The Computing Science and Technology International Journal
ISSN 2162-0660
Vol. 12 Issue 1 February 2023

APPROXIMATE SOLUTION OF DIFFERENTIAL EQUATIONS BY EULER AND RUNGE-KUTTA METHODS

Mamatov Abdugani Ermamatovich

Rakhimova Feruza Saidovna

Professors department "Advanced mathematics" Tashkent university information technologies named after Muhammad al-Kwharazmy

Since differential equations form the basis of mathematical modeling of processes in chemistry, biology, medicine, sociology, construction, architecture, agriculture and economy, all specialists are required to have the skills of solving and researching differential equations.

In the process of learning the course of differential equations, we considered the methods of solving differential equations with special forms. These methods cannot cover many other situations. That is why the search for universal methods that do not depend on the form of the equation. The development of computing machines has made it possible to successfully apply almost any number of methods.

1. Euler's method

Let's start with the Cauchy problem for first-order differential equations. Let's say

$$y' = f(x, y), \qquad x_0 \leq x \leq b \qquad (1)$$

differential equation of the form

$$y(x_0) = y_0 \qquad (2)$$

let the problem of finding a solution satisfying the initial condition, i.e. the Cauchy problem, be given. In general, the Cauchy problem cannot be solved. $f(x, y)$ there are ways to find the general solution of (1) only in certain forms of the function. In many cases, the methods of approximate solution of differential equations are used in practical problems. We assume that the conditions of the theorem on the existence and uniqueness of the solution are fulfilled. $M_0(x_0, y_0)$ around the point $f(x, y)$ function x continuous on, y and satisfy the condition for Lifshitz.

(1)-(2) Solution of the Cauchy problem y(x) to x_0 around the point we spread the Taylor series:

$$y(x) = y_0 + (x - x_0)y'(x_0) + \frac{(x - x_0)^2}{2!}y''(x_0) +$$

$$+\frac{(x-x_0)^3}{3!}y'''(x_0)+\frac{(x-x_0)^4}{4!}y^{IV}(x_0)+\cdots+\cdots \quad (3)$$

x_0 around the small point, we take the first two terms of the Taylor series and discard the remaining terms, as a result, we arrive at the following approximate formula

$$y(x)\approx y_0+(x-x_0)y'. \quad (4)$$

If we use y' the form in formula (1), then formula (4) can be written as follows:

$$y(x)\approx y_0+(x-x_0)f(x_0,y_0) \quad (5)$$

(5)- the formula $x_0\le x\le b$ to generalize to an interval, we divide this interval into n parts. Partitioning step:

$$h=\frac{b-x_0}{n}\ ;\quad x_i=x_0+ih,\ \ i=0,1,2,\dots,n.$$

The solution to the problem x_i we aim to find it in tabular form at points. Approx $y(x_i)$ values (5) we find by the formula:

$$y_{i+1}\approx y_i+h*f(x_i,y_i)\quad i=0,1,2,\dots,n-1 \quad (6)$$

in this $y_{i+1}=y(x_{i+1}),\ y_i=y(x_i).$ This formula is called Euler's method. Euler's method is a universal method, f(x,y) does not depend on the appearance of , but the error is relatively large. Error at every step $O(h^2)$ is in order, and this error increases step by step until it reaches point b. $O(h)$ can increase up to In the coordinate plane $(x_0,y_0);\ (x_1,y_1);\dots;\ (x_n,y_n)$ broken line of an integral curve formed by connecting points with straight line segments.

Note: $y=\varphi_h(x)$ with (1)- equation $\Delta x=h$ we determine the approximate solution corresponding to the Euler fracture line in If the initial conditions of equation (1) are satisfied and $[x_0;b]$ the only one identified in cross-section $y=\varphi^*(x)$ if a solution exists, then $[x_0;b]$any in cross section x for

$$\lim_{h\to 0}|\varphi_h(x)-\varphi^*(x)|=0$$

can be proved.

Euler's method is convenient for programming. Based on the formula (6), arbitrary Cauchy problem (1)-(2) can be solved with any previously given accuracy. To increase the accuracy, it is enough to increase the number of steps n. For this we use the following relations:

$$h=\frac{b-x_0}{n}\ ;\quad R=O(h^2)=\varepsilon;\quad \left(\frac{b-x_0}{n}\right)^2=\varepsilon$$

$$n \approx \frac{b-x_0}{\sqrt{\varepsilon}} \tag{7}$$

Runge-Kutta method

During the study of this course, it was noted that the differential equations section of mathematics originated from practical problems. Therefore, the creation of universal methods similar to formula (6) has been a constant problem. Representatives of various fields also tried to solve this problem. A clear example of this is the Runge-Kutta method, which was created by scientists, one physicist and one astronomer. Line (3) served as the basis for the creation of this method. Unlike Euler's method (6), they used five terms instead of two. In addition, the Runge-Kutta method proposed formulas that do not require the calculation of the derivatives included in row (3). Without dwelling on the theoretical origin of these formulas, we will dwell on working formulas. The division step is the same as in Euler's method $h = \frac{b-x_0}{n}$ determined by the formula.

The formulas for finding the values of the function are as follows:

$$\begin{cases} K_1 = h \cdot f(x_i, y_i) \\ K_2 = h \cdot f\left(x_i + \frac{h}{2}, y_i + \frac{K_1}{2}\right) \\ K_3 = h \cdot f\left(x_i + \frac{h}{2}, y_i + \frac{K_2}{2}\right) \\ K_4 = h \cdot f(x_i + h, y_i + K_3) \\ y_{i+1} = y_i + \frac{1}{6}(K_1 + 2K_2 + 2K_3 + K_4)\,; \end{cases} \tag{8}$$

$i = 0,1,2,\ldots,n-1$.

It can be seen that the Runge-Kutta method formally increases the calculation volume by 5 times compared to the Euler method at each step. The error of the Runge-Kutta method at each step $O(h^5)$ will be in order. It is for this reason that the Runge-Kutta method has become the main method for the approximate solution of the Cauchy problem for first-order differential equations.

It should be noted that implementation programs for a large number of methods have been created and these programs are available in modern computer operating systems. Applying for these programs is simplified. Differential equations can be solved using mathematical packages (WolframAlpha, Matlab, Mathcad). WolframAlpha is famous for its ability to solve very complex equations in mathematics and physics. The right-hand side of the differential equation is f(x,y), the initial condition x_0, y_0 and ε it is enough to give precision.

An example. $\boldsymbol{y' = 3x + 0,1y \quad , \quad y(0) = 0,4}$

Cauchy's problem on the interval (0;1) $\varepsilon = 0{,}0001$ find the approximate values of the solution in steps h=0.2.

Solving. 1) Analytical solution:

$$y' = 3x + 0{,}1y$$

$y' - 0{,}1y = 3x$ this is a linear equation

p(x) =-0,1, q(x) =3x .

$$y = e^{-\int p\,dx}\{c_1 + \int q(x)e^{\int (P(x))\,dx}dx\} =$$

$$= e^{0{,}1x}\{C_1 + \int 3xe^{-0{,}1^x}dx\}_= \begin{bmatrix} u = x & ; & e^{-0{,}1x}dx = dv \\ du = dx\,; & & v = -10e^{-0{,}1x} \end{bmatrix}_=$$

$$= e^{0{,}1x}\{c_1 + 3[-10xe^{-0{,}1x} + 10\int e^{-0{,}1x}dx\,]\}_=$$

$$= e^{0{,}1x}\{C_1 + (-30x - 300)e^{-0{,}1x}\} = C_1e^{0{,}1x} - 30x - 300.$$

General solution. $y = C_1e^{0{,}1x} - 30x - 300$.

Private solution: $y(0) = c_1 - 300 = 0{,}4, \quad c_1 = 300{,}4$.

$$y = 300{,}4e^{0{,}1x} - 30x - 300.$$

2) Solving by Euler's method:

Since h=0.2, we divide the interval [0, 1] into 5 parts:

$$x_0 = 0,\ x_1 = 0.2\,,\ x_2 = 0.4,\ x_3 = 0.6,\ x_4 = 0.8,\ x_5 = 1.$$

By initial condition $y(0) = 0.4,\ x_0 = 0,\ y_0 = 0.4$.

We make calculations according to formula (6).

$$y_1 = y_0 + h \cdot f(x_0; y_0) = 0{,}4 + 0{,}2 * (3 \cdot 0 + 0{,}1 \cdot 0{,}4)_=$$
$$0{,}4 + 0{,}2 \cdot 0{,}04 = 0{,}408 \approx 0{,}41$$

$$y_2 = y_2 + h \cdot f(x_1; y_1) = 0{,}41 + 0{,}2(3 * 0{,}2 + 0{,}1 \cdot 0{,}41)$$
$$= 0{,}41 + 0{,}2(0{,}6 + 0{,}041) = 0{,}41 + 0{,}2 \cdot 0{,}641 = 0{,}5382 \approx 0{,}54$$

$$y_3 = y_2 + h \cdot f(x_2; y_2) = 0{,}54 + 0{,}2 * (3 \cdot 0{,}4 + 0{,}1 \cdot 0{,}54)_=$$
$$0{,}54 + 0{,}2(1{,}2 + 0{,}054) = 0{,}54 + 0{,}2 \cdot 1{,}254 = 0{,}54 + 0{,}2508 \approx 0{,}79$$

$$y_4 = y_3 + h \cdot f(x_3; y_3) = 0{,}79 + 0{,}2(3 * 0{,}6 + 0{,}1 \cdot 0{,}79)$$
$$= 0{,}7 + 0{,}2(1{,}8 + 0{,}079) = 0{,}79 + 0{,}2 \cdot 0{,}3758 \approx 1{,}16$$

$$y_5 = y_4 + h \cdot f(x_4; y_4) = 1{,}16 + 0{,}2(3 * 0{,}8 + 0{,}1 \cdot 1{,}16)$$
$$= 1{,}16 + 0{,}2(2{,}4 + 0{,}116) = 1{,}16 + 0{,}5032 = 1{,}6632 \approx 1{,}66$$

x_i	0	0,2	0,4	0,6	0,8	1
y_i	0,4	0,41	0,54	0,79	1,16	1,66

3) Solving by Runge-Kutta method:

count when: $x_1 = 0,2$

$$\begin{Bmatrix} K_1 = 0,800 \\ K_2 = 0,080 \\ K_3 = 0,091 \\ K_4 = 0,1062 \end{Bmatrix} y_1 = 0,4913$$

count when: $x_2 = 0,4$

$$\begin{Bmatrix} K_1 = 0,10626 \\ K_2 = 0,1268 \\ K_3 = 0,1289 \\ K_4 = 0,1560 \end{Bmatrix} y_2 = 0,6202$$

count when: $x_3 = 0,6$

$$\begin{Bmatrix} K_1 = 0,15504 \\ K_2 = 0,1896 \\ K_3 = 0,183 \\ K_4 = 0,2346 \end{Bmatrix} y_3 = 0,8126$$

count when: $x_4 = 0,8$

$$\begin{Bmatrix} K_1 = 0,23456 \\ K_2 = 0,2840 \\ K_3 = 0,2389 \\ K_4 = 0,3483 \end{Bmatrix} y_4 = 1,1009$$

count when: $x_5 = 1$

$$\begin{Bmatrix} K_1 = 0,34818 \\ K_2 = 0,4169 \\ K_3 = 0,4238 \\ K_4 = 0,5043 \end{Bmatrix} y_5 = 1,5233$$

.

x	0	0,2	0,4	0,6	0,8	1
y	0,4	0,4913	0,6202	0,8128	1,1009	1,5233

Reference

1. Lapchik MP Numerical methods: textbook. allowance for students. universities / M.P. Lapchik, M.I. Ragulin, E.K. Henner.— M.: Academy, 2005.- 384 p.

2. Ivanov V.M. Numerical methods / V. M. Ivanov. - Yekaterinburg: Publishing house Ural. state economy un-ta, 2003. - 114 p.

3. Amosov A.A. Computational methods for engineers: textbook. allowance / A.A. Amosov, Yu.A. Dubinsky, N.V. Kopchenova. - M .: Higher. school., 1994. - 544 p.

4. Tikhonov A.N. Introductory lectures on applied mathematics / A.N. Tikhonov, D.P. Kostomarov. - M .: Nauka, 1984. - 192 p.

5. Volkov E.A. Numerical methods / E.A. Volkov.- M. : Nauka, 1982.-256 s

6. FS Rakhimova, KU Namozovich "On Teaching Students Newton's Method Of Solving One-Dimensional Optimization Problems" Central Asian Journal Of Mathematical Theory And Computer Sciences 4 (1), 2023, 57-60,

7. F Rakhimova Model of Developing Communicative and Didactic Competence among Future Teachers Eastern European Scientific Journal 2019

8. RF Saidovna, PN Abduvakhabovna, TS Ergashevna ALGORITHM OF CALCULATION OF FACTORS IN PIECEWISE-QUADRATIC WALSH'S BASES PEDAGOGS joornal 11 (4), 2022, 187-191

9. OA Islamova, ZS Chay, FS Rakhimova, FS Abdullayeva Calculate Central Limit Theorem for the Number of Empty Cells after Allocation of Particles

10. Rakhimov BS, Mekhmanov MS, Bekchanov BG. Parallel algorithms for the creation of medical database. J Phys Conf Ser. 2021;1889(2):022090. doi:10.1088/1742-6596/1889/2/022090

11. Rakhimov BS, Rakhimova FB, Sobirova SK. Modeling database management systems in medicine. J Phys Conf Ser. 2021;1889(2):022028. doi:10.1088/1742-6596/1889/2/022028

12. Rakhimov B, Ismoilov O. Management systems for modeling medical database. In: ; 2022:060031. doi:10.1063/5.0089711

13. Rakhimov BS, Khalikova GT, Allaberganov OR, Saidov AB. Overview of graphic processor architectures in data base problems. In: ; 2022:020041. doi:10.1063/5.0092848

14. Allaberganov, O.R., Rakhimov, B.S., Sobirova, S.K., Rakhimova, F.B., Saidov, A.B. Problem for Medical System with Infinite Zone Potential in the Half Line AIP Conference Proceedings, 2022, 2647, 050025

15. RB Saidovich, SA Bakhtiyarovich, BB Farkhodovich, KDA Ugli, MMZ Qizi Analysis And Using of the Features Graphics Processors for Medical Problems Texas Journal of Medical Science 7, 105-110

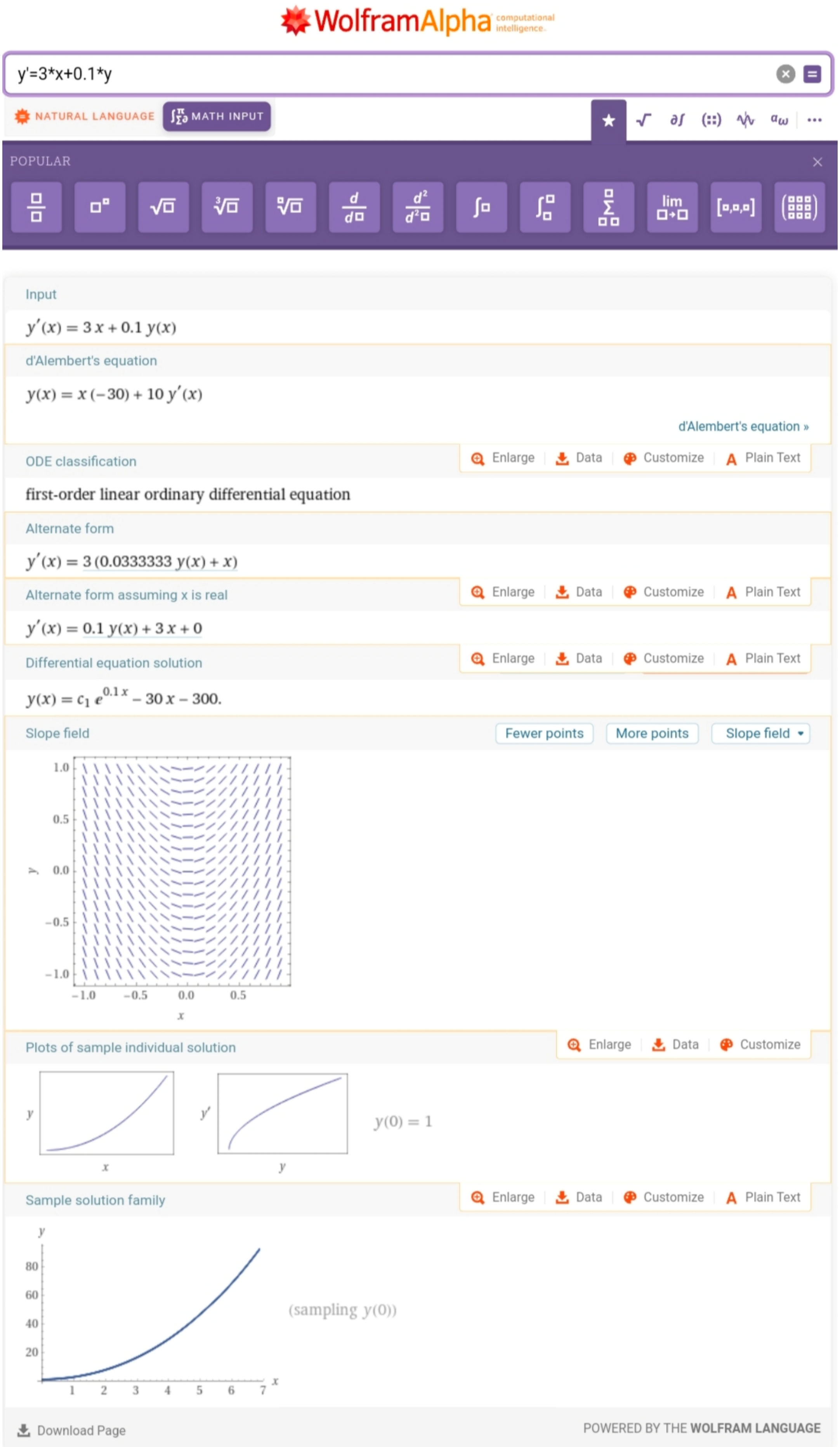
16. P. P. Kudryashov Algorithms for detecting a human face for solving applied problems of image analysis and processing: author. dis. Cand. tech. Sciences: 05.13.01. - M, 2007.

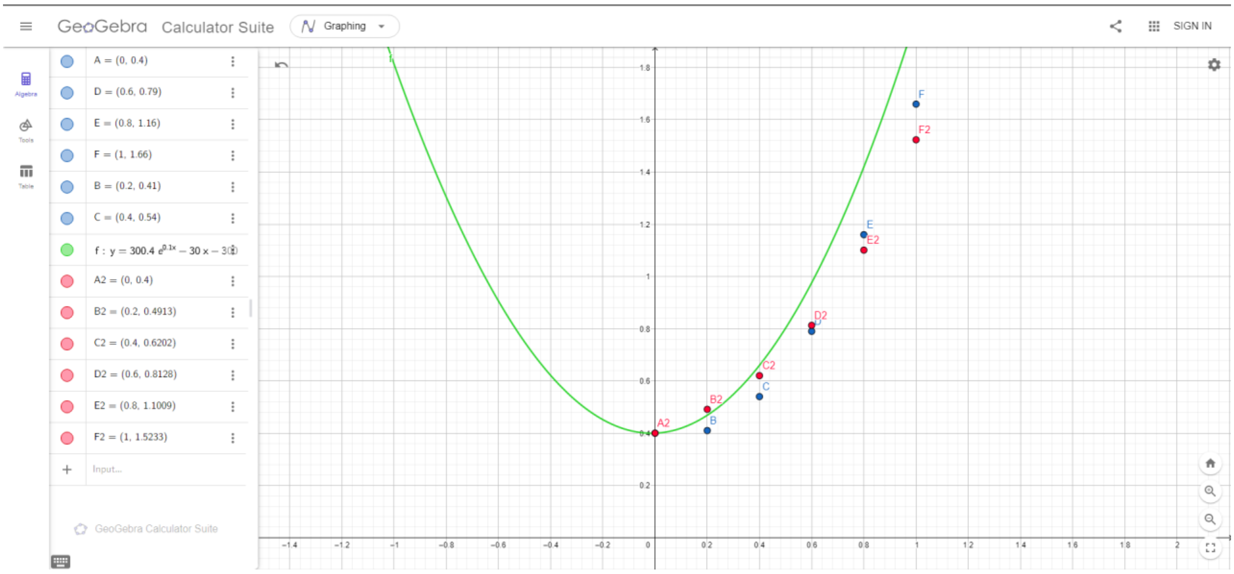
17. Tanenbaum E. Modern operating systems. 2nd ed. - SPb .: Peter, 2002 .-- 1040 p .: ill.

18. Forsyth DA, Pons, J. Computer vision. Modern approach / D.A. Forsyth, J. Pons: Trans. from English - M .: Publishing house "Williams", 2004. - 928 p .: ill. - Parallel. tit. English

19. Frolov V. Solution of systems of linear algebraic equations by the preconditioning method on graphic processor devices

20. Brodtkorb A.R., Dyken C., Hagen T.R., Hjelmervik J.M., Storaasli O.O. State-of-the-art in heterogeneous computing / A.R. Brodtkorb, C. Dyken, T.R. Hagen, J.M. Hjelmervik, O.O. Storaasli // Scientific Programming, T. 18, 2010. - S. 1-33.


WolframAlpha computational intelligence.
y'=3*x+0.1*y
NATURAL LANGUAGE
MATH INPUT
POPULAR
Input
$y'(x) = 3x + 0.1\, y(x)$
d'Alembert's equation
$y(x) = x(-30) + 10\, y'(x)$
d'Alembert's equation »
ODE classification
Enlarge
Data
Customize
Plain Text
first-order linear ordinary differential equation
Alternate form
$y'(x) = 3(0.0333333\, y(x) + x)$
Alternate form assuming x is real
$y'(x) = 0.1\, y(x) + 3x + 0$
Differential equation solution
$y(x) = c_1 e^{0.1x} - 30x - 300.$
Slope field
Fewer points
More points
Slope field
Plots of sample individual solution
$y(0) = 1$
Sample solution family
(sampling y(0))
Download Page
POWERED BY THE WOLFRAM LANGUAGE


GeoGebra Calculator Suite
Graphing
SIGN IN
Algebra
Tools
Table
A = (0, 0.4)
D = (0.6, 0.79)
E = (0.8, 1.16)
F = (1, 1.66)
B = (0.2, 0.41)
C = (0.4, 0.54)
f : y = 300.4 e^0.1x − 30 x − 30
A2 = (0, 0.4)
B2 = (0.2, 0.4913)
C2 = (0.4, 0.6202)
D2 = (0.6, 0.8128)
E2 = (0.8, 1.1009)
F2 = (1, 1.5233)
Input...
GeoGebra Calculator Suite
A2
B
B2
C
C2
D2
E
E2
F
F2