');
disp([' t    = ' num2str(tspan(1))]);
disp('           y                  dy/dt         ');
disp([y0 ODEfun(tspan(1),y0)]);
[t,y]=ode45(@ODEfun,tspan,y0);
for i=1:size(y,2)
    disp([' Solution for dependent variable y' int2str(i)]);
    disp(['              t                  y' int2str(i)]);
    disp([t y(:,i)]);
    plot(t,y(:,i));
    title([' Plot of dependent variable y' int2str(i)]);
    xlabel(' Independent variable (t)');
    ylabel([' Dependent variable y' int2str(i)]);
    pause
end
%- - - - - - - - - - - - - - - - - - - - - - -
function dYfuncvecdt = ODEfun(t,Yfuncvec);
CA = Yfuncvec(1);
CB = Yfuncvec(2);
CC = Yfuncvec(3);
k1 = 2;
k2 = 3;
dCAdt = 0 - (k1 * CA);
dCBdt = k1 * CA - (k2 * CB);
dCCdt = k2 * CB;
dYfuncvecdt = [dCAdt; dCBdt; dCCdt];
```

MATLAB code is entered into template

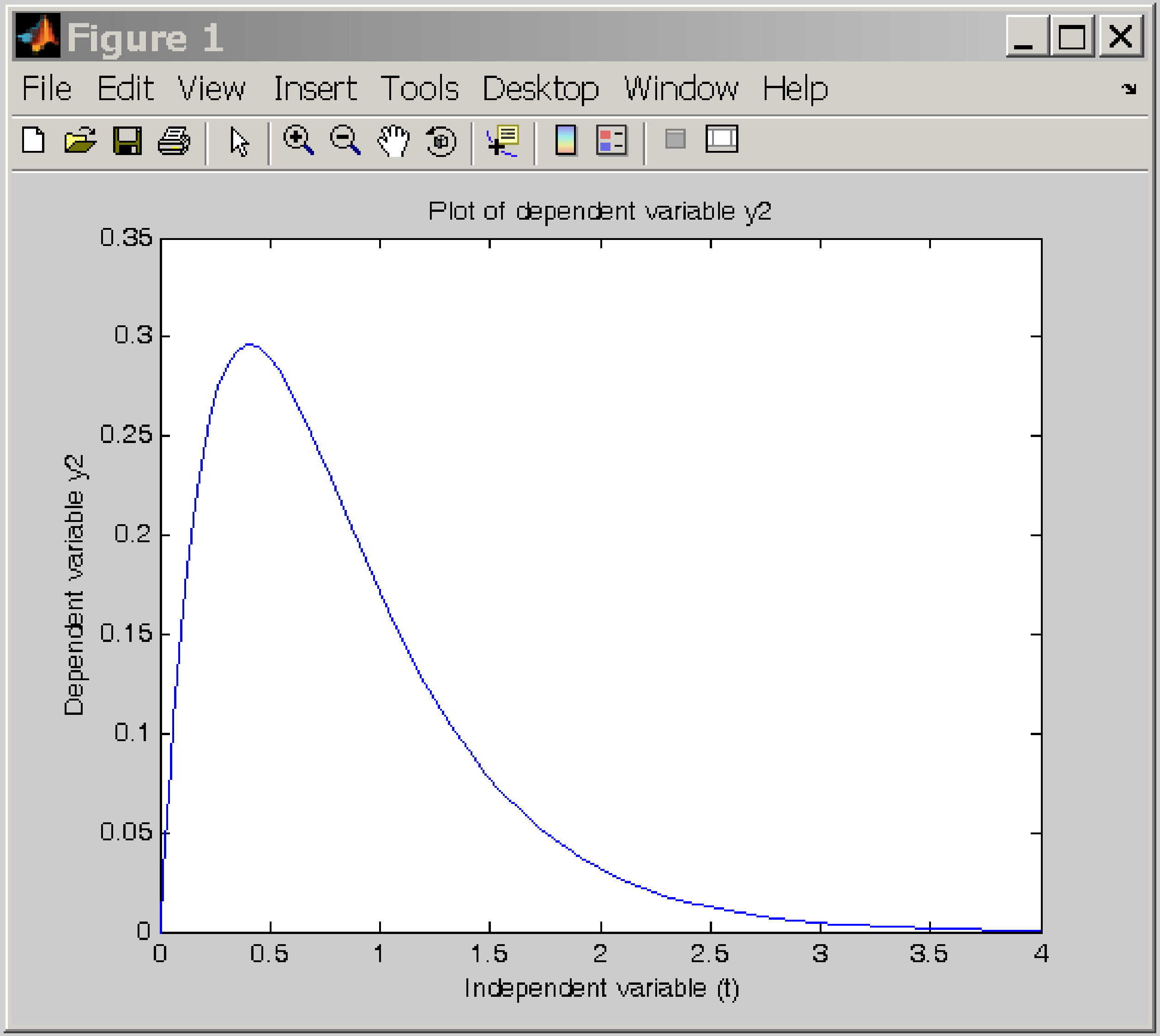
Differential Equations – Simultaneous ODEs

The MATLAB m-file thus created provides graphical output for all differential variables.



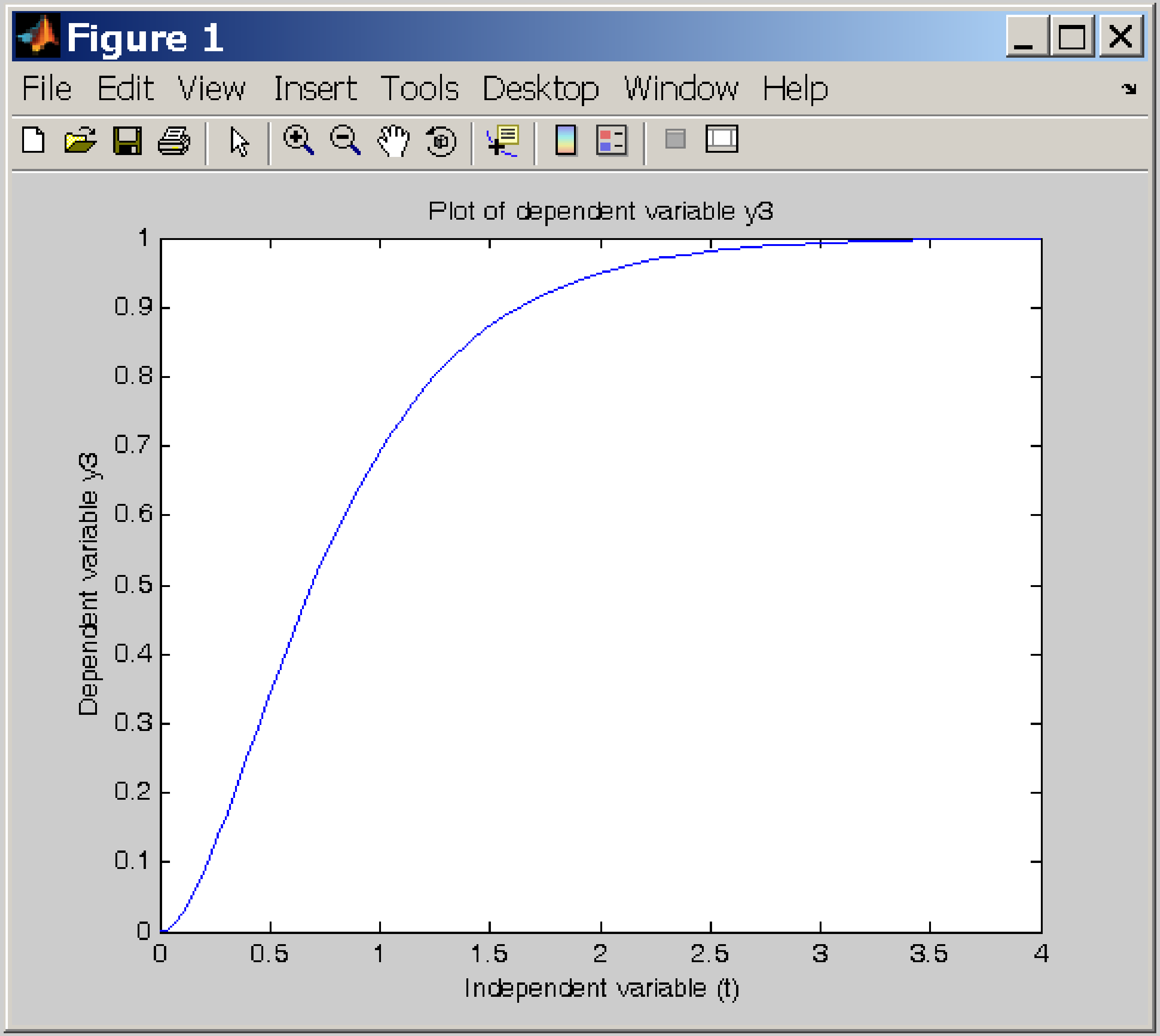
Differential Equations – Simultaneous ODEs

The MATLAB m-file thus created provides graphical output for all differential variables.



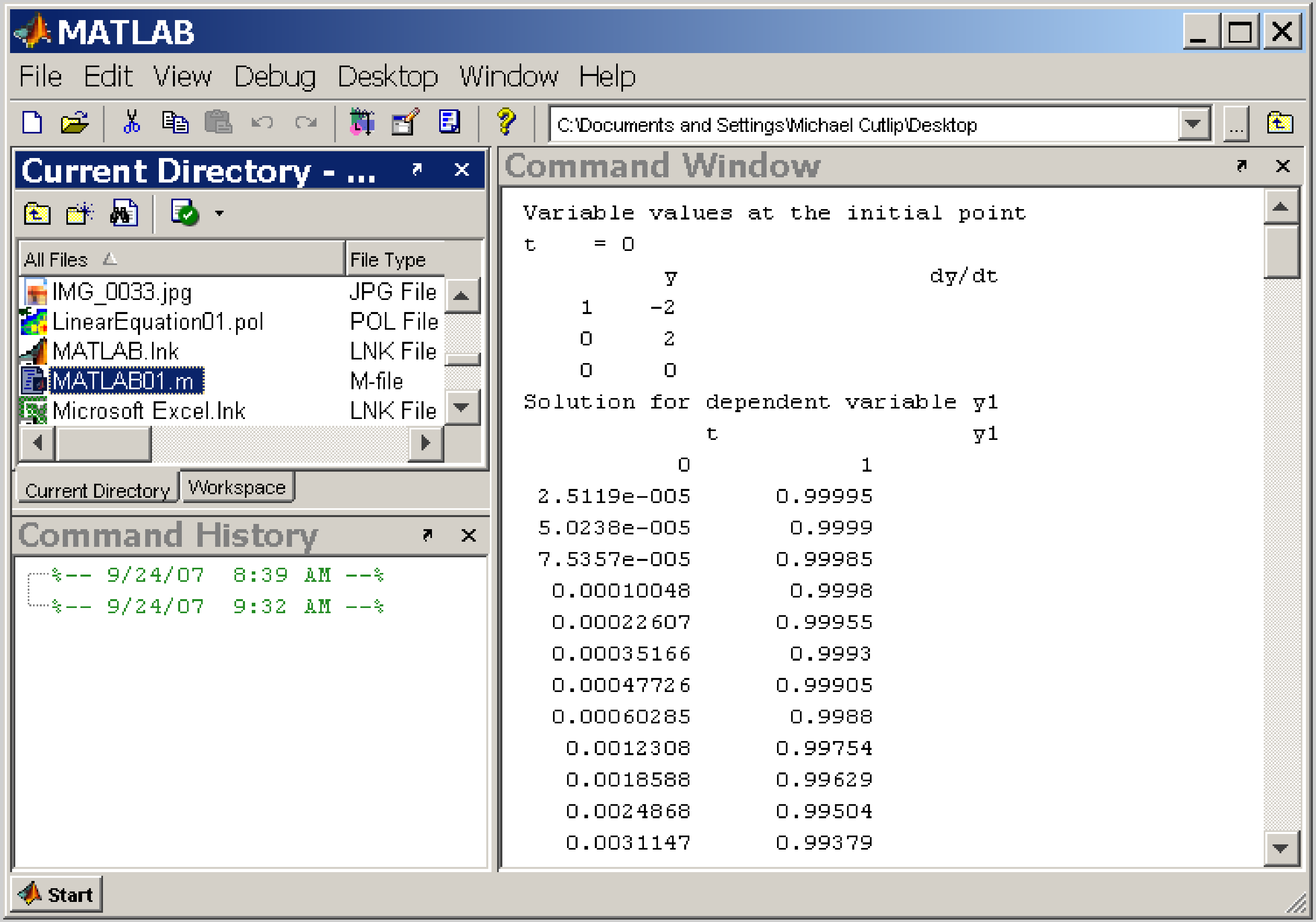
Differential Equations – Simultaneous ODEs

The MATLAB m-file thus created provides graphical output for all differential variables.



Differential Equations – Simultaneous ODEs

The MATLAB m-file thus created also provides tabular output within the MATLAB editor.



```
%-- 9/24/07  8:39 AM --%
%-- 9/24/07  9:32 AM --%
```

```
Variable values at the initial point
t     = 0
          y                  dy/dt
    1     -2
    0      2
    0      0
Solution for dependent variable y1
             t                    y1
             0             1
  2.5119e-005       0.99995
  5.0238e-005        0.9999
  7.5357e-005       0.99985
   0.00010048        0.9998
   0.00022607       0.99955
   0.00035166        0.9993
   0.00047726       0.99905
   0.00060285        0.9988
    0.0012308       0.99754
    0.0018588       0.99629
    0.0024868       0.99504
    0.0031147       0.99379
```

Problem 4 – Batch Reactor Kinetics
Differential Equations – Simultaneous ODEs

This optional demonstration requires the use of MATLAB program on your PC.

Let’s Go to the POLYMATH Problem, Solve the Problem, and Generate MATLAB Code!

POLYMATH – This file is DifferentialEquation01.pol

Let’s Open MATLAB with Template for Multiple Differential Equations, Insert Generated Code, and Solve in MATLAB

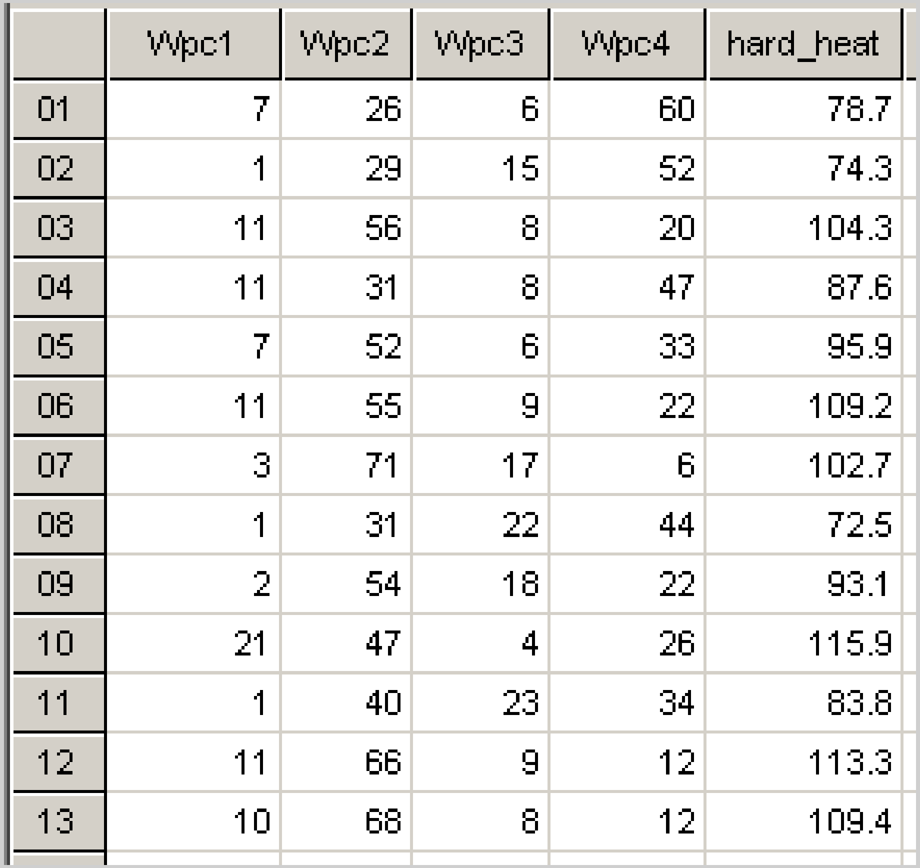
Template File is MultipleDEQtemplate.m

Solution File is MATLAB01.m

Problem 5 – Regression of Hardening of Concrete

Regression – Multiple Linear

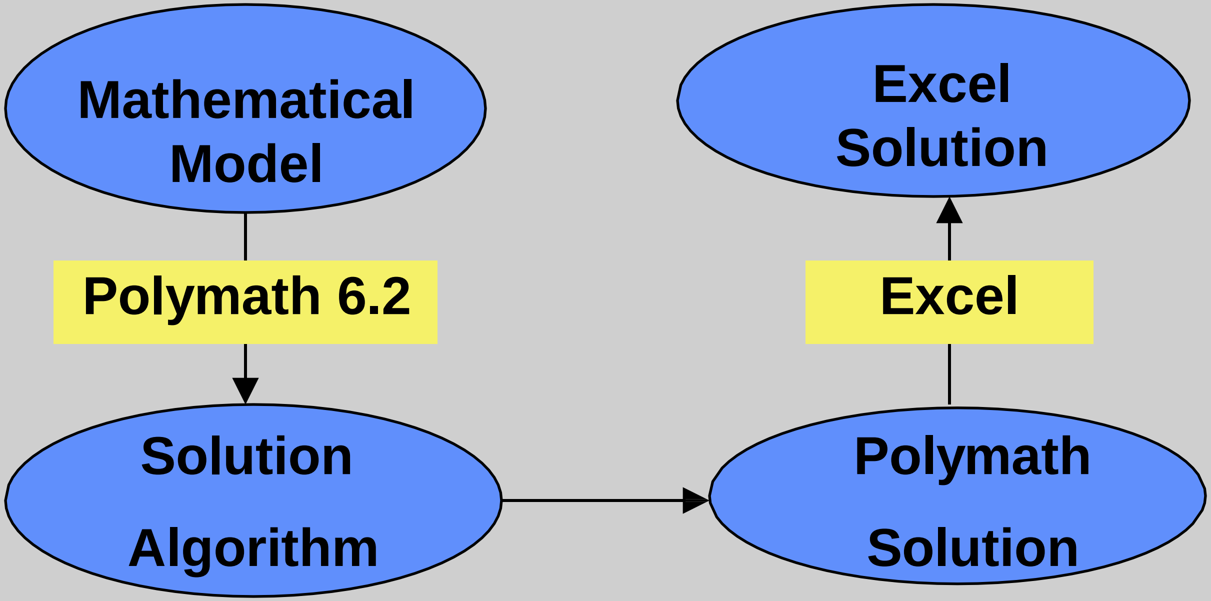
Consider laboratory data for the hardening of cement with four components.



	Wpc1	Wpc2	Wpc3	Wpc4	hard_heat
01	7	26	6	60	78.7
02	1	29	15	52	74.3
03	11	56	8	20	104.3
04	11	31	8	47	87.6
05	7	52	6	33	95.9
06	11	55	9	22	109.2
07	3	71	17	6	102.7
08	1	31	22	44	72.5
09	2	54	18	22	93.1
10	21	47	4	26	115.9
11	1	40	23	34	83.8
12	11	66	9	12	113.3
13	10	68	8	12	109.4

Problem 5 – Regression of Hardening of Concrete

Regression – Multiple Linear



Mathematical Model

Polymath 6.2

Solution Algorithm

Polymath Solution

Excel

Excel Solution

Problem 5 – Regression of Hardening of Concrete

Regression – Multiple Linear

Use Multiple Linear Regression to correlate the hardening of cement with four components.

This Polymath option will fit a linear function of the form:

$y(x_1, x_2, ..., x_n) = a_0 + a_1*x_1 + a_2*x_2 + ... + a_n*x_n$

where a_0, a_1, ..., a_n are regression parameters, to a set of N tabulated values of x_1, x_2, ..., x_n (independent variables) versus y (dependent variable). Note that the number of data points must be greater than n+1 (thus N >= n+1). The program calculates the coefficients a_0, a_1, ..., a_n by minimizing the sum of squares of the deviations between the calculated and the data for y.

Problem 5 – Regression of Hardening of Concrete

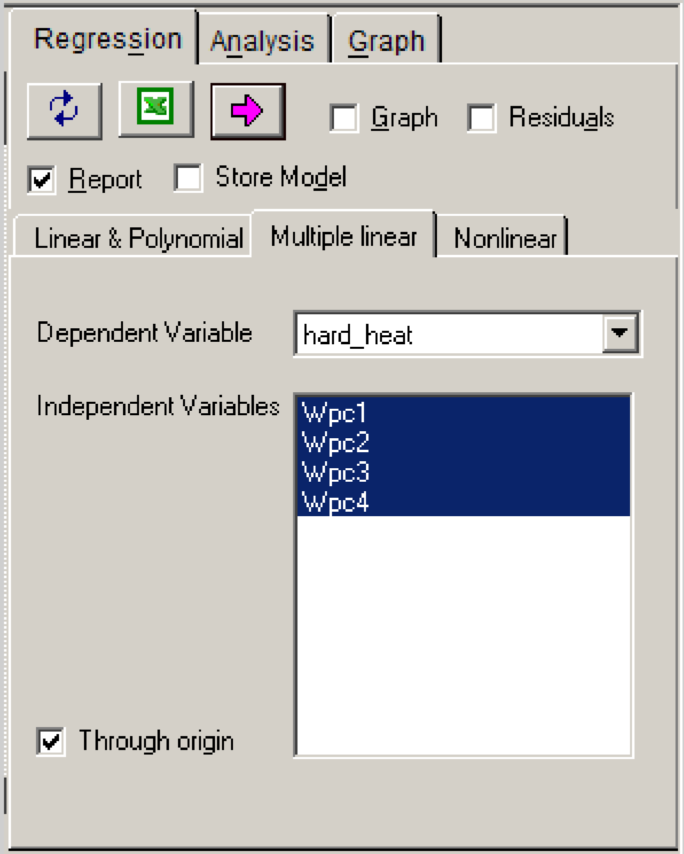
Regression – Multiple Linear

Live Demonstration of the Polymath Solution

Let's Go to POLYMATH and Generate the Problem Solution

POLYMATH – This file is Regression01.pol

Use the Regression Program to carry out a Multiple Linear Regression using the variables indicated where the holding down the Ctrl key allows all independent variables to be selected. This case yields the lowest variance.



☐ Graph ☐ Residuals

☑ Report ☐ Store Model

Dependent Variable: hard_heat

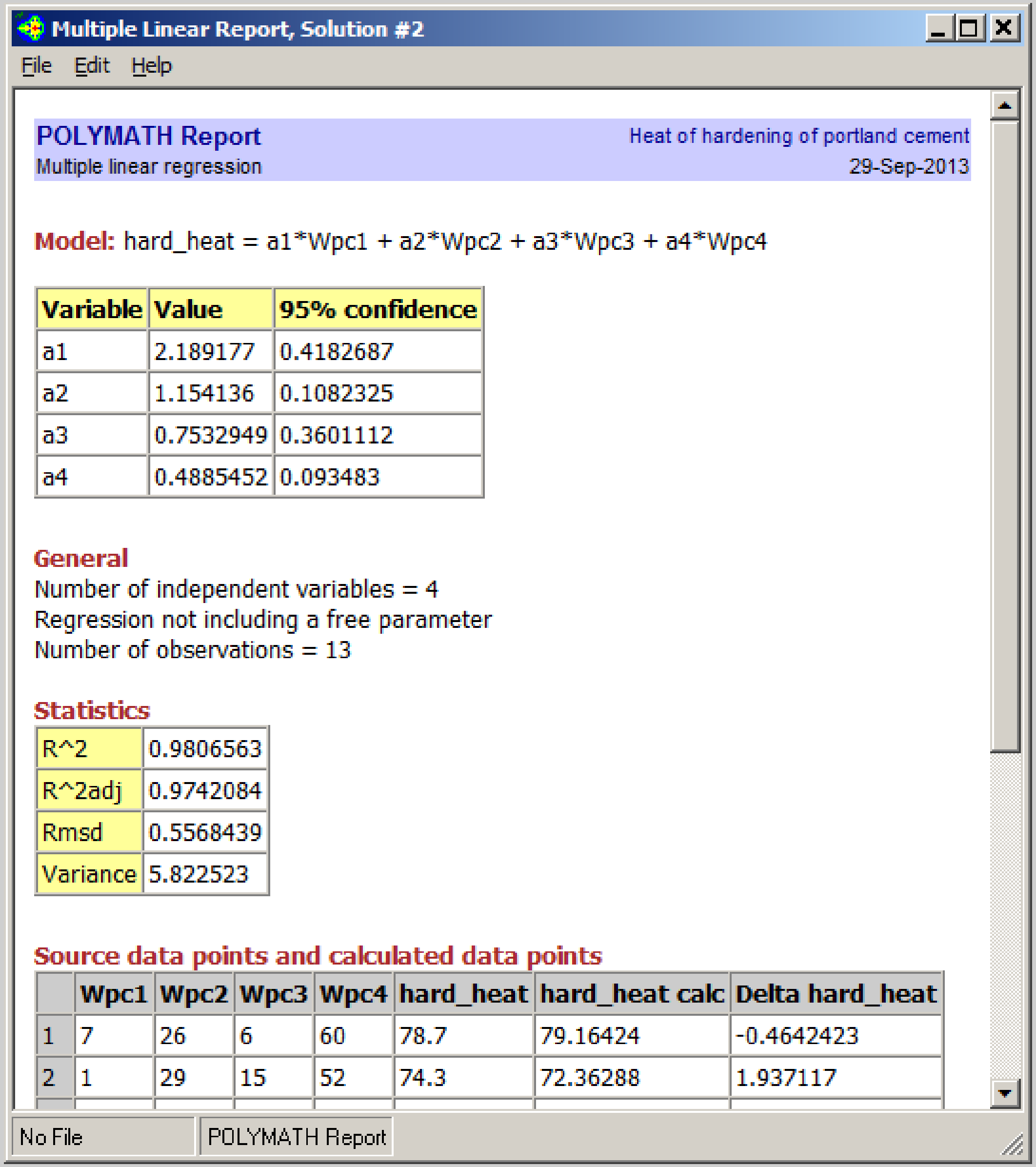
Independent Variables: Wpc1, Wpc2, Wpc3, Wpc4

☑ Through origin

Problem 5 – Regression of Hardening of Concrete

Regression – Multiple Linear

POLYMATH Multiple Linear Problem Report (through the origin)



POLYMATH Report
Multiple linear regression

Heat of hardening of portland cement
29-Sep-2013

Model: hard_heat = a1*Wpc1 + a2*Wpc2 + a3*Wpc3 + a4*Wpc4

Variable	Value	95% confidence
a1	2.189177	0.4182687
a2	1.154136	0.1082325
a3	0.7532949	0.3601112
a4	0.4885452	0.093483

General
Number of independent variables = 4
Regression not including a free parameter
Number of observations = 13

Statistics

R^2	0.9806563
R^2adj	0.9742084
Rmsd	0.5568439
Variance	5.822523

Source data points and calculated data points

	Wpc1	Wpc2	Wpc3	Wpc4	hard_heat	hard_heat calc	Delta hard_heat
1	7	26	6	60	78.7	79.16424	-0.4642423
2	1	29	15	52	74.3	72.36288	1.937117

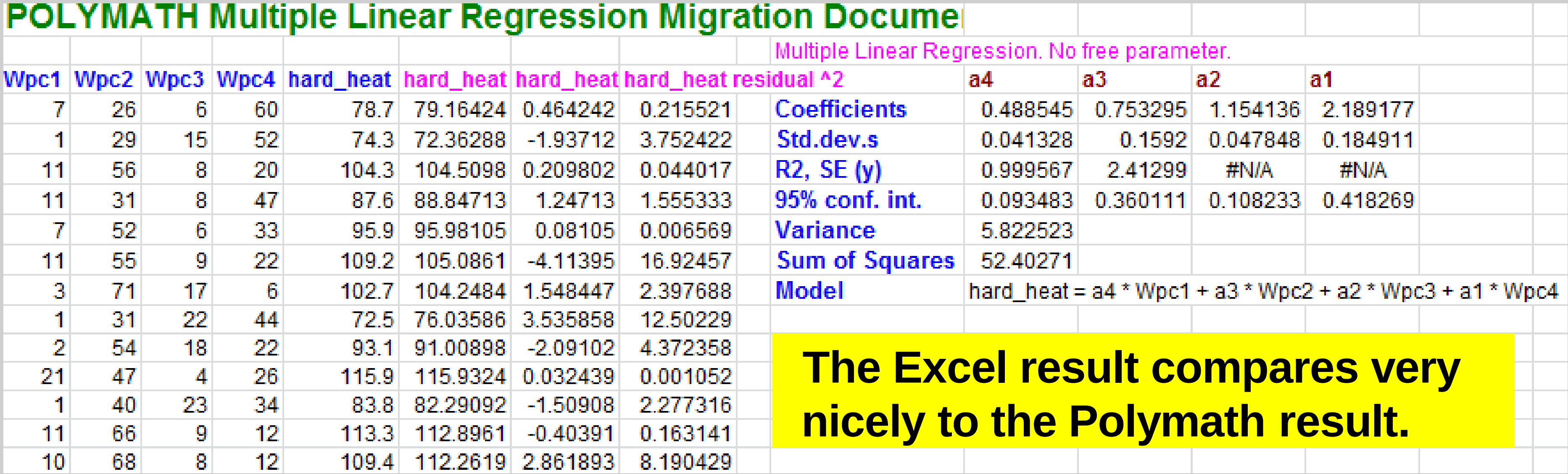
Problem 5 – Regression of Hardening of Concrete

Regression – Multiple Linear

Live Demonstration of the Polymath Solution and Solution of Same Problem in Excel

Let's Open Excel and Export Polymath Problem to Excel.

Excel – The Solution file is Regression01.xls



POLYMATH Multiple Linear Regression Migration Docume

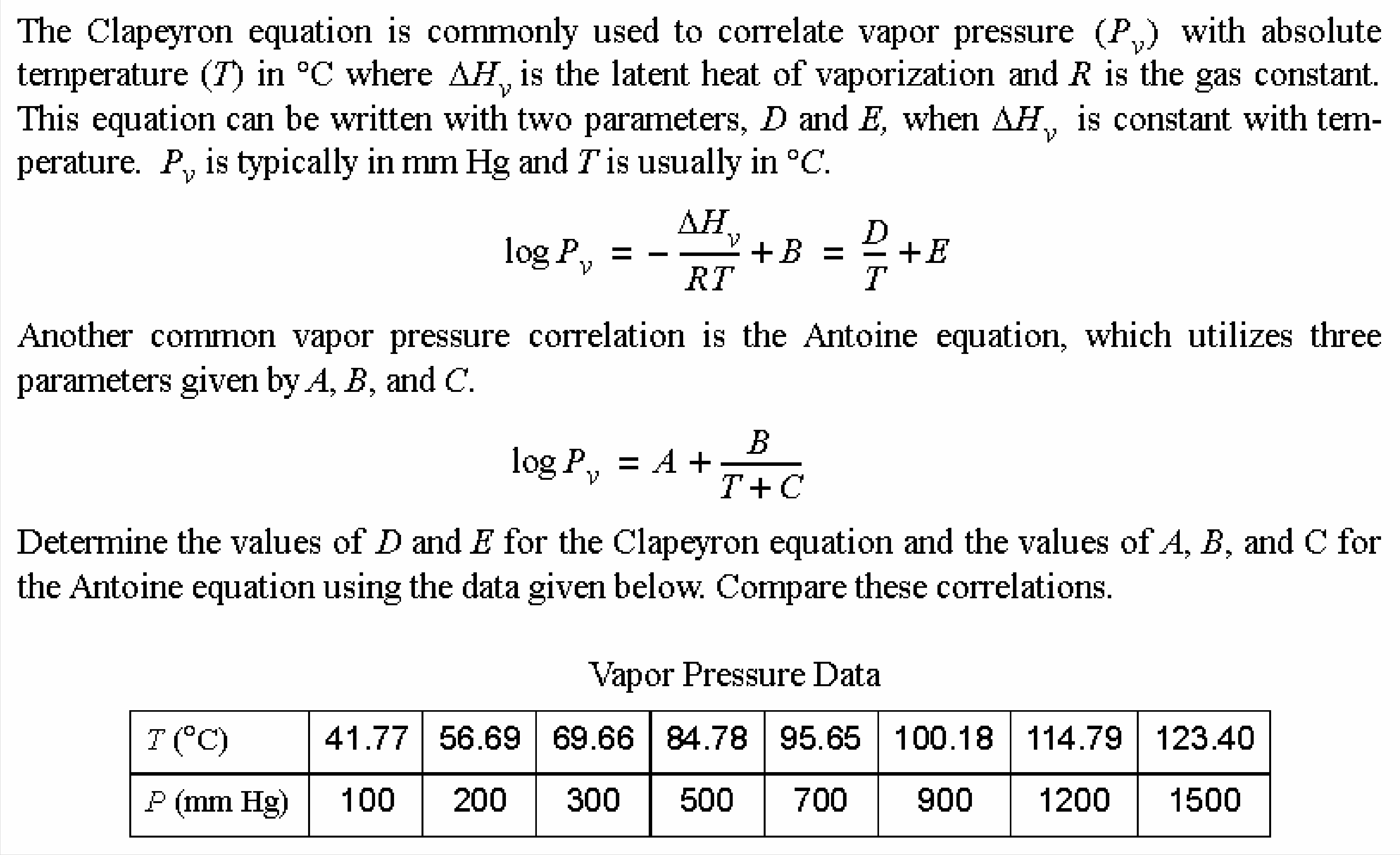
Wpc1	Wpc2	Wpc3	Wpc4	hard_heat	hard_heat	hard_heat	hard_heat residual ^2
7	26	6	60	78.7	79.16424	0.464242	0.215521
1	29	15	52	74.3	72.36288	-1.93712	3.752422
11	56	8	20	104.3	104.5098	0.209802	0.044017
11	31	8	47	87.6	88.84713	1.24713	1.555333
7	52	6	33	95.9	95.98105	0.08105	0.006569
11	55	9	22	109.2	105.0861	-4.11395	16.92457
3	71	17	6	102.7	104.2484	1.548447	2.397688
1	31	22	44	72.5	76.03586	3.535858	12.50229
2	54	18	22	93.1	91.00898	-2.09102	4.372358
21	47	4	26	115.9	115.9324	0.032439	0.001052
1	40	23	34	83.8	82.29092	-1.50908	2.277316
11	66	9	12	113.3	112.8961	-0.40391	0.163141
10	68	8	12	109.4	112.2619	2.861893	8.190429

Multiple Linear Regression. No free parameter.

	a4	a3	a2	a1
Coefficients	0.488545	0.753295	1.154136	2.189177
Std.dev.s	0.041328	0.1592	0.047848	0.184911
R2, SE (y)	0.999567	2.41299	#N/A	#N/A
95% conf. int.	0.093483	0.360111	0.108233	0.418269
Variance	5.822523			
Sum of Squares	52.40271			
Model	hard_heat = a4 * Wpc1 + a3 * Wpc2 + a2 * Wpc3 + a1 * Wpc4			

The Excel result compares very nicely to the Polymath result.

Problem 6 – Regressions – Vapor Pressure Data



The Clapeyron equation is commonly used to correlate vapor pressure (P_v) with absolute temperature (T) in °C where ΔH_v is the latent heat of vaporization and R is the gas constant. This equation can be written with two parameters, D and E, when ΔH_v is constant with temperature. P_v is typically in mm Hg and T is usually in °C.

$$\log P_v = -\frac{\Delta H_v}{RT} + B = \frac{D}{T} + E$$

Another common vapor pressure correlation is the Antoine equation, which utilizes three parameters given by A, B, and C.

$$\log P_v = A + \frac{B}{T + C}$$

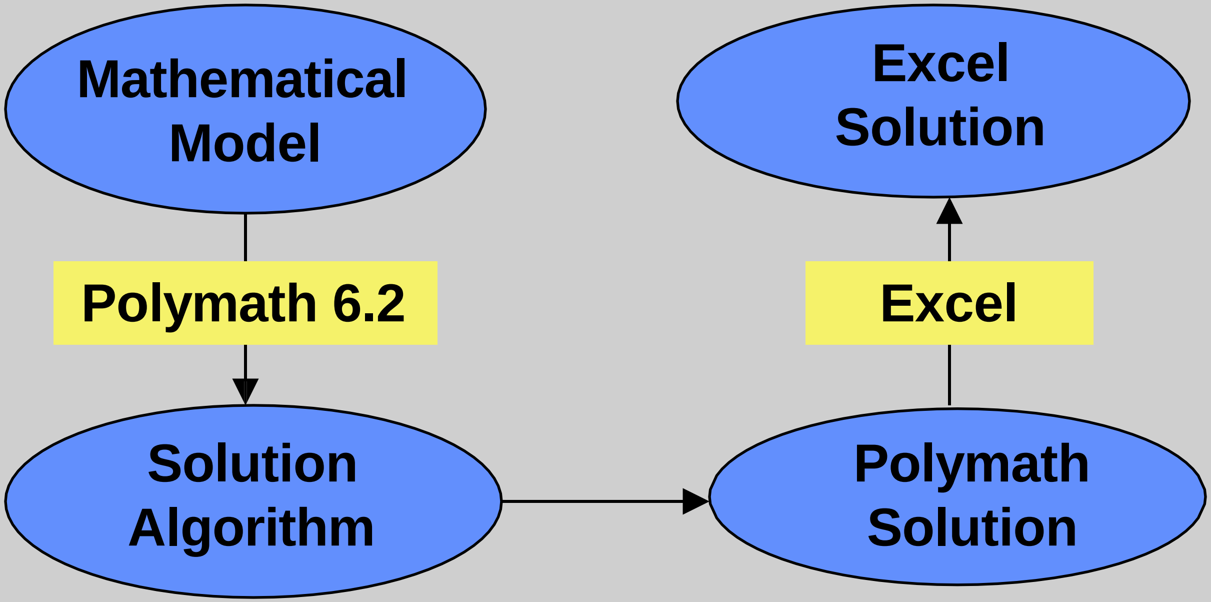
Determine the values of D and E for the Clapeyron equation and the values of A, B, and C for the Antoine equation using the data given below. Compare these correlations.

Vapor Pressure Data

T (°C)	41.77	56.69	69.66	84.78	95.65	100.18	114.79	123.40
P (mm Hg)	100	200	300	500	700	900	1200	1500

Problem 6 – Regressions - Vapor Pressure Data

Regressions – Linear and Nonlinear



Mathematical Model

Polymath 6.2

Solution Algorithm

Polymath Solution

Excel

Excel Solution

Problem 6 – Regressions – Vapor Pressure Data

POLYMATH Clapeyron Equation Linear Regression **EXERCISE**

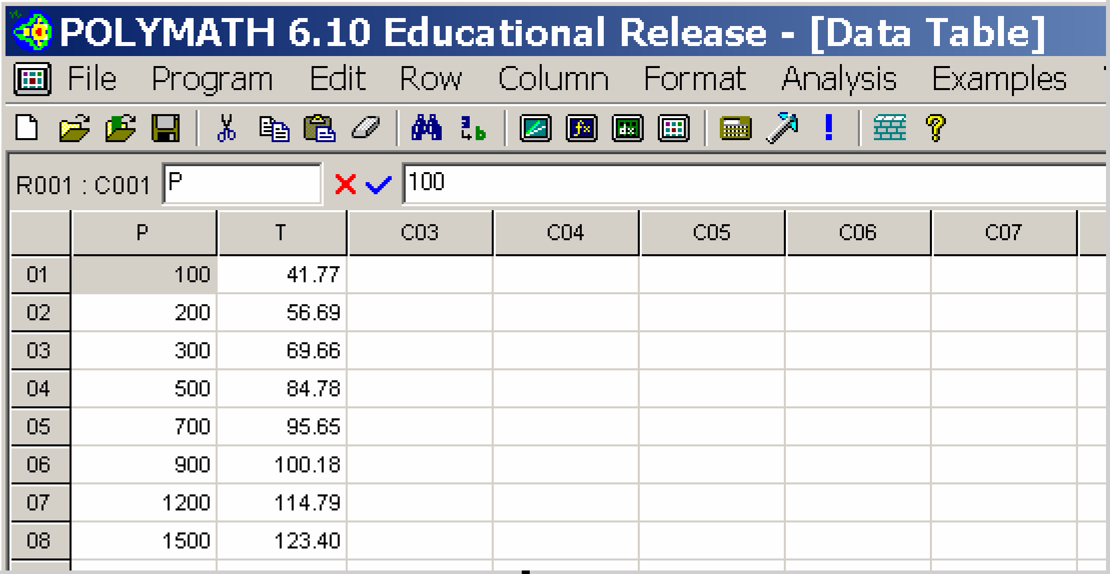
Utilize the Polymath Regression Program to input the data to the Data Table.

Create a new column for a variable logP that is the log of the pressure.

logP = log(P)

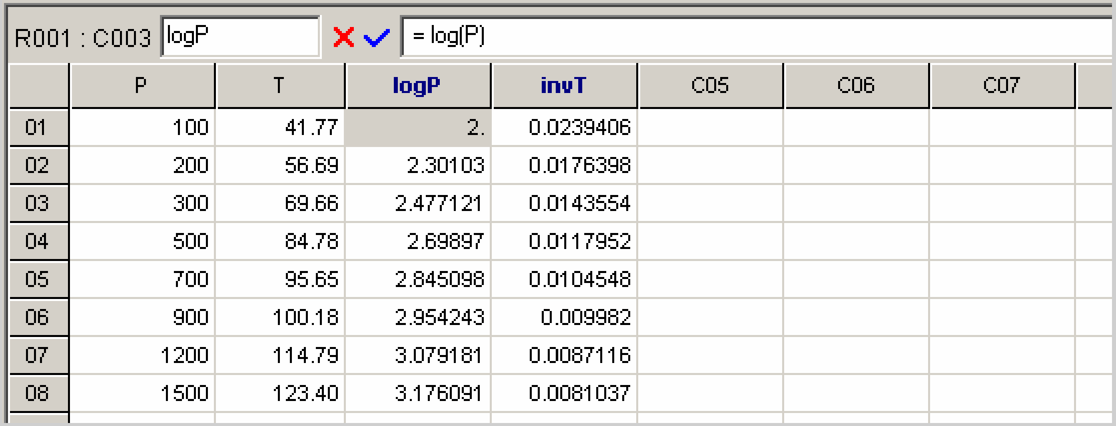
Then create another column for a variable invT that is the inverse of the temperature in ^{o}C.

InvT = 1/T



R001 : C001 P 100

	P	T	C03	C04	C05	C06	C07
01	100	41.77					
02	200	56.69					
03	300	69.66					
04	500	84.78					
05	700	95.65					
06	900	100.18					
07	1200	114.79					
08	1500	123.40					



R001 : C003 logP = log(P)

	P	T	**logP**	**invT**	C05	C06	C07
01	100	41.77	2.	0.0239406			
02	200	56.69	2.30103	0.0176398			
03	300	69.66	2.477121	0.0143554			
04	500	84.78	2.69897	0.0117952			
05	700	95.65	2.845098	0.0104548			
06	900	100.18	2.954243	0.009982			
07	1200	114.79	3.079181	0.0087116			
08	1500	123.40	3.176091	0.0081037			

Problem 6 – Regressions – Vapor Pressure Data POLYMATH/Excel Solution EXERCISE

Utilize the Polymath Regression Program to make a Linear Regression of logP versus invTK to yield the parameters D and E of the Clapeyron equation.

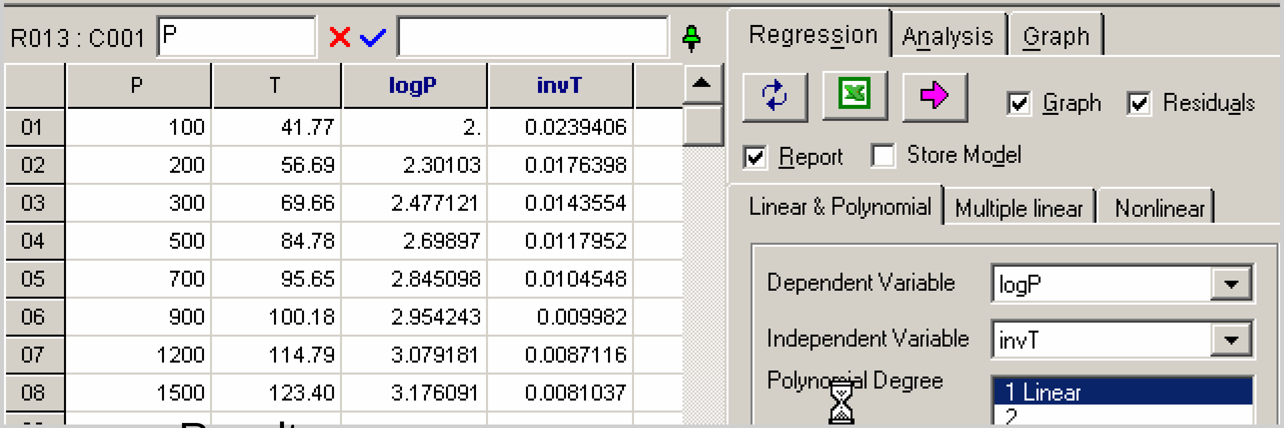
E = a0 = 3.658

D = a1 = -73.61

Use the Polymath Problem Data File

OR

Use the Polymath Solution File



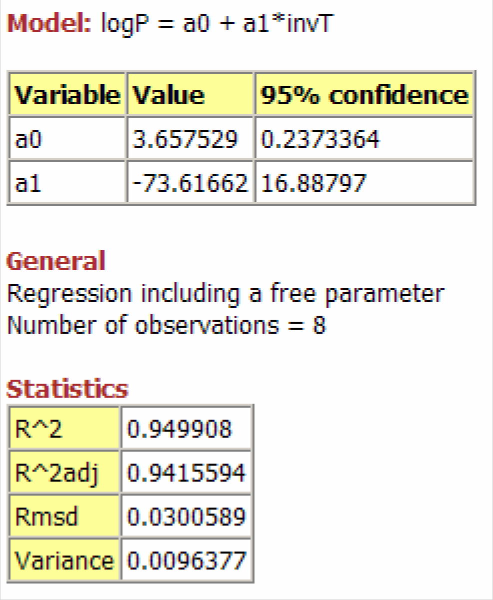
	P	T	logP	invT
01	100	41.77	2.	0.0239406
02	200	56.69	2.30103	0.0176398
03	300	69.66	2.477121	0.0143554
04	500	84.78	2.69897	0.0117952
05	700	95.65	2.845098	0.0104548
06	900	100.18	2.954243	0.009982
07	1200	114.79	3.079181	0.0087116
08	1500	123.40	3.176091	0.0081037

Dependent Variable: logP

Independent Variable: invT

Polynomial Degree: 1 Linear
2

Results



Model: logP = a0 + a1*invT

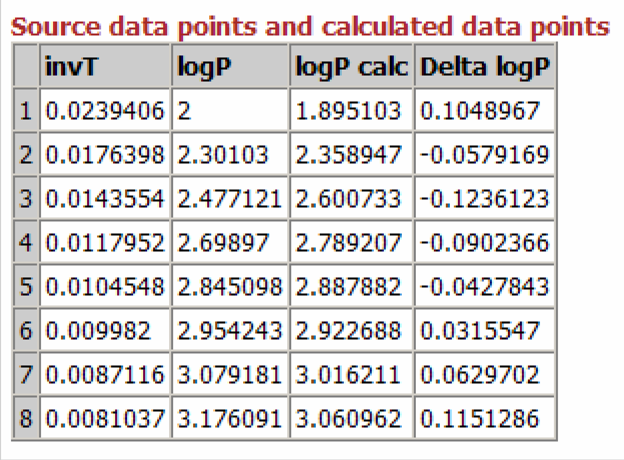
Variable	Value	95% confidence
a0	3.657529	0.2373364
a1	-73.61662	16.88797

General

Regression including a free parameter
Number of observations = 8

Statistics

R^2	0.949908
R^2adj	0.9415594
Rmsd	0.0300589
Variance	0.0096377



Source data points and calculated data points

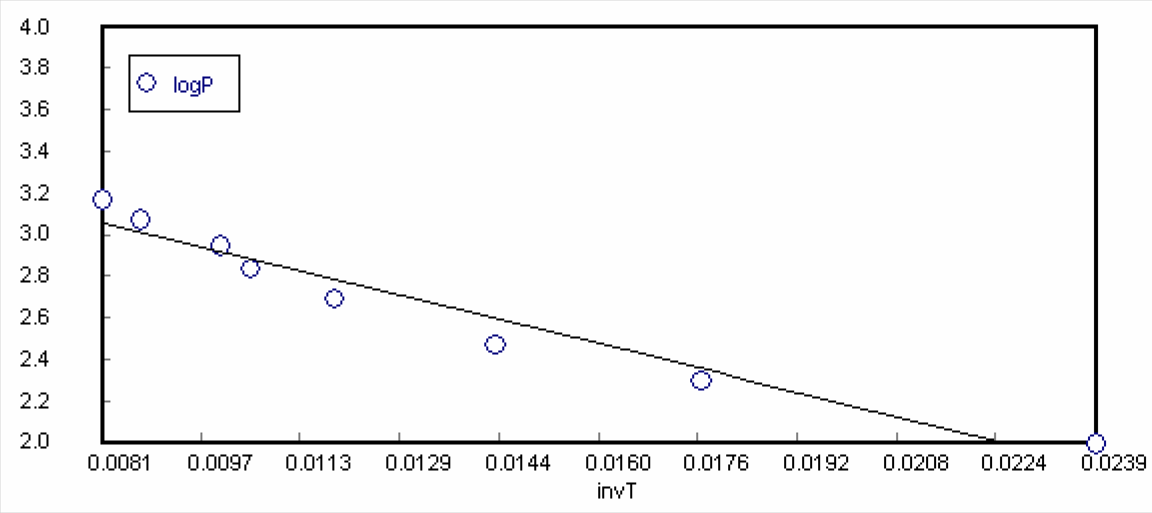
	invT	logP	logP calc	Delta logP
1	0.0239406	2	1.895103	0.1048967
2	0.0176398	2.30103	2.358947	-0.0579169
3	0.0143554	2.477121	2.600733	-0.1236123
4	0.0117952	2.69897	2.789207	-0.0902366
5	0.0104548	2.845098	2.887882	-0.0427843
6	0.009982	2.954243	2.922688	0.0315547
7	0.0087116	3.079181	3.016211	0.0629702
8	0.0081037	3.176091	3.060962	0.1151286

Polymath Data File is RegressionData02.pol

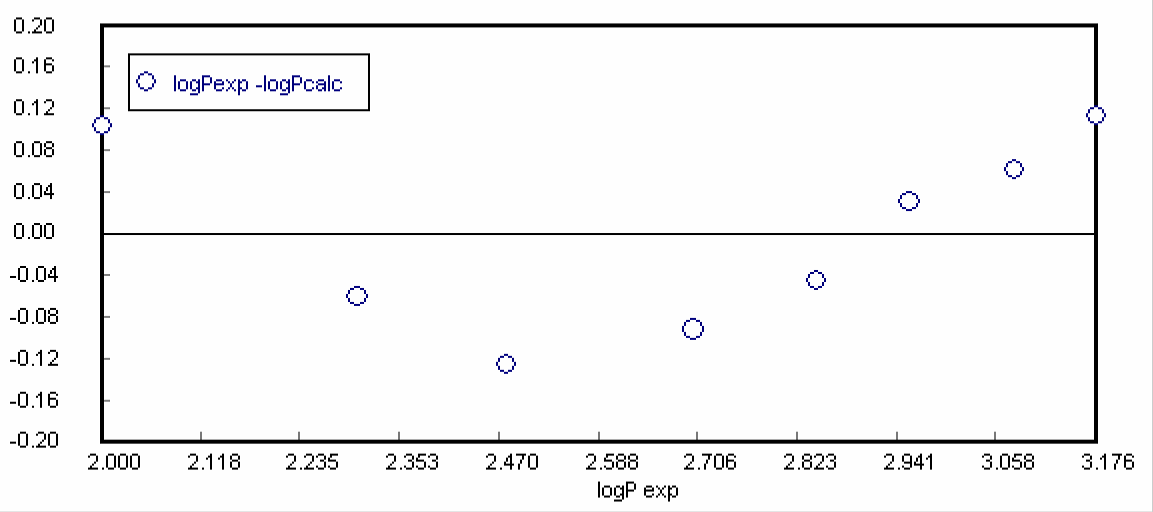
Polymath Solution File is Regression02.pol

Problem 6 – Regressions – Vapor Pressure Data POLYMATH/Excel Solution EXERCISE

The Graph Option from the Polymath Regression Program indicates a reasonable representation of the data.

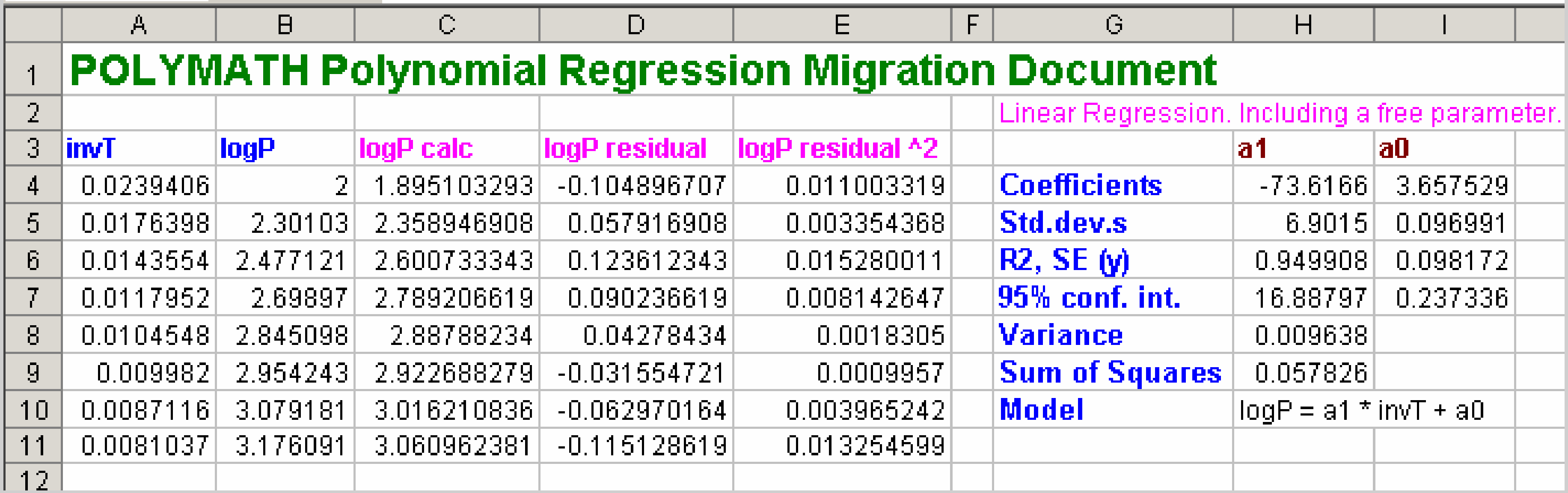


However, the Residuals Plot Option shows a trend in the errors.



Problem 6 – Regressions – Vapor Pressure Data POLYMATH/Excel Solution EXERCISE

Utilize the Export to EXCEL Option from the Polymath Regression Program to make a Linear Regression of logP versus invTK. The results, shown below, are essentially the same as those obtained with Polymath.



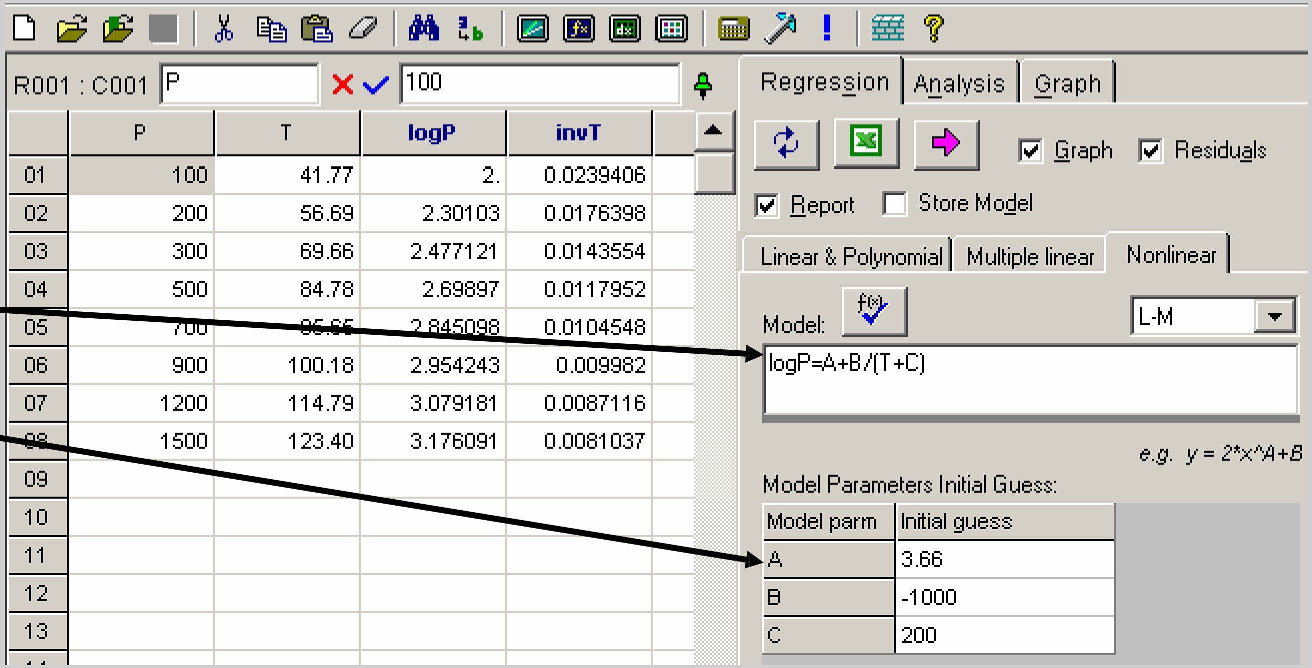
	A	B	C	D	E	F	G	H	I
1	**POLYMATH Polynomial Regression Migration Document**								
2							Linear Regression. Including a free parameter.		
3	**invT**	**logP**	**logP calc**	**logP residual**	**logP residual ^2**			**a1**	**a0**
4	0.0239406	2	1.895103293	-0.104896707	0.011003319		**Coefficients**	-73.6166	3.657529
5	0.0176398	2.30103	2.358946908	0.057916908	0.003354368		**Std.dev.s**	6.9015	0.096991
6	0.0143554	2.477121	2.600733343	0.123612343	0.015280011		**R2, SE (y)**	0.949908	0.098172
7	0.0117952	2.69897	2.789206619	0.090236619	0.008142647		**95% conf. int.**	16.88797	0.237336
8	0.0104548	2.845098	2.88788234	0.04278434	0.0018305		**Variance**	0.009638	
9	0.009982	2.954243	2.922688279	-0.031554721	0.0009957		**Sum of Squares**	0.057826	
10	0.0087116	3.079181	3.016210836	-0.062970164	0.003965242		**Model**	logP = a1 * invT + a0	
11	0.0081037	3.176091	3.060962381	-0.115128619	0.013254599				
12									

You may need the EXCEL Problem Solution File

File is Regression02.xls

Problem 6 – Regressions - Vapor Pressure Data POLYMATH/Excel Solution EXERCISE

Utilize the Polymath Regression Program to make a Nonlinear Regression of the Antoine Equation. Use the initial guesses as shown. Plot the Graph and the Residual for this regression.



R001 : C001 P 100

	P	T	logP	invT
01	100	41.77	2.	0.0239406
02	200	56.69	2.30103	0.0176398
03	300	69.66	2.477121	0.0143554
04	500	84.78	2.69897	0.0117952
05	700	95.65	2.845098	0.0104548
06	900	100.18	2.954243	0.009982
07	1200	114.79	3.079181	0.0087116
08	1500	123.40	3.176091	0.0081037
09				
10				
11				
12				
13				

☑ Graph ☑ Residuals

☑ Report ☐ Store Model

Model: L-M

logP=A+B/(T+C)

e.g. y = 2*x^A+B

Model Parameters Initial Guess:

Model parm	Initial guess
A	3.66
B	-1000
C	200

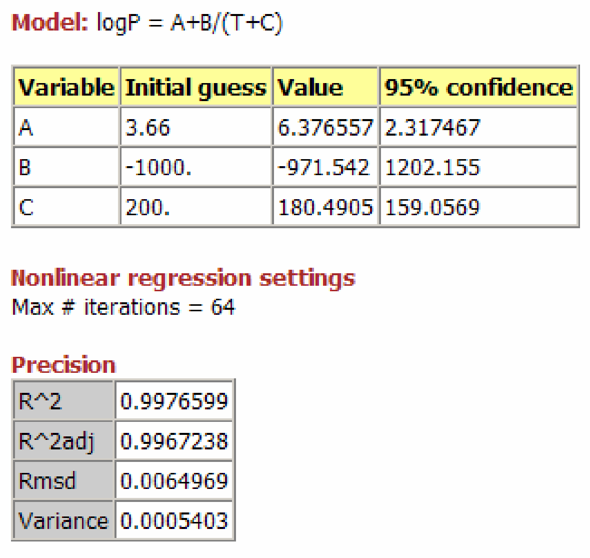
You may use the Polymath Problem Data File

File is RegressionData02.pol

OR

You may use the Polymath Solution File

File is Regression03.pol



Model: logP = A+B/(T+C)

Variable	Initial guess	Value	95% confidence
A	3.66	6.376557	2.317467
B	-1000.	-971.542	1202.155
C	200.	180.4905	159.0569

Nonlinear regression settings

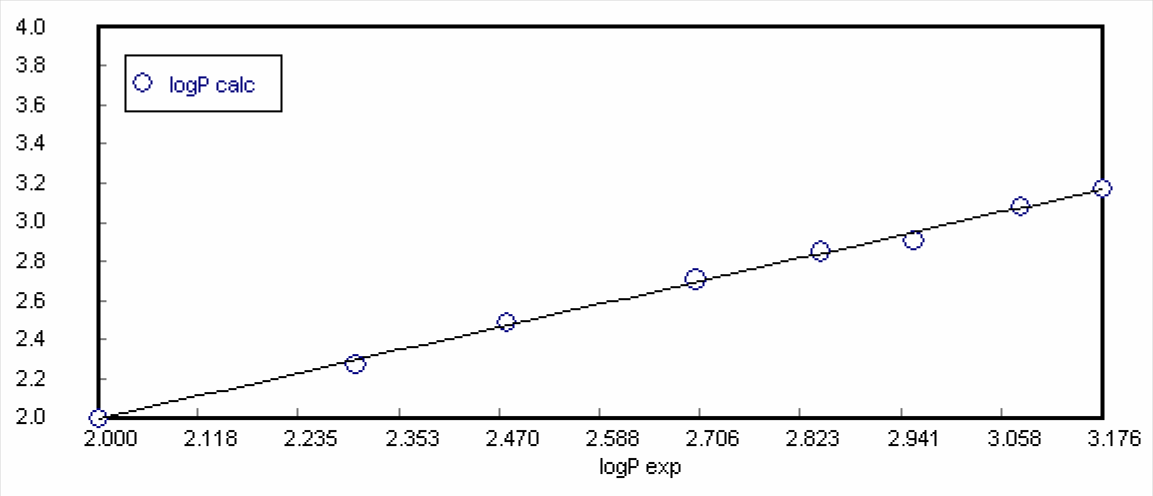
Max # iterations = 64

Precision

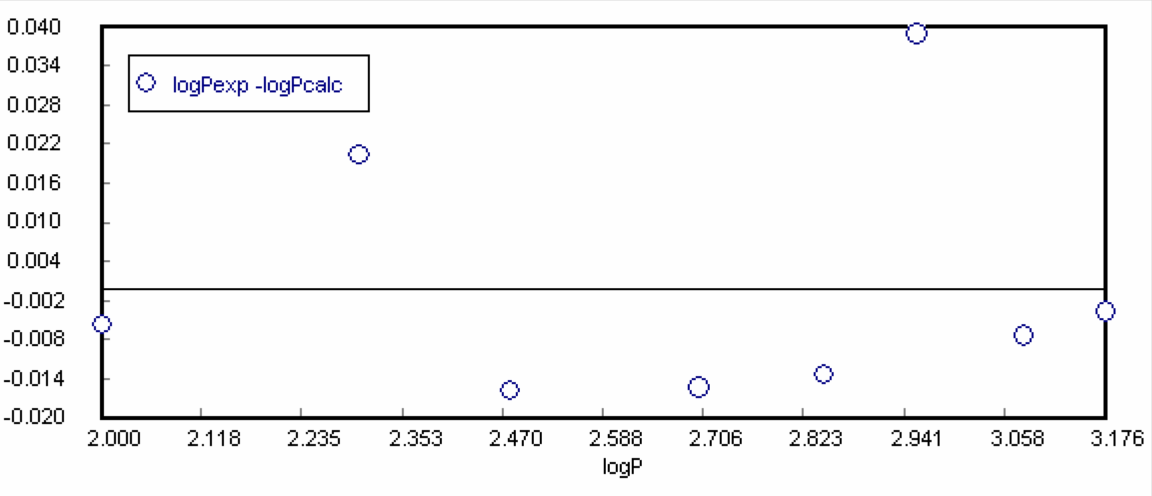
R^2	0.9976599
R^2adj	0.9967238
Rmsd	0.0064969
Variance	0.0005403

Problem 6 – Regressions – Vapor Pressure Data POLYMATH/Excel Solution EXERCISE

The Graph Option from the Polymath Nonlinear Regression Program indicates a reasonable representation of the data.



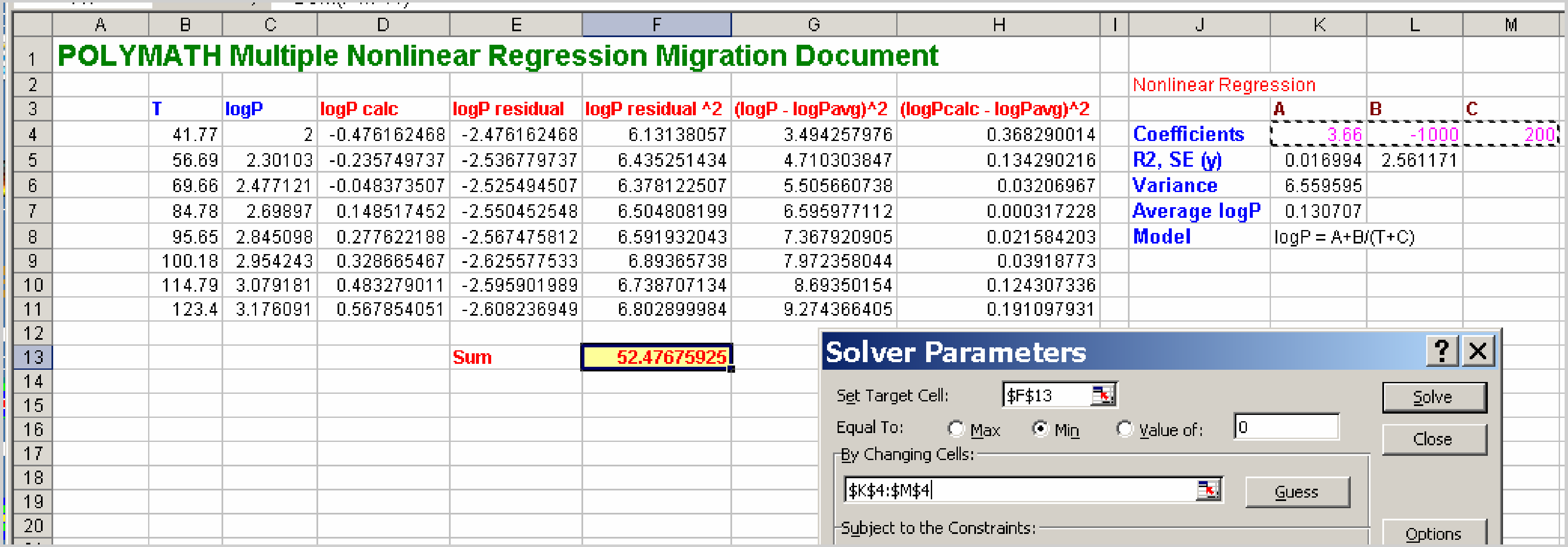
The Residuals Plot Option shows a more random distribution of the errors.



These graphs plus the lower variance for the Antoine equation indicate that the data are well represented.

Problem 6 – Regressions – Vapor Pressure Data POLYMATH/Excel Solution EXERCISE

Utilize the Export to EXCEL Option from the Polymath Regression Program to make a Nonlinear Regression of logP versus invTK. The results, shown below, are essentially the same as those obtained with Polymath. Note that the EXCEL Add-In Solver must be used to complete the Nonlinear Regression.



	A	B	C	D	E	F	G	H	I	J	K	L	M
1	**POLYMATH Multiple Nonlinear Regression Migration Document**												
2										Nonlinear Regression			
3		**T**	**logP**	**logP calc**	**logP residual**	**logP residual ^2**	**(logP - logPavg)^2**	**(logPcalc - logPavg)^2**			**A**	**B**	**C**
4		41.77	2	-0.476162468	-2.476162468	6.13138057	3.494257976	0.368290014		**Coefficients**	3.66	-1000	200
5		56.69	2.30103	-0.235749737	-2.536779737	6.435251434	4.710303847	0.134290216		**R2, SE (y)**	0.016994	2.561171	
6		69.66	2.477121	-0.048373507	-2.525494507	6.378122507	5.505660738	0.03206967		**Variance**	6.559595		
7		84.78	2.69897	0.148517452	-2.550452548	6.504808199	6.595977112	0.000317228		**Average logP**	0.130707		
8		95.65	2.845098	0.277622188	-2.567475812	6.591932043	7.367920905	0.021584203		**Model**	logP = A+B/(T+C)		
9		100.18	2.954243	0.328665467	-2.625577533	6.89365738	7.972358044	0.03918773					
10		114.79	3.079181	0.483279011	-2.595901989	6.738707134	8.69350154	0.124307336					
11		123.4	3.176091	0.567854051	-2.608236949	6.802899984	9.274366405	0.191097931					
12													
13					Sum	52.47675925							

Solver Parameters

Set Target Cell: $F$13

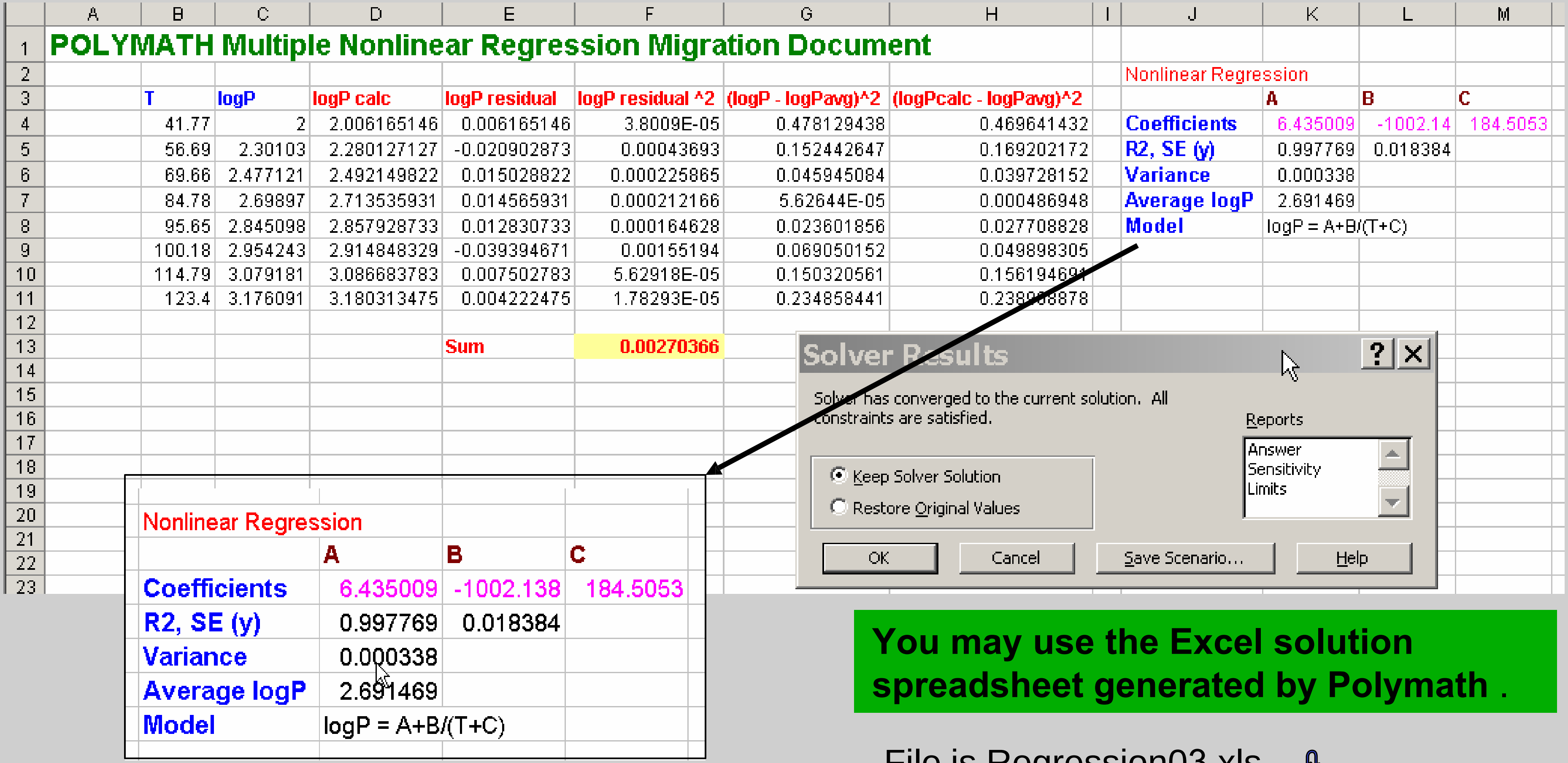
Equal To: ○ Max ● Min ○ Value of: 0

By Changing Cells: $K$4:$M$4

Subject to the Constraints:

Problem 6 – Regressions – Vapor Pressure Data POLYMATH/Excel Solution EXERCISE

The EXCEL Nonlinear Regression results, obtained with Solver that is available from the Data tab, are shown below in spreadsheet and magnified view. Results are essentially the same as those obtained with Polymath.



POLYMATH Multiple Nonlinear Regression Migration Document

T	logP	logP calc	logP residual	logP residual ^2	(logP - logPavg)^2	(logPcalc - logPavg)^2
41.77	2	2.006165146	0.006165146	3.8009E-05	0.478129438	0.469641432
56.69	2.30103	2.280127127	-0.020902873	0.00043693	0.152442647	0.169202172
69.66	2.477121	2.492149822	0.015028822	0.000225865	0.045945084	0.039728152
84.78	2.69897	2.713535931	0.014565931	0.000212166	5.62644E-05	0.000486948
95.65	2.845098	2.857928733	0.012830733	0.000164628	0.023601856	0.027708828
100.18	2.954243	2.914848329	-0.039394671	0.00155194	0.069050152	0.049898305
114.79	3.079181	3.086683783	0.007502783	5.62918E-05	0.150320561	0.156194691
123.4	3.176091	3.180313475	0.004222475	1.78293E-05	0.234858441	0.238908878
			Sum	**0.00270366**		

Nonlinear Regression

	A	B	C
Coefficients	6.435009	-1002.14	184.5053
R2, SE (y)	0.997769	0.018384	
Variance	0.000338		
Average logP	2.691469		
Model	logP = A+B/(T+C)		

Solver Results

Solver has converged to the current solution. All constraints are satisfied.

● Keep Solver Solution

○ Restore Original Values

Reports

Answer
Sensitivity
Limits

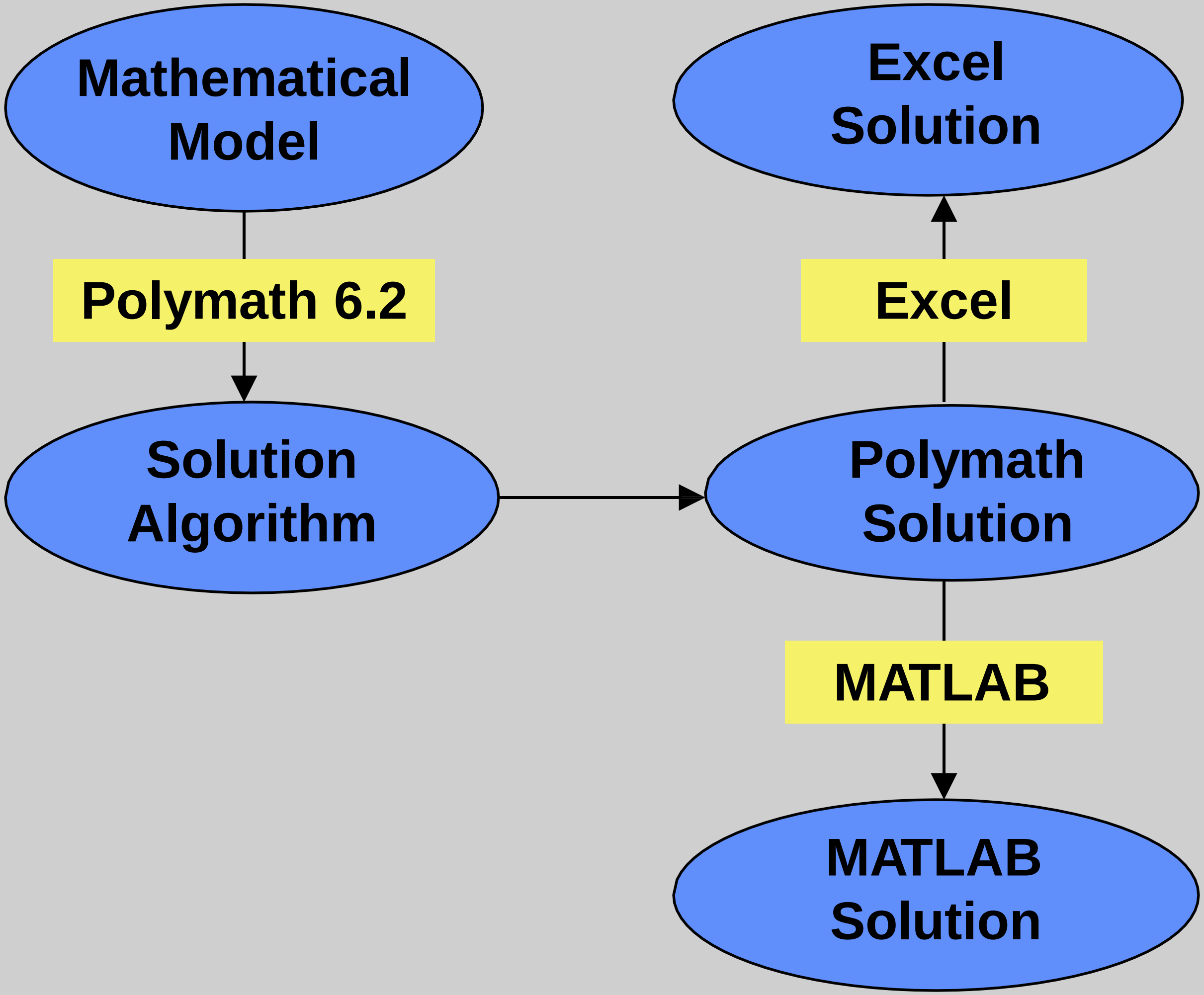
Nonlinear Regression

	A	B	C
Coefficients	6.435009	-1002.138	184.5053
R2, SE (y)	0.997769	0.018384	
Variance	0.000338		
Average logP	2.691469		
Model	logP = A+B/(T+C)		

You may use the Excel solution spreadsheet generated by Polymath .

File is Regression03.xls

SUMMARY - Desktop Problem Solving Involving Polymath, Excel, and MATLAB



Mathematical Model

Polymath 6.2

Solution Algorithm

Polymath Solution

Excel

Excel Solution

MATLAB

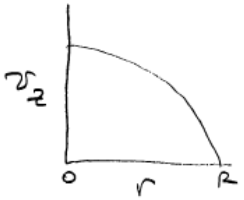
MATLAB Solution

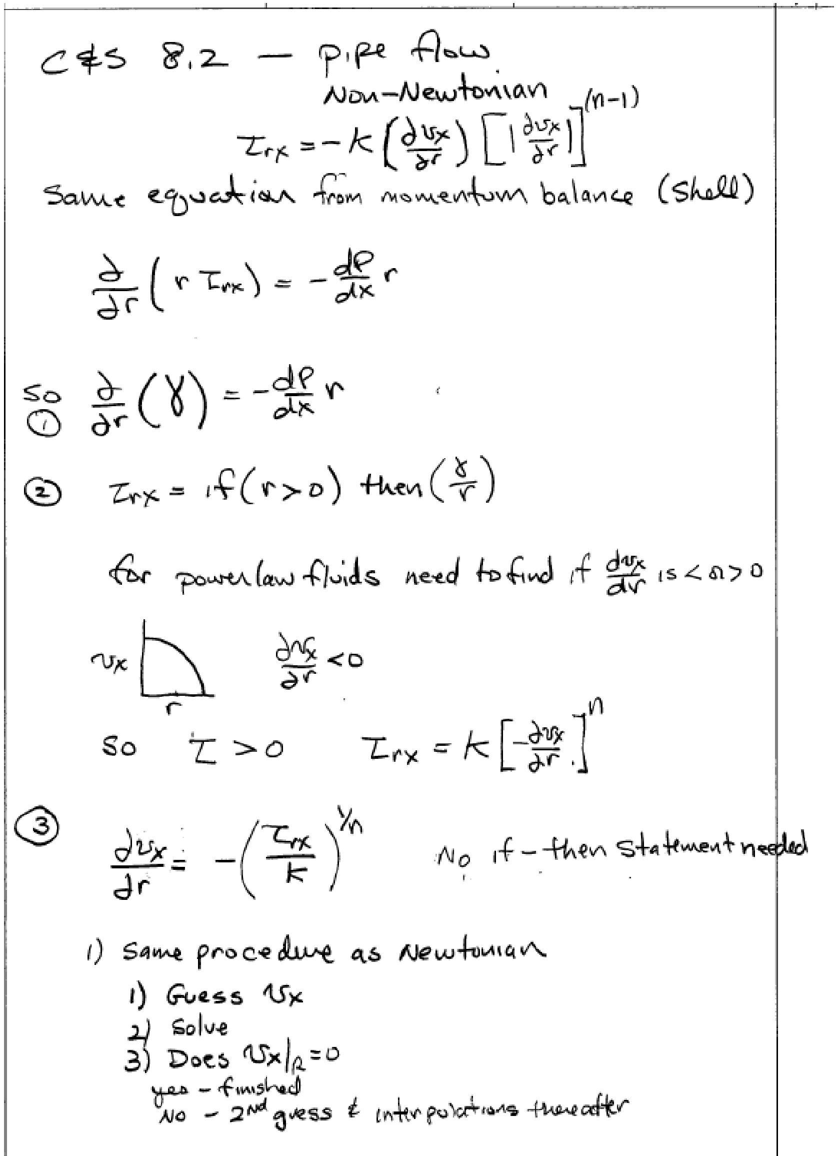
Happy Problem Solving!

Advanced Topics

2nd Order ODE or 2 coupled ODE's with Split Boundary Conditions

- NonNewtonian fluid flow through a pipe (POLYMATH Text 8.2c)
- $\frac{d(r\tau_{rx})}{dr} = -\frac{dP}{dx}r$
- $\tau_{rx} = -K\left(\frac{dv_x}{dr}\right)\left(\left|\frac{dv_x}{dr}\right|\right)^{(n-1)}$
- Boundary Conditions:
  $r = 0\ v_x = \max \quad \tau_{rx} = 0$
  $r = wall\ v_x = 0\ \tau_{rx} = max$





C&S 8.2 — pipe flow
Non-Newtonian

$$\tau_{rx} = -K\left(\frac{\partial v_x}{\partial r}\right)\left[\left|\frac{\partial v_x}{\partial r}\right|\right]^{(n-1)}$$

Same equation from momentum balance (Shell)

$$\frac{\partial}{\partial r}\left(r\,\tau_{rx}\right) = -\frac{dP}{dx}r$$

so ① $\frac{\partial}{\partial r}\left(\gamma\right) = -\frac{dP}{dx}r$

② $\tau_{rx} = \text{if}\,(r>0)\ \text{then}\left(\frac{\gamma}{r}\right)$

for power law fluids need to find if $\frac{dv_x}{dr}$ is < or > 0

$\frac{\partial v_x}{\partial r} < 0$

so $\tau > 0$ $\quad \tau_{rx} = K\left[-\frac{\partial v_x}{\partial r}\right]^n$

③ $\frac{\partial v_x}{\partial r} = -\left(\frac{\tau_{rx}}{K}\right)^{1/n}$ No if-then statement needed

1) Same procedure as Newtonian
  1) Guess v_x
  2) Solve
  3) Does $v_x|_R = 0$
    yes - finished
    No - 2nd guess & interpolations thereafter

POLYMATH Report 8.2 a&b Laminar Flow of Non-Newtonian Fluids in a Horizontal Pipe
Ordinary Differential Equations

Calculated values of DEQ variables

	Variable	Initial value	Minimal value	Maximal value	Final value
1	delP	100.	100.	100.	100.
2	K	1.0E-06	1.0E-06	1.0E-06	1.0E-06
3	L	10.	10.	10.	10.
4	n	2.	2.	2.	2.
5	r	0	0	0.009295	0.009295
6	R	0.009295	0.009295	0.009295	0.009295
7	tau	0	0	0.046475	0.046475
8	tau_anal	0	0	0.046475	0.046475
9	tau_r	0	0	0.000432	0.000432
10	vavg	0	0	0.5725206	0.5725206
11	vx	1.335881	5.281E-08	1.335881	5.281E-08
12	vx_anal	1.335881	0	1.335881	0

This is a trial and error solution since you do not know the values for velocity at r=0. You do know that at r=0 the stress is zero. So you must guess a value for the velocity and then run POLYMATH. You will know that you have guessed the correct velocity when the velocity at r=R is very small e.g. nearly zero. To minimize the number of trials, guess 2 initial velocities and then use interpolation to find the 3rd guess. And then every new value after that will be found by interpolation.

Differential equations

1 d(tau_r)/d(r) = delP/L*r
2 d(vx)/d(r) = if(tau>0) then (-(tau/K)^(1/n)) else ((-tau/K)^(1/n))
3 d(vavg)/d(r) = vx*r*2/R^2

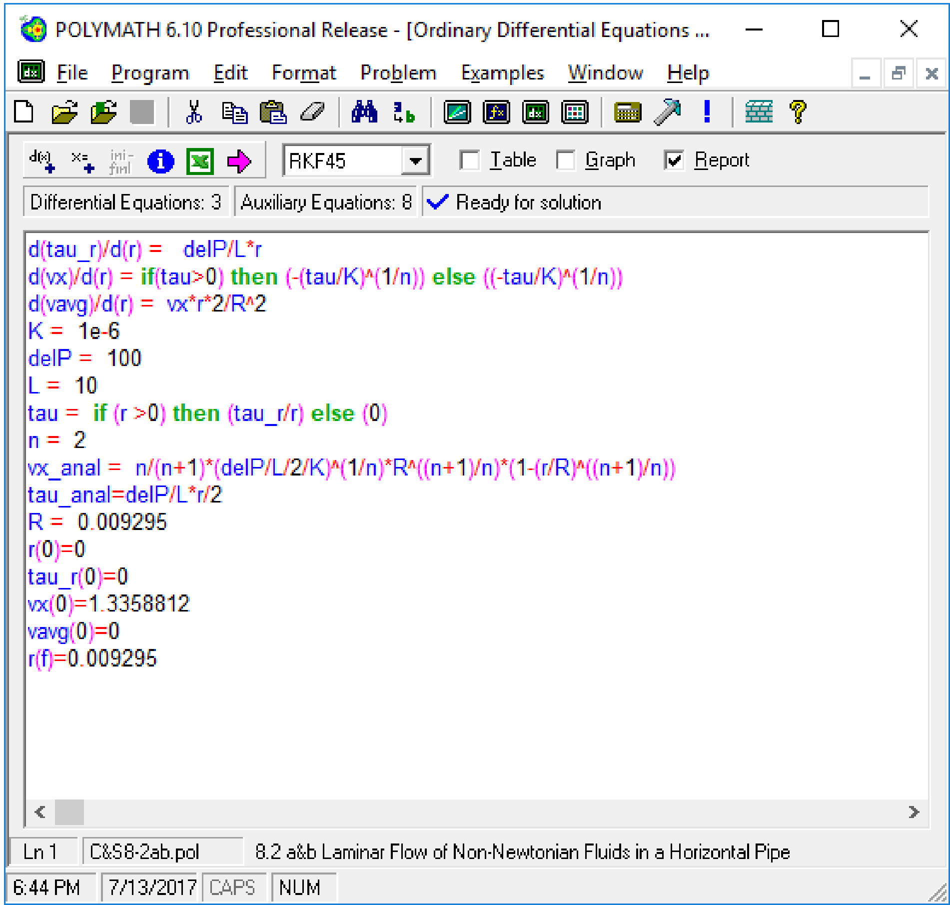
Explicit equations

1 K = 1e-6
2 delP = 100
3 L = 10
4 tau = if (r >0) then (tau_r/r) else (0)
5 n = 2
6 R = 0.009295
7 tau_anal = delP/L*r/2
8 vx_anal = n/(n+1)*(delP/L/2/K)^(1/n)*R^((n+1)/n)*(1-(r/R)^((n+1)/n))

```
Trial & Error Results
V at r=0   V at wall
1.5        0.1641189
0.5        -0.8358811
1.34       0.0041189
1.3359     0.0001189
```

General

Total number of equations	11
Number of differential equations	3
Number of explicit equations	8
Elapsed time	0.000 sec
Solution method	RKF_45
Step size guess. h	0.000001
Truncation error tolerance. eps	0.000001



RKF45 ☐ Table ☐ Graph ☑ Report

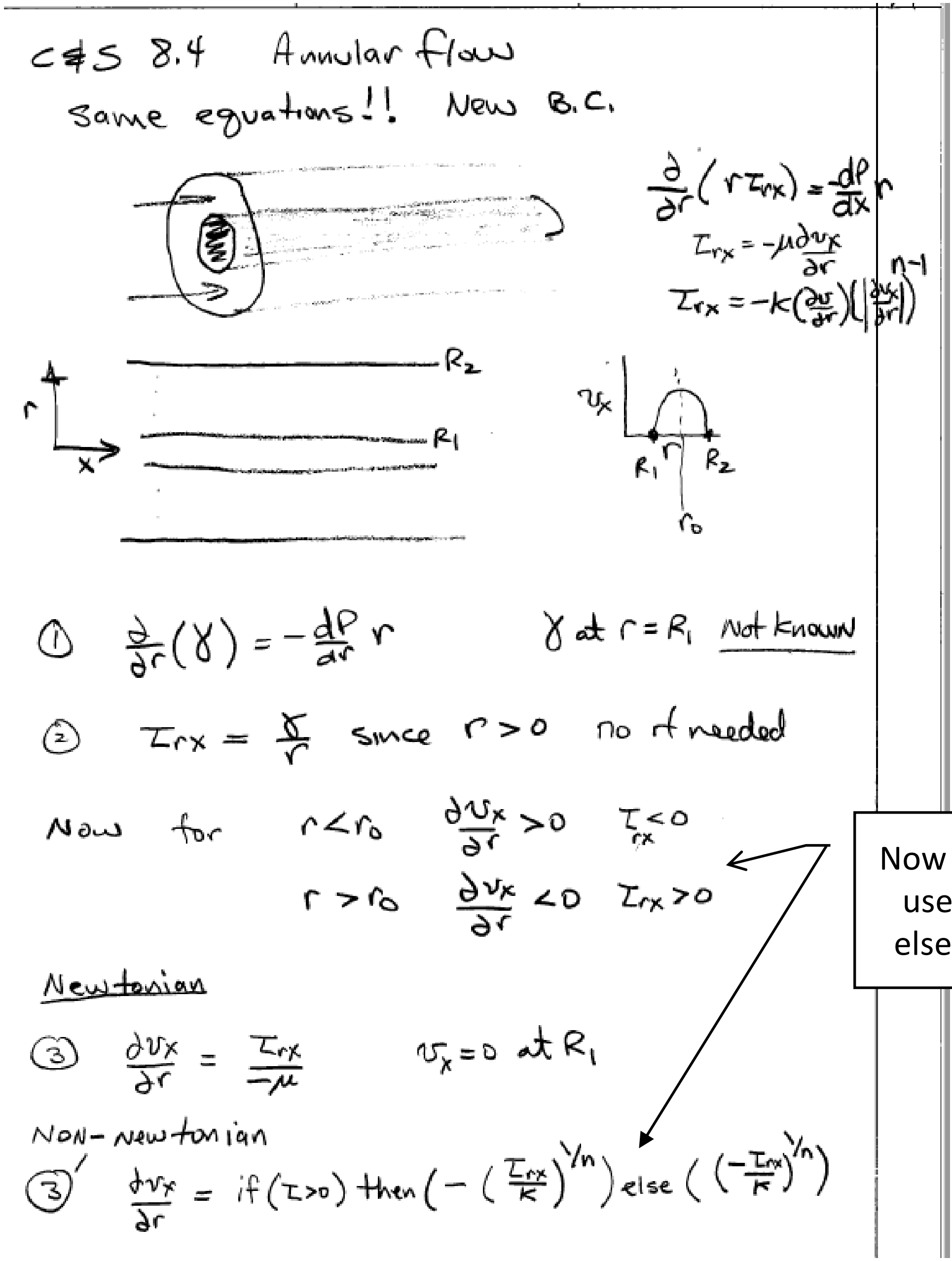
Differential Equations: 3 Auxiliary Equations: 8 ✓ Ready for solution

```
d(tau_r)/d(r) =   delP/L*r
d(vx)/d(r) = if(tau>0) then (-(tau/K)^(1/n)) else ((-tau/K)^(1/n))
d(vavg)/d(r) =  vx*r*2/R^2
K =  1e-6
delP =  100
L =  10
tau =  if (r >0) then (tau_r/r) else (0)
n =  2
vx_anal =  n/(n+1)*(delP/L/2/K)^(1/n)*R^((n+1)/n)*(1-(r/R)^((n+1)/n))
tau_anal=delP/L*r/2
R =  0.009295
r(0)=0
tau_r(0)=0
vx(0)=1.3358812
vavg(0)=0
r(f)=0.009295
```

Ln 1 | C&S8-2ab.pol | 8.2 a&b Laminar Flow of Non-Newtonian Fluids in a Horizontal Pipe

6:44 PM | 7/13/2017 | CAPS | NUM

Annular Flow C&S 8.4



C&S 8.4 Annular flow

Same equations!! New B.C.

$$\frac{\partial}{\partial r}(r\tau_{rx}) = \frac{-dP}{dx} r$$

$$\tau_{rx} = -\mu \frac{dv_x}{dr}$$

$$\tau_{rx} = -k\left(\frac{\partial v}{\partial r}\right)\left|\frac{\partial v_x}{\partial r}\right|^{n-1}$$

① $\frac{\partial}{\partial r}(\gamma) = -\frac{dP}{dr} r$ γ at $r = R_1$ Not known

② $\tau_{rx} = \frac{\gamma}{r}$ since $r > 0$ no if needed

Now for $r < r_0$ $\frac{\partial v_x}{\partial r} > 0$ $\tau_{rx} < 0$

$r > r_0$ $\frac{\partial v_x}{\partial r} < 0$ $\tau_{rx} > 0$

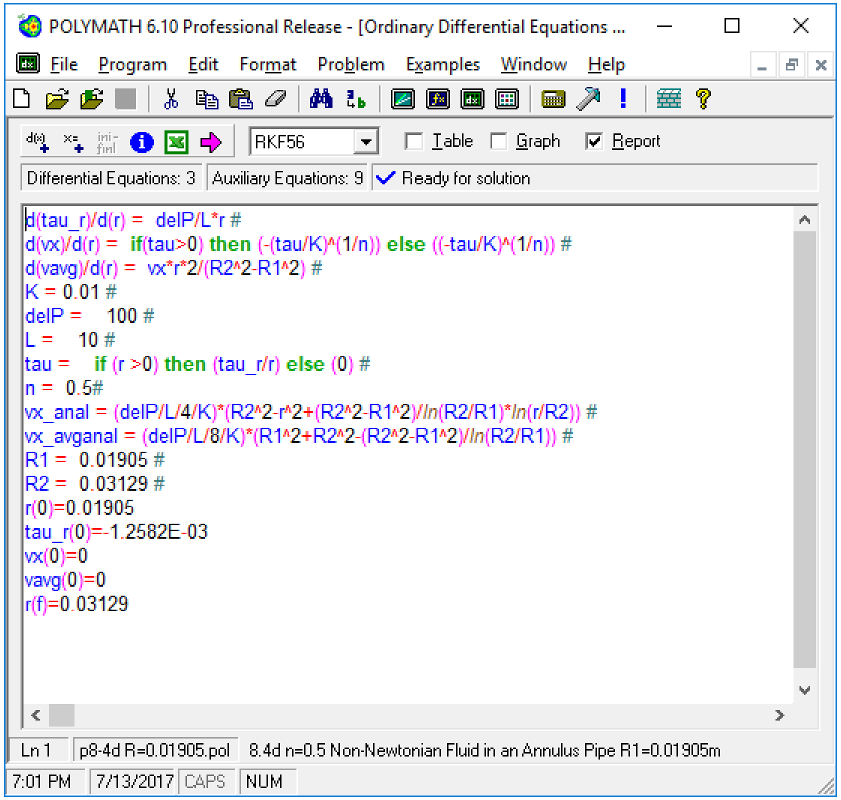
Newtonian

③ $\frac{dv_x}{dr} = \frac{\tau_{rx}}{-\mu}$ $v_x = 0$ at R_1

Non-newtonian

③' $\frac{dv_x}{dr} = \text{if}(\tau>0)\ \text{then}\left(-\left(\frac{\tau_{rx}}{K}\right)^{1/n}\right)\text{else}\left(\left(\frac{-\tau_{rx}}{K}\right)^{1/n}\right)$

Now you have to use an if then else statement



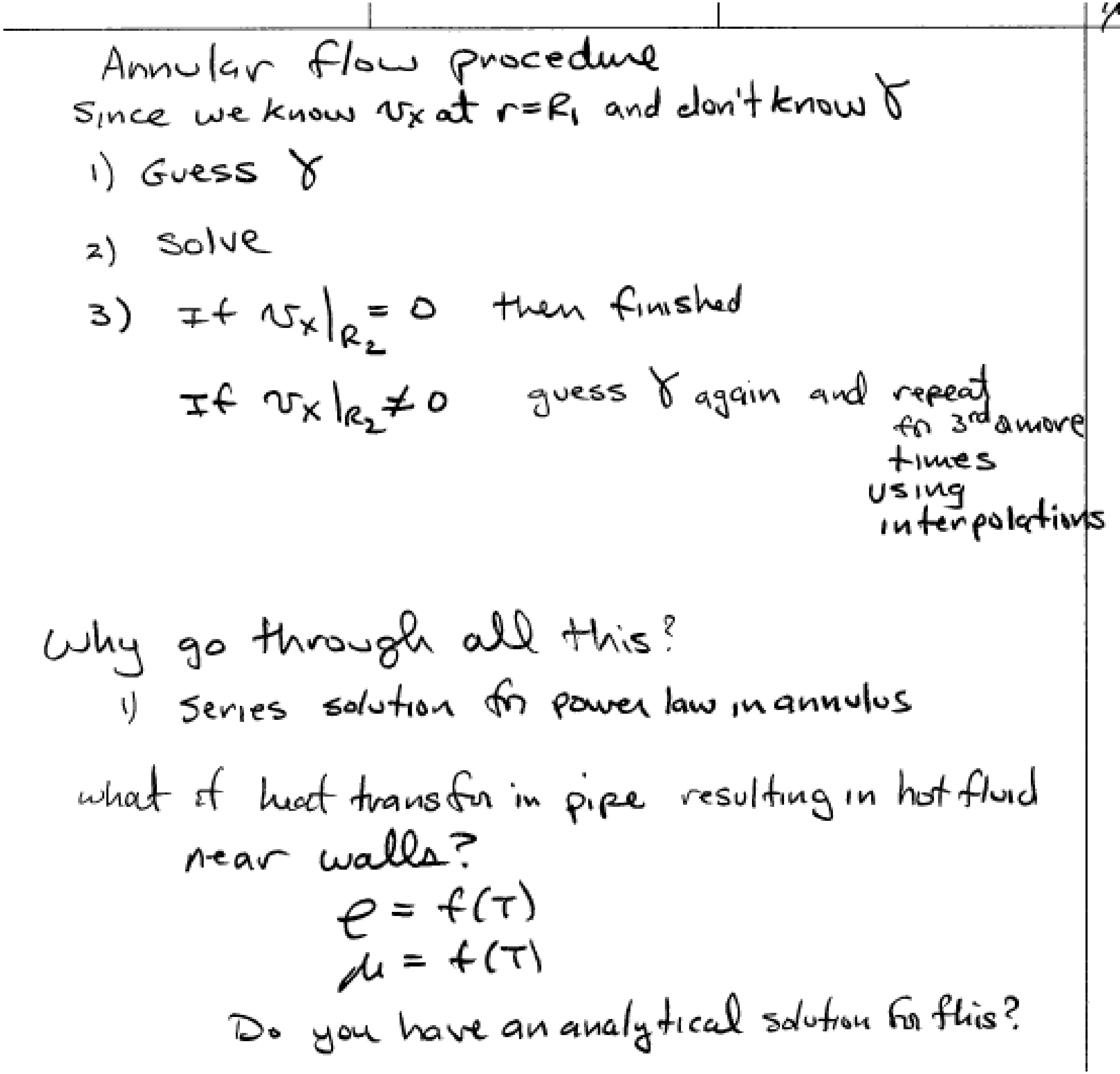
Differential Equations: 3 Auxiliary Equations: 9 ✓ Ready for solution

```
d(tau_r)/d(r) =  delP/L*r #
d(vx)/d(r) =  if(tau>0) then (-(tau/K)^(1/n)) else ((-tau/K)^(1/n)) #
d(vavg)/d(r) =  vx*r*2/(R2^2-R1^2) #
K = 0.01 #
delP =    100 #
L =   10 #
tau =    if (r >0) then (tau_r/r) else (0) #
n =  0.5#
vx_anal = (delP/L/4/K)*(R2^2-r^2+(R2^2-R1^2)/ln(R2/R1)*ln(r/R2)) #
vx_avganal = (delP/L/8/K)*(R1^2+R2^2-(R2^2-R1^2)/ln(R2/R1)) #
R1 =  0.01905 #
R2 =  0.03129 #
r(0)=0.01905
tau_r(0)=-1.2582E-03
vx(0)=0
vavg(0)=0
r(f)=0.03129
```

Ln 1 | p8-4d R=0.01905.pol | 8.4d n=0.5 Non-Newtonian Fluid in an Annulus Pipe R1=0.01905m

7:01 PM | 7/13/2017 | CAPS | NUM

Procedure



Annular flow procedure

Since we know v_x at $r=R_1$ and don't know γ

1) Guess γ
2) Solve
3) If $v_x|_{R_2}=0$ then finished

If $v_x|_{R_2}\neq 0$ guess γ again and repeat fr 3rd & more times using interpolations

Why go through all this?

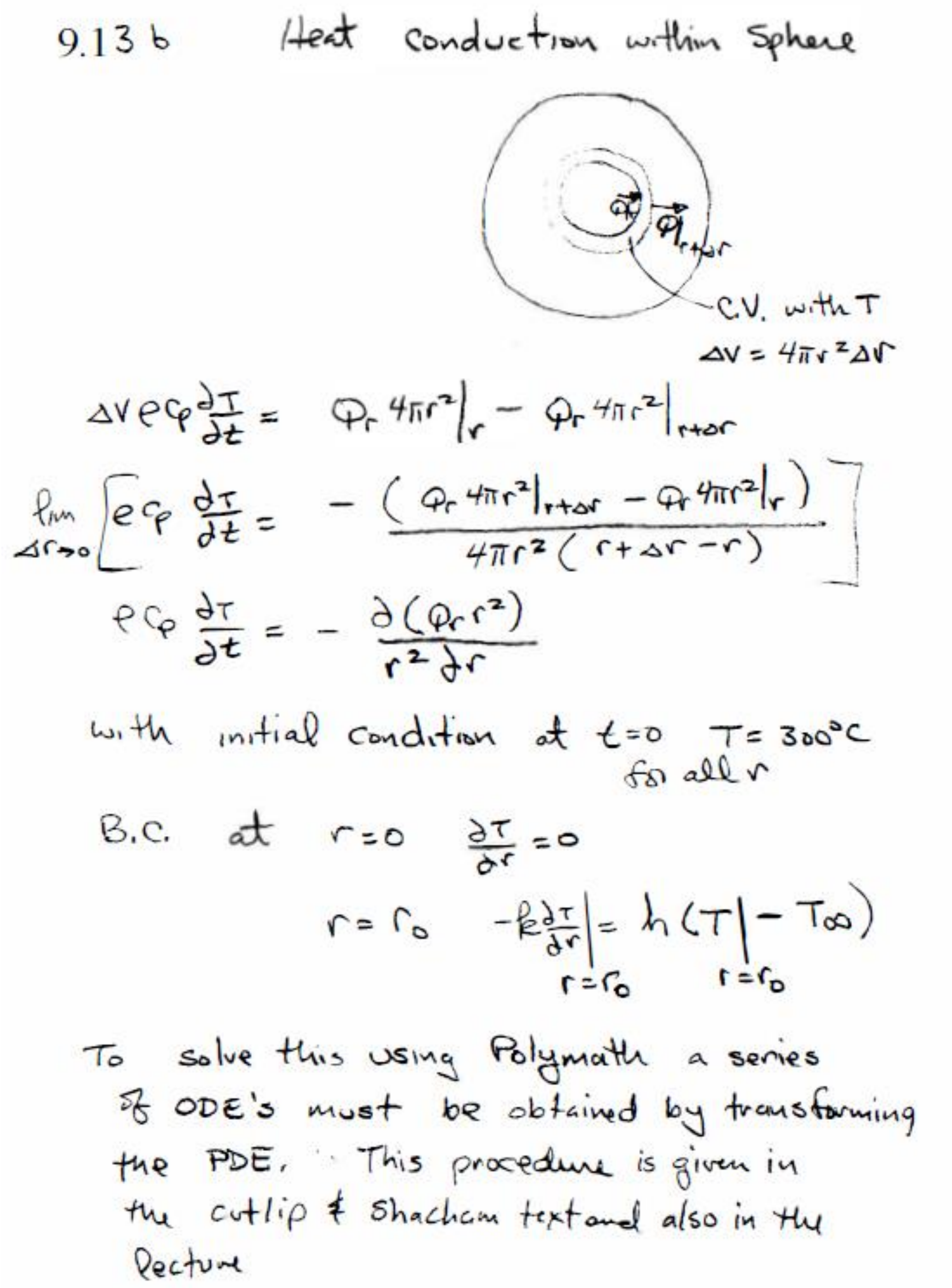
1) Series solution fr power law in annulus

what if heat transfer in pipe resulting in hot fluid near walls?

$$\rho = f(T)$$
$$\mu = f(T)$$

Do you have an analytical solution fr this?

Unsteady-state cooling of a sphere (POLYMATH Text 9.13)



9.13 b Heat conduction within Sphere

q_r $q_r|_{r+\Delta r}$

C.V. with T

$\Delta V = 4\pi r^2 \Delta r$

$$\Delta V \rho c_p \frac{\partial T}{\partial t} = q_r 4\pi r^2 \big|_r - q_r 4\pi r^2 \big|_{r+\Delta r}$$

$$\lim_{\Delta r \to 0} \left[\rho c_p \frac{\partial T}{\partial t} = - \frac{\left(q_r 4\pi r^2 \big|_{r+\Delta r} - q_r 4\pi r^2 \big|_r \right)}{4\pi r^2 (r + \Delta r - r)} \right]$$

$$\rho c_p \frac{\partial T}{\partial t} = - \frac{\partial (q_r r^2)}{r^2 \partial r}$$

with initial condition at $t=0$ $T = 300°C$ for all r

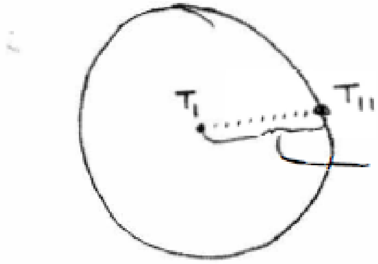
B.C. at $r=0$ $\frac{\partial T}{\partial r} = 0$

$r = r_0$ $-k \frac{\partial T}{\partial r}\Big|_{r=r_0} = h\left(T\big|_{r=r_0} - T_\infty\right)$

To solve this using Polymath a series of ODE's must be obtained by transforming the PDE. This procedure is given in the cutlip & Shacham text and also in the lecture

Transform PDE's using Method of Lines

- Divide into 10 segments (11 lines)
- $\frac{\partial T_n}{\partial t} = \frac{1}{4\pi r^2 \rho C_p}\left(Approximation\,\frac{dq}{dr}\right)$
- Use 2nd-order central difference formula for all but the boundary conditions
- Use forward and backward finite difference for boundary condition



divide into 10 segments

$T|_{r=0} = T_1$

$T|_{r=r_0} = T_{11}$

$\Delta r = \frac{0.05m}{10} = 0.005m$

for $2 \le n \le 10$

$$\frac{\partial T_n}{\partial t} \simeq \text{approximation of}\left(-\frac{1}{r^2}\frac{\partial(Qr^2)}{dr}\right)\frac{1}{\rho c_p}$$

AND

$$4\pi r^2 Q = -k\frac{\partial T}{\partial r}4\pi r^2 = q$$

$$\frac{dT_n}{dt} = -\frac{1}{4\pi r_n^2 \rho c_p}\left[\text{approx}\ \frac{\partial q}{dr}\right]$$

$$\text{appoximation of}\ \frac{dq}{dr} \simeq \frac{q_{n+1} - q_{n-1}}{2\Delta r}$$

$$\therefore\quad \frac{dT_n}{dt} = -\frac{1}{4\pi r_n^2 \rho c_p}\left[\frac{q_{n+1} - q_{n-1}}{2\Delta r}\right]$$

and $r_n = (n-1)(\Delta r)$ for $n=2$ $r_n = \Delta r$

$n=1$ $r_n = 0$

etc.

for $q_n \simeq -k 4\pi r_n^2 \frac{(T_{n+1} - T_{n-1})}{2\Delta r}$

for Boundary conditions:

at $r=0$ $q_1 = 0$

$-k\frac{\partial T}{\partial r}\Big|_{r=0} = 0 =$ using forward difference 2nd order

$$0 = \frac{-T_3 + 4T_2 - 3T_1}{2\Delta r}$$

$$T_1 = \frac{-T_3 + 4T_2}{3}$$

at surface of sphere using backward central difference

$$-k\frac{dT}{dr}\bigg|_{r=r_0} = h(T_{11}-T_\infty)$$

$$-k\left(\frac{3T_{11}-4T_{10}+T_9}{2\Delta r}\right) = h(T_{11}-T_\infty)$$

$$3T_{11}-4T_{10}+T_9 = \frac{2\Delta r h}{k}T_\infty - \frac{2\Delta r h}{k}T_{11}$$

$$T_{11} = \frac{\frac{2\Delta r h}{k}T_\infty - T_9 + 4T_{10}}{3+\frac{2\Delta r h}{k}}$$

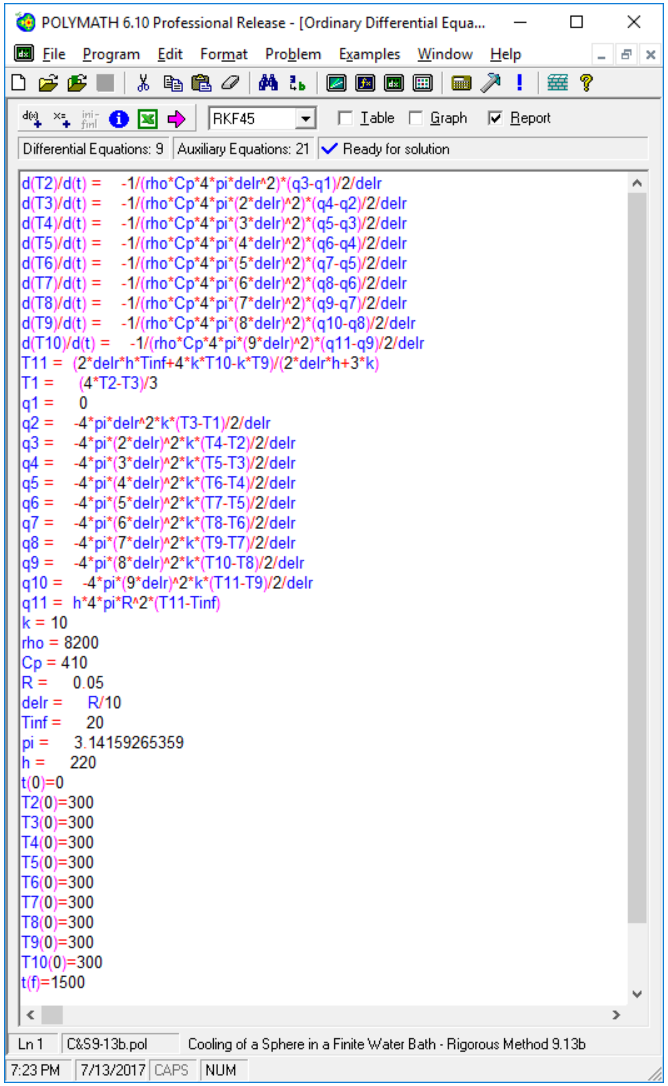
$$T_{11} = \frac{2\Delta r h T_\infty - kT_9 + 4kT_{10}}{3k+2\Delta r h}$$

Now T_2 through T_{10} are ODE's and T_1 and T_{11} are algebraic expressions

finally we must define q_{11}

$$q_{11} = 4\pi r_{11}^2 h(T_{11}-T_\infty)$$

with $r_{11} = (10)(\Delta r)$



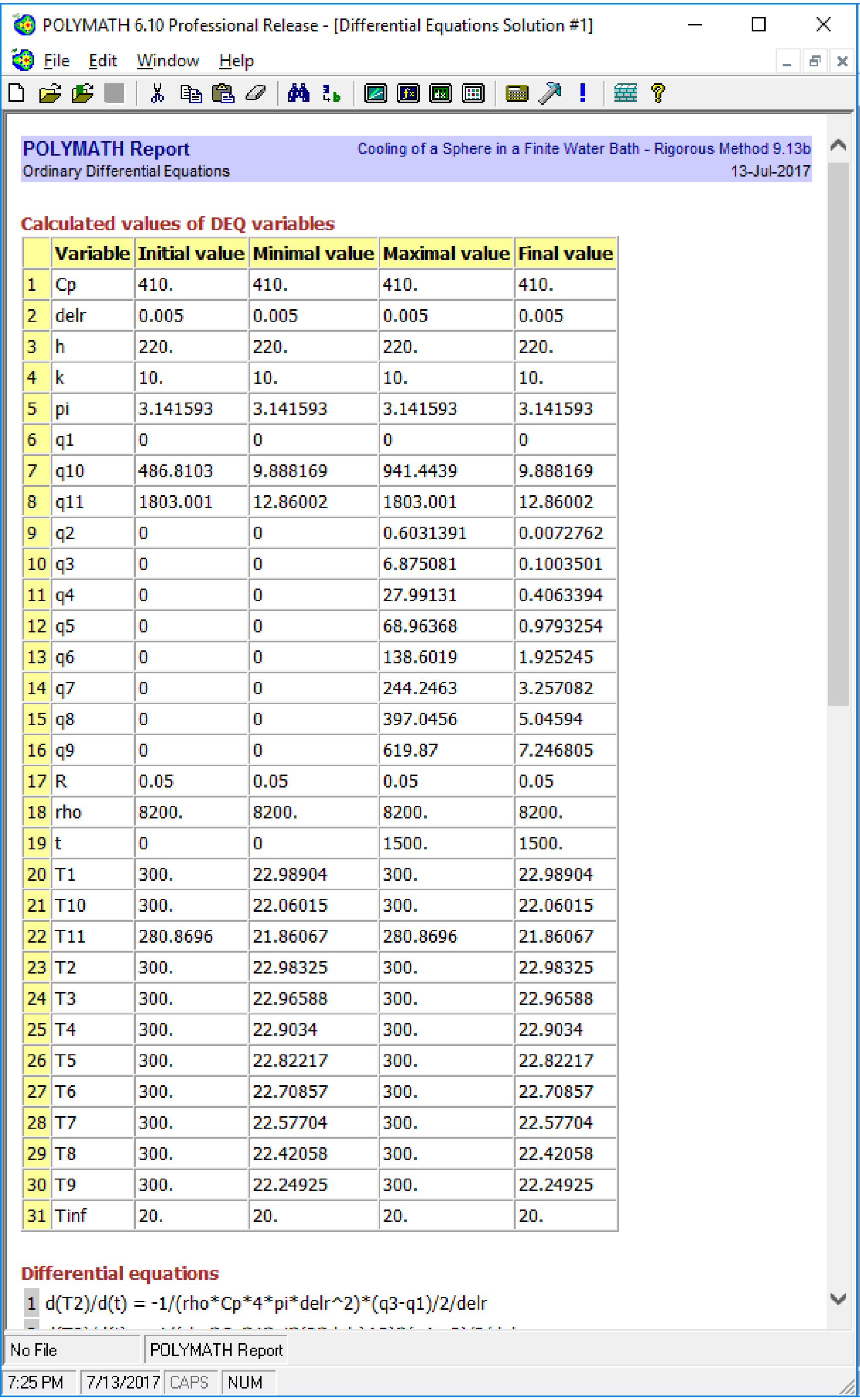
RKF45 ☐ Table ☐ Graph ☑ Report

Differential Equations: 9 | Auxiliary Equations: 21 | ✓ Ready for solution

```
d(T2)/d(t) =    -1/(rho*Cp*4*pi*delr^2)*(q3-q1)/2/delr
d(T3)/d(t) =    -1/(rho*Cp*4*pi*(2*delr)^2)*(q4-q2)/2/delr
d(T4)/d(t) =    -1/(rho*Cp*4*pi*(3*delr)^2)*(q5-q3)/2/delr
d(T5)/d(t) =    -1/(rho*Cp*4*pi*(4*delr)^2)*(q6-q4)/2/delr
d(T6)/d(t) =    -1/(rho*Cp*4*pi*(5*delr)^2)*(q7-q5)/2/delr
d(T7)/d(t) =    -1/(rho*Cp*4*pi*(6*delr)^2)*(q8-q6)/2/delr
d(T8)/d(t) =    -1/(rho*Cp*4*pi*(7*delr)^2)*(q9-q7)/2/delr
d(T9)/d(t) =    -1/(rho*Cp*4*pi*(8*delr)^2)*(q10-q8)/2/delr
d(T10)/d(t) =    -1/(rho*Cp*4*pi*(9*delr)^2)*(q11-q9)/2/delr
T11 =  (2*delr*h*Tinf+4*k*T10-k*T9)/(2*delr*h+3*k)
T1 =     (4*T2-T3)/3
q1 =     0
q2 =    -4*pi*delr^2*k*(T3-T1)/2/delr
q3 =    -4*pi*(2*delr)^2*k*(T4-T2)/2/delr
q4 =    -4*pi*(3*delr)^2*k*(T5-T3)/2/delr
q5 =    -4*pi*(4*delr)^2*k*(T6-T4)/2/delr
q6 =    -4*pi*(5*delr)^2*k*(T7-T5)/2/delr
q7 =    -4*pi*(6*delr)^2*k*(T8-T6)/2/delr
q8 =    -4*pi*(7*delr)^2*k*(T9-T7)/2/delr
q9 =    -4*pi*(8*delr)^2*k*(T10-T8)/2/delr
q10 =    -4*pi*(9*delr)^2*k*(T11-T9)/2/delr
q11 =  h*4*pi*R^2*(T11-Tinf)
k = 10
rho = 8200
Cp = 410
R =     0.05
delr =     R/10
Tinf =     20
pi =     3.14159265359
h =     220
t(0)=0
T2(0)=300
T3(0)=300
T4(0)=300
T5(0)=300
T6(0)=300
T7(0)=300
T8(0)=300
T9(0)=300
T10(0)=300
t(f)=1500
```

Ln 1 | C&S9-13b.pol | Cooling of a Sphere in a Finite Water Bath - Rigorous Method 9.13b

7:23 PM | 7/13/2017 | CAPS | NUM



POLYMATH Report
Ordinary Differential Equations

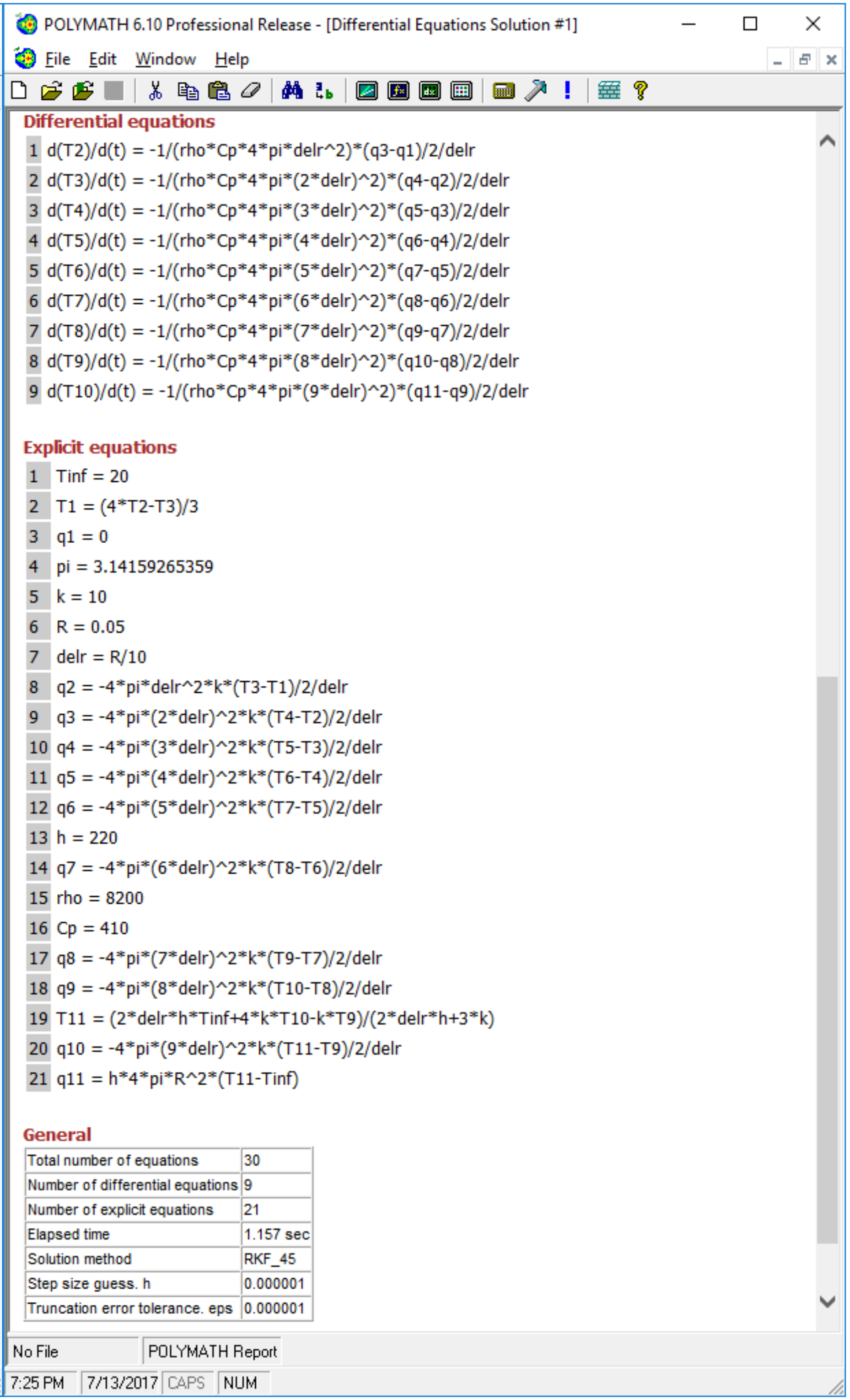
Cooling of a Sphere in a Finite Water Bath - Rigorous Method 9.13b
13-Jul-2017

Calculated values of DEQ variables

	Variable	Initial value	Minimal value	Maximal value	Final value
1	Cp	410.	410.	410.	410.
2	delr	0.005	0.005	0.005	0.005
3	h	220.	220.	220.	220.
4	k	10.	10.	10.	10.
5	pi	3.141593	3.141593	3.141593	3.141593
6	q1	0	0	0	0
7	q10	486.8103	9.888169	941.4439	9.888169
8	q11	1803.001	12.86002	1803.001	12.86002
9	q2	0	0	0.6031391	0.0072762
10	q3	0	0	6.875081	0.1003501
11	q4	0	0	27.99131	0.4063394
12	q5	0	0	68.96368	0.9793254
13	q6	0	0	138.6019	1.925245
14	q7	0	0	244.2463	3.257082
15	q8	0	0	397.0456	5.04594
16	q9	0	0	619.87	7.246805
17	R	0.05	0.05	0.05	0.05
18	rho	8200.	8200.	8200.	8200.
19	t	0	0	1500.	1500.
20	T1	300.	22.98904	300.	22.98904
21	T10	300.	22.06015	300.	22.06015
22	T11	280.8696	21.86067	280.8696	21.86067
23	T2	300.	22.98325	300.	22.98325
24	T3	300.	22.96588	300.	22.96588
25	T4	300.	22.9034	300.	22.9034
26	T5	300.	22.82217	300.	22.82217
27	T6	300.	22.70857	300.	22.70857
28	T7	300.	22.57704	300.	22.57704
29	T8	300.	22.42058	300.	22.42058
30	T9	300.	22.24925	300.	22.24925
31	Tinf	20.	20.	20.	20.

Differential equations

```
1 d(T2)/d(t) = -1/(rho*Cp*4*pi*delr^2)*(q3-q1)/2/delr
```



Differential equations

```
1 d(T2)/d(t) = -1/(rho*Cp*4*pi*delr^2)*(q3-q1)/2/delr
2 d(T3)/d(t) = -1/(rho*Cp*4*pi*(2*delr)^2)*(q4-q2)/2/delr
3 d(T4)/d(t) = -1/(rho*Cp*4*pi*(3*delr)^2)*(q5-q3)/2/delr
4 d(T5)/d(t) = -1/(rho*Cp*4*pi*(4*delr)^2)*(q6-q4)/2/delr
5 d(T6)/d(t) = -1/(rho*Cp*4*pi*(5*delr)^2)*(q7-q5)/2/delr
6 d(T7)/d(t) = -1/(rho*Cp*4*pi*(6*delr)^2)*(q8-q6)/2/delr
7 d(T8)/d(t) = -1/(rho*Cp*4*pi*(7*delr)^2)*(q9-q7)/2/delr
8 d(T9)/d(t) = -1/(rho*Cp*4*pi*(8*delr)^2)*(q10-q8)/2/delr
9 d(T10)/d(t) = -1/(rho*Cp*4*pi*(9*delr)^2)*(q11-q9)/2/delr
```

Explicit equations

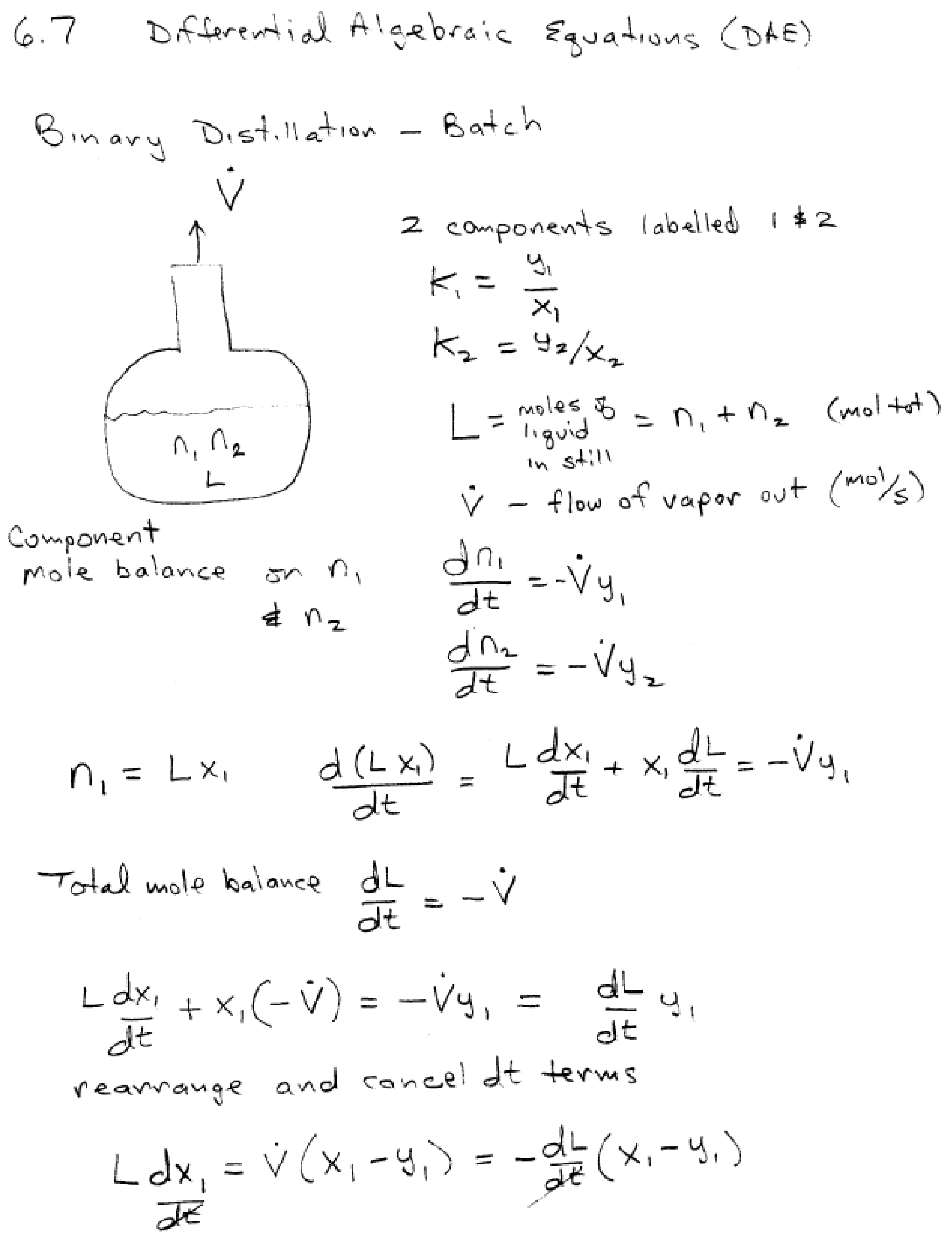
```
1  Tinf = 20
2  T1 = (4*T2-T3)/3
3  q1 = 0
4  pi = 3.14159265359
5  k = 10
6  R = 0.05
7  delr = R/10
8  q2 = -4*pi*delr^2*k*(T3-T1)/2/delr
9  q3 = -4*pi*(2*delr)^2*k*(T4-T2)/2/delr
10 q4 = -4*pi*(3*delr)^2*k*(T5-T3)/2/delr
11 q5 = -4*pi*(4*delr)^2*k*(T6-T4)/2/delr
12 q6 = -4*pi*(5*delr)^2*k*(T7-T5)/2/delr
13 h = 220
14 q7 = -4*pi*(6*delr)^2*k*(T8-T6)/2/delr
15 rho = 8200
16 Cp = 410
17 q8 = -4*pi*(7*delr)^2*k*(T9-T7)/2/delr
18 q9 = -4*pi*(8*delr)^2*k*(T10-T8)/2/delr
19 T11 = (2*delr*h*Tinf+4*k*T10-k*T9)/(2*delr*h+3*k)
20 q10 = -4*pi*(9*delr)^2*k*(T11-T9)/2/delr
21 q11 = h*4*pi*R^2*(T11-Tinf)
```

General

Total number of equations	30
Number of differential equations	9
Number of explicit equations	21
Elapsed time	1.157 sec
Solution method	RKF_45
Step size guess. h	0.000001
Truncation error tolerance. eps	0.000001

Differential-algebraic system of equations using the controlled integration technique

- $\frac{dn_1}{dt} = -\dot{n}_1 = \dot{n}_T y_1$
- $\frac{dn_T}{dt} = -\dot{n}_T$
- $y_1 = K_1 x_1$
- $error = 1 - K_1 x_1 - K_2 x_2$
- $\frac{dT}{dx_1} = K_c(error)$



6.7 Differential Algebraic Equations (DAE)

Binary Distillation – Batch

$\dot{V}$

$n_1\ n_2$

L

2 components labelled 1 & 2

$K_1 = \frac{y_1}{x_1}$

$K_2 = y_2/x_2$

$L =$ moles of liquid in still $= n_1 + n_2$ (mol tot)

$\dot{V}$ – flow of vapor out (mol/s)

Component mole balance on n_1 & n_2

$$\frac{dn_1}{dt} = -\dot{V} y_1$$

$$\frac{dn_2}{dt} = -\dot{V} y_2$$

$$n_1 = L x_1 \qquad \frac{d(L x_1)}{dt} = L\frac{dx_1}{dt} + x_1\frac{dL}{dt} = -\dot{V} y_1$$

Total mole balance $\frac{dL}{dt} = -\dot{V}$

$$L\frac{dx_1}{dt} + x_1(-\dot{V}) = -\dot{V} y_1 = \frac{dL}{dt} y_1$$

rearrange and cancel dt terms

$$L\frac{dx_1}{\cancel{dt}} = \dot{V}(x_1 - y_1) = -\frac{dL}{\cancel{dt}}(x_1 - y_1)$$

$$L\,dx_1 = dL\,(y_1 - x_1)$$

$$y_1 = K_1 x_1$$

$$L\,dx_1 = dL(K_1 x_1 - x_1)$$

$$\frac{dL}{dx_1} = \frac{L}{x_1(K_1 - 1)}$$

or similarly

$$\frac{dL}{dx_2} = \frac{L}{x_2(K_2 - 1)}$$

for an ideal system $y_i P = x_i \gamma_i p^{vap}$ with $\gamma_i = 1$

$$\frac{y_i}{x_i} = \frac{p_i^{vap}}{P} = K_i$$

P – total pressure

p_i^{vap} – vapor pressure of i

$$p_i^{vap} = 10^{\left(A + \frac{B}{T+C}\right)}$$

for T in °C p_i^{vap} = mm Hg

	A	B	C
Benzene :	6.90565	−1211.033	220.79
Toluene	6.95464	−1344.8	219.482

$P = 1.2$ atm

L at $t=0$ 100 mol

$x_B = 0.60$

$x_T = 0.40$

find L at $x_T = 0.80$

start integrating at $x_T = 0.4$ and stop at $x_T = 0.8$

$$\frac{dL}{dx_T} = \frac{L}{x_T(K_T - 1)}$$

L at $x_T = 0.40$ is 100 mol

$$K_T = \frac{p_T^{vap}}{P}$$

$$P = \left(\frac{760\text{ mm Hg}}{1\text{ atm}}\right) 1.2\text{ atm}$$

$$p_T = 10^\wedge\left(6.95464 + \frac{-1344.8}{T + 219.482}\right)$$

Need Temperature

the removal of vapor would start at the bubble point of the mixture.

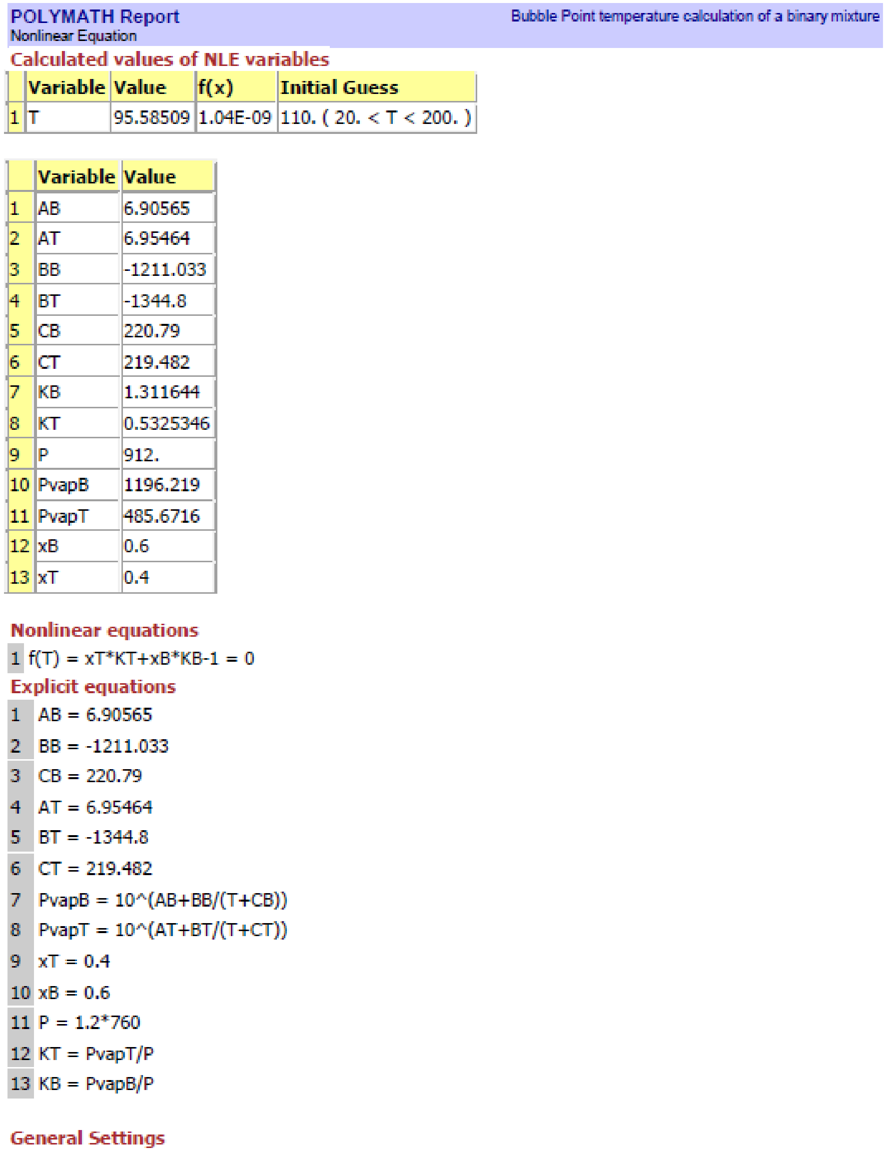
$$\Sigma y_i = 1$$

$$y_T + y_B = 1$$

$$K_T x_T + K_B x_B = 1$$

Solving using non-linear equation solver to find T at initial bubble point. T = 95.58509°C

Determination of starting temperature of batch distillation



POLYMATH Report Bubble Point temperature calculation of a binary mixture

Nonlinear Equation

Calculated values of NLE variables

	Variable	Value	f(x)	Initial Guess
1	T	95.58509	1.04E-09	110. (20. < T < 200.)

	Variable	Value
1	AB	6.90565
2	AT	6.95464
3	BB	-1211.033
4	BT	-1344.8
5	CB	220.79
6	CT	219.482
7	KB	1.311644
8	KT	0.5325346
9	P	912.
10	PvapB	1196.219
11	PvapT	485.6716
12	xB	0.6
13	xT	0.4

Nonlinear equations

1 f(T) = xT*KT+xB*KB-1 = 0

Explicit equations

1 AB = 6.90565
2 BB = -1211.033
3 CB = 220.79
4 AT = 6.95464
5 BT = -1344.8
6 CT = 219.482
7 PvapB = 10^(AB+BB/(T+CB))
8 PvapT = 10^(AT+BT/(T+CT))
9 xT = 0.4
10 xB = 0.6
11 P = 1.2*760
12 KT = PvapT/P
13 KB = PvapB/P

General Settings

Approach I Controlled Integration
from Shacham

$$\text{Error} = 1 - K_1X_1 - K_2x_2$$

goal is to have error sufficiently small

Use a proportional controller to change T to match the bubble point T.

$$\frac{dT}{dX_2} = K_c(\text{error})$$

choose K_c to keep error less than a tolerance value.

$K_c = 1$ $\text{error}_{max} = .0.31$ too high

$K_c = 1000$ error max $= 0.037$ too high

$K_c = 1e^4$ error max $= 0.004$ close

$K_c = 1e^5$ error max $= 0.0004$ good

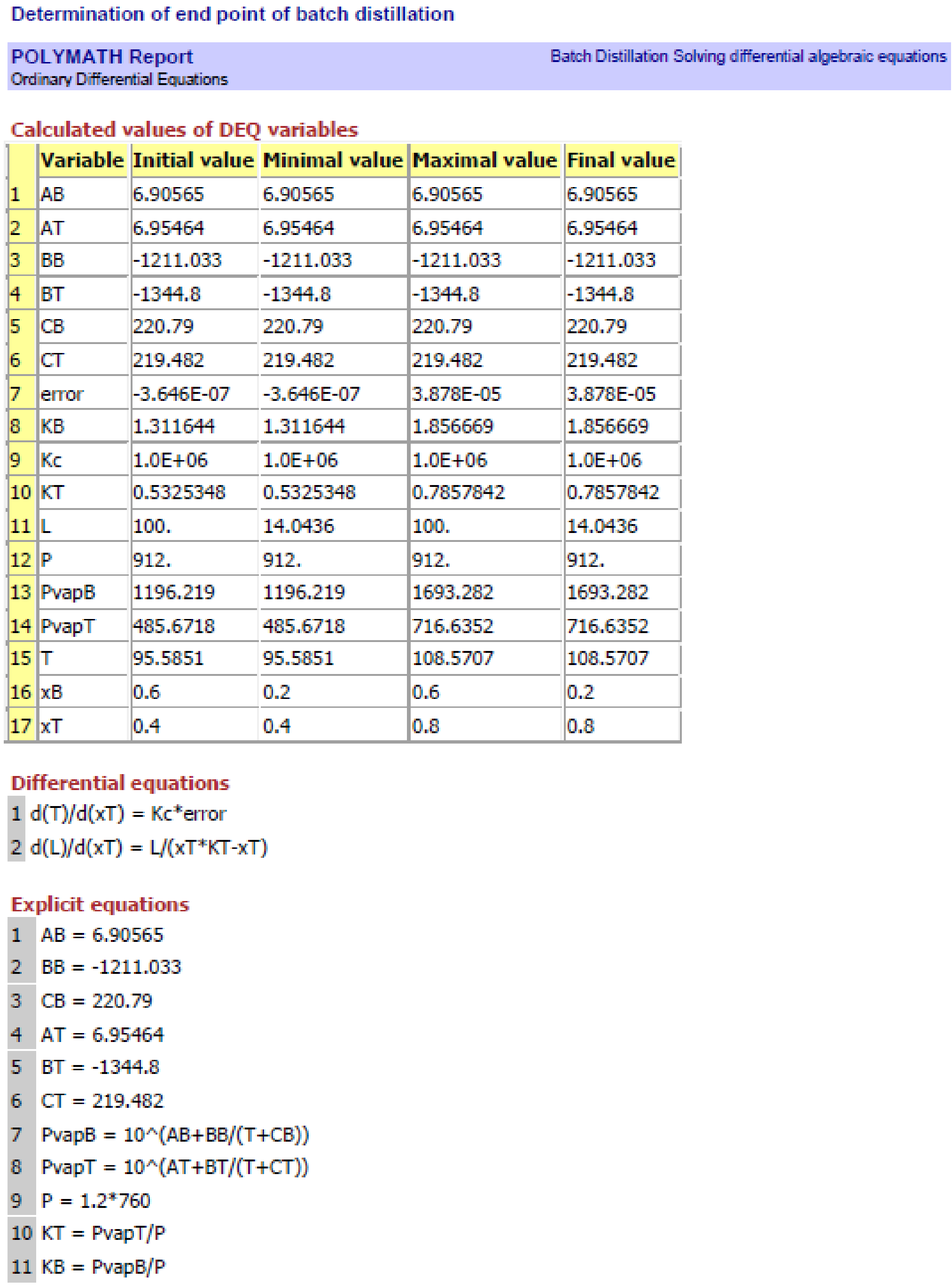
$K_c = 1e^6$ max $= 4\times10^{-5}$ excellent

at end of batch distillation

$X_B = 0.2$

$L = 14.0436$ mol

$T = 108.6°C$



Determination of end point of batch distillation

POLYMATH Report
Ordinary Differential Equations

Batch Distillation Solving differential algebraic equations

Calculated values of DEQ variables

	Variable	Initial value	Minimal value	Maximal value	Final value
1	AB	6.90565	6.90565	6.90565	6.90565
2	AT	6.95464	6.95464	6.95464	6.95464
3	BB	-1211.033	-1211.033	-1211.033	-1211.033
4	BT	-1344.8	-1344.8	-1344.8	-1344.8
5	CB	220.79	220.79	220.79	220.79
6	CT	219.482	219.482	219.482	219.482
7	error	-3.646E-07	-3.646E-07	3.878E-05	3.878E-05
8	KB	1.311644	1.311644	1.856669	1.856669
9	Kc	1.0E+06	1.0E+06	1.0E+06	1.0E+06
10	KT	0.5325348	0.5325348	0.7857842	0.7857842
11	L	100.	14.0436	100.	14.0436
12	P	912.	912.	912.	912.
13	PvapB	1196.219	1196.219	1693.282	1693.282
14	PvapT	485.6718	485.6718	716.6352	716.6352
15	T	95.5851	95.5851	108.5707	108.5707
16	xB	0.6	0.2	0.6	0.2
17	xT	0.4	0.4	0.8	0.8

Differential equations

1 d(T)/d(xT) = Kc*error
2 d(L)/d(xT) = L/(xT*KT-xT)

Explicit equations

1 AB = 6.90565
2 BB = -1211.033
3 CB = 220.79
4 AT = 6.95464
5 BT = -1344.8
6 CT = 219.482
7 PvapB = 10^(AB+BB/(T+CB))
8 PvapT = 10^(AT+BT/(T+CT))
9 P = 1.2*760
10 KT = PvapT/P
11 KB = PvapB/P

Thank you for your interest in this workshop!