

Modelling of real kinematics situation as a method of the system approach to the algorithm development thinking

Stepan Hubalovsky

Abstract - One of the most important tasks in teaching of algorithm development and programming is to use a suitable method of developing theoretical knowledge of algorithm development and programming. Drawing on his own experience, the author has found that the best method to be used is the system approach. This method, described in the paper, is presented through the computer simulation of physical problems from the sphere of the kinematics of solid bodies. To be more specific, the solution of the problem of the trajectory modelling of the motion of a cylinder over the edge of two planes is presented. The solution is demonstrated progressively – it starts with the physical and mathematical analysis of the problem, then the algorithm development is presented and it finishes with the programming in MS Excel worksheet, through which the solution and visualization of the problem is reached. A case study of the system approach as a suitable method of the algorithm development thinking is presented in the paper step by step; the text also emphasizes interdisciplinary learning of university students.

Keywords—Algorithm development, animation, interdisciplinary learning, modeling, system approach.

I. INTRODUCTION

COURSES in the algorithm development and programming are currently provided in the Czech Republic at secondary schools, technical colleges and universities in the subjects focusing on informatics. The ability to create algorithms develops logical thinking skills and imagination and is an inseparable part of study skills of prospective and undergraduate teachers specializing in “Informatics” at the Faculty of Education.

Algorithms are encountered in all practical activities without being realized. An algorithm generally involves defining the rules and giving the sequences of steps necessary for any activities. The most common examples of algorithms in everyday life are various descriptions and giving directions.

In a more narrow sense, an algorithm can be defined [1] as “a precise rule defining the process leading from mutable input data up to the required results”. Such regulations are composed of various steps, stages, which are written in a particular order. A number of steps has to be finite. An algorithm can be understood as an activity which consists of problem analysis, designing phases, a method of algorithm

solution and finally of compilation of the relevant algorithmic language.

II. THE SYSTEM APPROACH TO THE ALGORITHM DEVELOPMENT AND PROGRAMMING

Teaching of the algorithm development as well as programming is usually carried out by teachers who also teach or are close to mathematics. The issue of the algorithm development is explained through mathematical tasks, which can be clearly described, defined and developed by algorithm. Altogether, the exercises are based on a routine rewriting of mathematical equations and formulas using algorithms and practicing the standard algorithm, regardless of their complexity and system integration to the whole entity [5], when the exercises used are “artificial”, inconsistent and disconnected from reality.

The system approach is still entirely missing in the teaching of the algorithm development and programming. The students who do not have sufficient mathematical experience and knowledge do not understand the tasks. They cannot see an interconnection between mathematical problems and problems which occur in real life. The knowledge of the algorithm development is not reached by these students; and that results in indifference or resistance to the algorithm and subsequent programming.

In recent years the structure of teaching of the algorithm development at the University of Hradec Kralove has significantly moved towards the use of the system approach to teaching algorithms and programming. This shift can be understood as important progress for students - prospective teachers - who have to be able to apply new methods in their work and disseminate these new methods among secondary school students. Benefits of the new method in teaching and learning computer science is also described e.g. in [2], [3], [4].

The following text will give a brief presentation of a method for developing of algorithmic thinking of our students. This development should be enabled thanks to the method of the system approach applied in the subject *Algorithms*, which contributes to the motivation of students.

Then a case study will be presented. It illustrates a step by step modelling of a real physical situation as a suitable example of an algorithmic thinking development and also as an example of the interdisciplinary system learning that is useful

to be included in secondary school teachers' education.

A. Principles of the system approach

Education at secondary schools and colleges in the area of informatics is directed mainly to a user's attitude. Only students who have attended optional subjects dealing with programming languages are familiar with creating algorithms. Thus a lot of students studying at our university lack any algorithmic knowledge at the beginning of their studies.

The system approach to the problem solution is such an approach which understands the studied phenomena and processes in complex internal and external contexts [6]. The system approach in education consists of the formulation, understanding and solving the problem, relevant processes, phenomena and processes that exist objectively in the outside world, and their transformation into a model educational situation.

The methodology, proposed in this paper, uses for training of algorithms the exercises which are based on modelling of real and idealized phenomena, processes and procedures. The methodology is intended to lead the student to a problem definition, to a creation of suitable models and to implementation and realization of the model through the principles of algorithms, i.e. the student should be able to determine about the input and output data and the procedure of the transformation of the data input into the data output under the system approach.

B. Application of the system approach

The proposed new methodological approach to teaching of algorithms can be applied:

- 1) In the field of sciences - in case of models of physical and chemical processes and phenomena;
- 2) In the technical field - in case of the process control of machines and simple robots;
- 3) In the humanities and social studies - in case of the processes and phenomena associated with this issue;
- 4) In management - in case of the management processes and quality control processes.

Mathematical calculus is essential in a "model" method, but only as a secondary issue. The standard method of a simple rewriting of mathematical expressions and equations into algorithms and then into the programs is maximally eliminated while using this approach.

It should be emphasized that the use of the system learning of algorithms based on a comprehensive methodological approach can be connected with the fact that mathematics, sciences and technology cannot do without the use of the ICT technology; new technologies have become an integral part. A current teacher has to master not only the teaching process of his/her discipline, but also the foundations of the computer science. Likewise, a teacher of informatics (a computer programmer) cannot be only a computer teacher (a programmer), but s/he has to be able to systematically apply her/his knowledge of ICT in other fields.

The current non-systematic approach to teaching leads to such situations in which subjects are taught in isolation, without inter-disciplinary linkages. E.g. mathematical definitions, theorems and formulas are taught independently; the basic principles of algorithms, including the development of a flowchart are taught independently; the basic principles of programming with emphasis on the programming language syntax are interpreted independently. The education lacks a comprehensive system approach which would include links to e.g. physics and physical description of the real world. As a result, students receive a set of isolated findings and rules they are unable to logically link together.

III. A MODEL PROBLEM SOLVED THROUGH A SYSTEM APPROACH

As it was already mentioned at the beginning of the paper, it is essential to emphasize interdisciplinary learning while training future teachers.

This section will describe one real situation using interdisciplinary approaches to its solution. Similar examples can be found e.g. in [7], [8], [9].

A. Problem definition

The situation shown in Figure 1 is to be considered – it concerns an inclined plane intersecting with a horizontal plane of angle α .

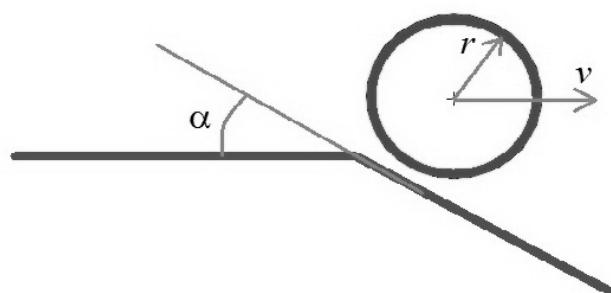


Fig.1 Illustration of the given problem

A homogeneous cylinder of radius r rolls the plane from the left to the right with velocity v . With respect to r , v and α it has to be decided whether the cylinder rolls over the edge or rebounds from the edge. If the cylinder, under specified conditions, rebounds, it is necessary to find a place on the inclined plane where the cylinder snaps. The situation has to be analyzed on the basis of available physical laws.

Students should define the appropriate algorithm of the problem solution and the whole situation should be animated with appropriate tools.

B. Analysis of the problem

First, it is necessary for the student to realize the problem and perform the physical analysis.

One of the possible solutions is as follows.

Let us introduce simplifying assumptions: the rolling cylinder can be replaced by a moving point mass, whose

position coincides with the center of the gravity of the cylinder. The center of gravity lies on the cylinder axis of the rotation of the roller. The task can then be transferred to determination of the trajectory of the point mass moving by speed v at the distance r from the horizontal plane. The trajectory of the gravity center will be expressed in the form of $y = y(x, v, r, \alpha)$ dependency.

On the left side (above the horizontal plane) the trajectory of the point mass is a line parallel to the plane. The distance between the plane and the line is r .

The situation changes when the point mass (the axis of the cylinder) is just above the edge (where the horizontal plane changes in the inclined plane). At this point a student of physics should realize how the task has to be solved. The kinematic or dynamic solutions are available. Both the possibilities lead to the correct solution, but the kinematic solution (i.e. a non-force solution) is simpler and easier for the algorithm development and programming.

IV. KINEMATIC ANALYSIS

The kinematic analysis of the problem will be shown in the following part of the paper.

A. Kinematic analysis of limit cases

Let us suppose, first, that the angle of $\alpha = 0$ rad. In this case, the inclined plane is horizontal (parallel to the left horizontal plane) and the break edge de facto does not exist. The point mass still moves at the distance r from the plane (its trajectory is the line parallel with the plane). The cylinder still rolls.

Another limiting situation is, if angle $\alpha = \pi/2$ rad, i.e. both the planes are perpendicular. The situation is shown in the Figure 2.

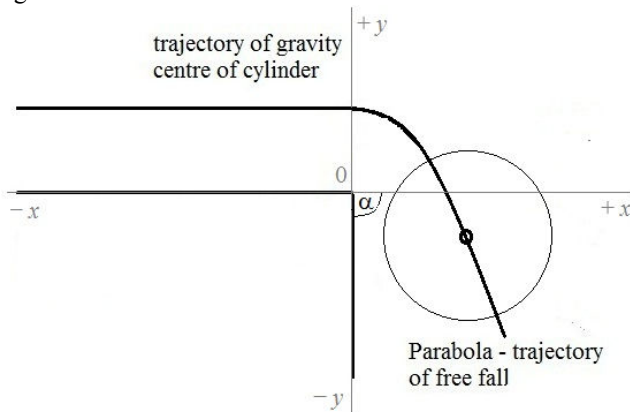


Fig.2 Trajectory of the cylinder for $\alpha = \pi/2$ rad

In this case, the trajectory of the point mass is identical with the trajectory of the horizontal throw and can be described in the $[x, y]$ coordinate system by the equations for the parabola

$$x = vt ; y = r - \frac{1}{2}gt^2, \tag{1}$$

where t is time, v is velocity of the cylinder, r is radius of the

cylinder and $g = 10 \text{ m/s}^2$ is gravity acceleration.

By eliminating time from (1) the following formula for y coordinate is received:

$$y = r - \frac{1}{2}g\left(\frac{x}{v}\right)^2 = r - 5\left(\frac{x}{v}\right)^2, \tag{2}$$

where x is x -coordinate with the initial point at the planes edge.

The cylinder rebounds at the edge and never falls back.

B. Kinematic analysis of general cases

The analysis of the situation for the angles in the interval $\alpha \in (0, \pi/2)$ rad can be presented by the following considerations. Under certain conditions, given by the input values and v, α and r , the cylinder will roll over the edge, the cylinder still "holds" the plane (it does not rebound from the edge). In this case, the trajectory of the point mass of the cylinder at the time of rolling over the edge is a part of the circle. The situation is shown on the Figure 3.

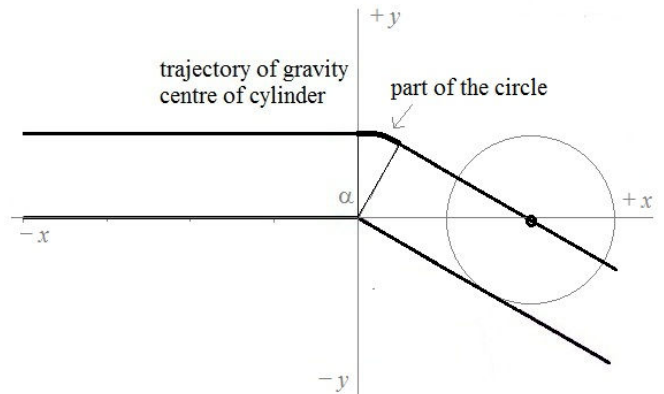


Fig.3 Trajectory of the cylinder – rolling over the edge

The trajectory of the circle is given by the equation:

$$y = \sqrt{r^2 - x^2}. \tag{3}$$

Then the point mass moves along a line parallel to the inclined plane at the distance r . This line can be described by slope–intercept straight line form:

$$y = \text{tg}(-\alpha)x + \frac{r}{\cos \alpha}, \tag{4}$$

where $\text{tg}(-\alpha)$ is the slope of the line and $r/\cos \alpha$ is intercept.

The overall trajectory is composed of three parts:

- the horizontal line;
- the part of the circle;
- the inclined line.

The trajectory can be described by the following formulas:

$$\begin{aligned}
 x \in (-\infty; 0): & \quad y_C(x) = r \\
 x \in (0; r \sin \alpha): & \quad y_C(x) = \sqrt{r^2 - x^2} \\
 x \in (r \sin \alpha; \infty): & \quad y_C(x) = \operatorname{tg}(-\alpha)x + \frac{r}{\cos \alpha}
 \end{aligned} \tag{5}$$

where $r \sin \alpha$ indicates x -coordinate of the point, where circle trajectory change to inclined trajectory.

If the cylinder (the point mass of the cylinder) will have a sufficient velocity, the cylinder will rebound from the edge. The trajectory of the point mass from the edge will be given by the (2), (the parabola (2) is geometrically higher than the circle (3) is). The trajectory of the point mass of the cylinder will be a parabola up to the point the cylinder impacts the inclined line, i.e. the distance of the cylinder point mass equals r . The overall trajectory is composed of three parts:

- the horizontal line;
- the part of the parabola (see (1));
- the inclined line.

The trajectory can be described by the following formulas:

$$\begin{aligned}
 x \in (-\infty; 0): & \quad y_p(x) = r \\
 x \in (0; x_{\text{impact}}): & \quad y_p(x) = r - 5\left(\frac{x}{v}\right)^2 \\
 x \in (x_{\text{impact}}; \infty): & \quad y_p(x) = \operatorname{tg}(-\alpha)x + \frac{r}{\cos \alpha}
 \end{aligned} \tag{6}$$

where x_{impact} indicates x -coordinate of the impact point and is given by the solution of the equation for x :

$$r - 5\left(\frac{x}{v}\right)^2 = \operatorname{tg}(-\alpha)x + \frac{r}{\cos \alpha} \tag{7}$$

The situation is shown in Figure 4.

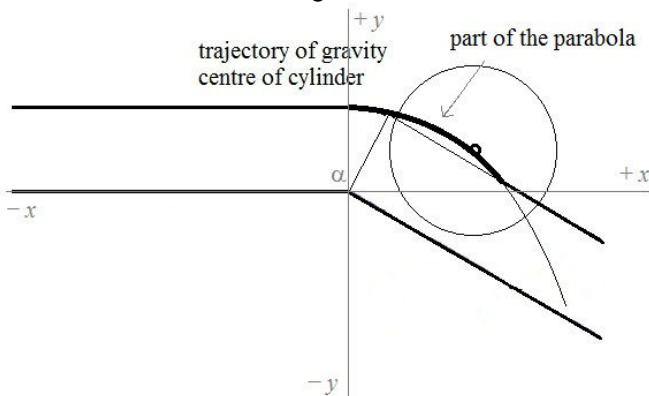


Fig.4 Trajectory of the cylinder – rebound from the edge

The cylinder rolls over the edge if the trajectory of the circle is higher than the trajectory of the parabola (the values of the equation (5) are greater than the values of the equation (6)).

The cylinder rebounds from the edge if the trajectory of the parabola is higher than the trajectory of the circle (the values of the equation (6) are greater than the values of the equation (5)).

V. ALGORITHM DEVELOPMENT

The algorithm can be developed on the basis of a physical analysis. The algorithm allows us to build up a computer program suitable for the visualization of the situation.

The algorithm is based on the numerical finding of the maximum value of the equations (5) and (6) (y coordinate) for the set of x coordinate. If the maximum is given by (5), the cylinder rolls over the edge, if the maximum is given by (6), the cylinder rebounds from the edge.

A. Input and output data

To build an algorithm, it is necessary to define the input data - this means the speed of the cylinder v , the radius of the cylinder r and the angle between the horizontal and inclined planes α .

The simplest form of the output is, whether the rebound occurs or not (variable of the Boolean type); and if it does, then what the x -coordinate of the impact on the inclined plane is.

B. Description of the algorithm

The final algorithm of the solution given in the form of the flowchart is presented in Figure 5.

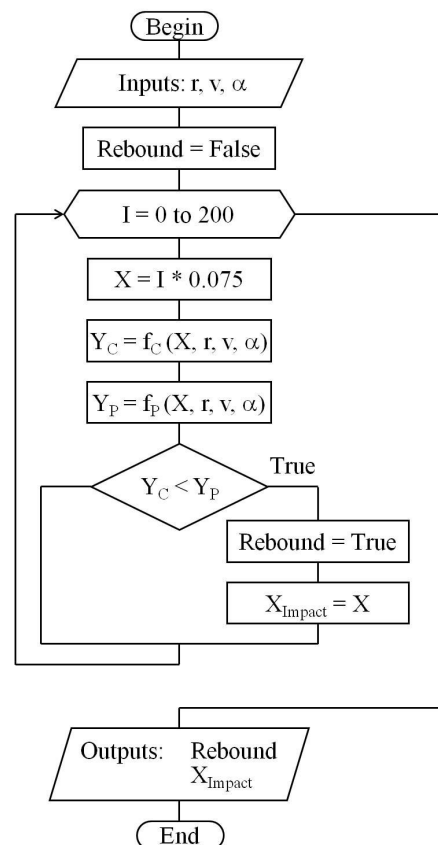


Fig.5 The algorithm of the problem solution

The algorithm is based on gradually step by step comparison of value of y_C given by (4) and y_P given by (5) calculated for a sufficiently small increases in the coordinates x . That means using a cycle with a finite number of repetitions and conditions.

In this particular case the interval of $x \in \langle 0,15 \rangle$ is split to 200 parts with increase $dx = 0.075$.

This algorithm can be rewritten in any programming language, e.g. in the algorithmic meta-language, as in [3].

```

begin
  read (r, v, α)
  Rebound := False;
  I := 0;
  while I <= 200 do
  begin
    X := I * 0.075;
    Yc := fC(X, r, v, a)
    Yp := fP(X, r, v, a)
    if Yc < Yp then
    begin
      Rebound := True;
      Ximpact := X;
    end;
    I := I + 1;
  end;
  write (Rebound, Ximpact);
end.
    
```

However, students can also consequently reach the results using MS Excel and then they can practice the work with this environment.

VI. SOLVING THE PROBLEM USING MS EXCEL

The task can be also easily demonstrated using formulas entered in the MS Excel spreadsheet cells.

This solution, given by a wide negotiability of MS Excel, appears to be the most advantageous. Another reason to use MS Excel is that students should be accustomed to using a programming tool as simple as possible.

In the following part of the paper there will be shown how various elements of the algorithm (**inputs, outputs, cycle, conditions**) can be rewritten by commands directly in an Excel worksheet, and how to create a simple animation chart.

A. Inputs and Outputs

The input and output data in the Excel worksheet are conducted directly into the cells. The cells should be colour-marked to indicate that those are the “significant” cells. In our particular case, the algorithm input values r , v and α are entered into cells H2, H3 and H4, the output data into cells H6, H7.

B. Cycle

Various steps of the cycle defined by the number of repetitions can be implemented for each step of the cycle to be

represented by different cells of one column. The formulas of the cycle are entered in cells A2, B2 and B3 and then, dragging the handle of the cells, copied into any number of cells (using relative addressing of the cells). The number of the copied cells corresponds to the total number of the repetitions of the cycle.

A possible solution is shown in Figure 6. The interval of the solution is for the values of x taken from 0 to +15, this interval is divided into 200 parts, which corresponds to the number of steps of the cycle of the algorithm flowchart. The increment dx is 0.075.

	A	B	C	D	E	F	G	H
1	x	y_C	y_P					
2	0,00	5,00	5,00	0		Inputs:	r =	5
3	0,08	5,00	5,00	1			v =	10
4	0,15	5,00	5,00	1			α =	30
5	0,23	4,99	5,00	1				
6	0,30	4,99	5,00	1		Outputs:	Rebound:	YES
7	0,38	4,99	4,99	1			X Impact	9,97
8	0,45	4,98	4,99	1				
9	0,53	4,97	4,99	1				
10	0,60	4,96	4,98	1				

Fig.6 Solution of the problem in MS Excel Spreadsheet

The set of values X is in column A. The values of Y_C given by (5) are in the column B. The values of Y_P given by (6) are in the column C.

C. Conditions

Column D contains a condition that compares the corresponding values in columns B and C. The formula in e.g. row 3 has following expression:

$$=IF(C3>B3;1;0).$$

In the case that at least one of the values of column B is bigger than the corresponding values in the cells of column C (which is identified by 1) the cylinder will rebound from the edge. This is one of the outputs given in cell H6. The formula in this cell has following expression:

$$=IF(SUM(D2:D202)>0;"YES";"NO"),$$

The output is “YES” if the summation of values in the column D is greater than 1, otherwise the output is “NO”.

The last value in the cells of column C which is bigger than that in the corresponding cells of column B (last value 1 in column D) approximately determined the x – coordinate of the impact point. To determine this point the following function is inserted in the cells of the column E:

$$=IF(AND(D3=1;D4=0);A3;"").$$

This condition compares if the corresponding cell of the column D equals 1 and the following one equals 0. If the condition is fulfilled the corresponding value from cell of column A, which indicates x -coordinate of impact point, is copied. This value is also copied to cells H7 as the second output.

VII. VISUALIZATION OF THE PROBLEM

This section will demonstrate how the given situation can be relatively simply and effectively visualized while using MS Excel. Based on the animation of the cylinder motion along the calculated trajectory, the visualization is created.

A. Animation

Animation generally means a way how a static image can be seemingly set in motion. The principle of an animation is recording of a sequence of images which slightly differ. During a quick viewing of subsequent images an impression of movement occurs, due to the phenomenon of the persistence of vision. The frequency of changing of individual images is proportional to the displacement of the object in consecutive images. (The bigger the displacement of objects in each image is, the greater the frequency of changing these images must be to maintain a sense of motion).

Animations of the cylinder motion can be realized by using an XY scatter chart. The chart itself does not change, however, only the position of the lines on the chart changes. The chart shows both the *static baselines* and the *dynamic animated objects*.

The static baselines represent the horizontal inclined plane and the trajectory of the cylinder center (the point mass representing the cylinder). The line representing the trajectory on the left side from the edge is horizontal in distance r from the horizontal plane. On the right side from the edge there are lines which represent both trajectories given by (5) and (6). The above lines do not change during the animation, they are static.

Dynamic objects which move on the chart with respect to static baselines are illustrated by a point, which represents the cylinder center, and the circle, which represents the edge of the circle – see Figure 7.

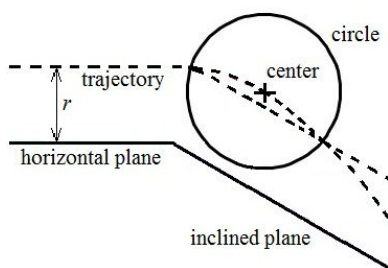


Fig.7 Static baselines and dynamic objects

The principle of the animation of the scatter chart in MS Excel is based on the change of one or more parameters of the charts input data of the dynamic objects (the cylinder center and the circle). The objects step by step change its position in relation to the static baselines. In this particular case one parameter – x coordinate of the cylinder center - will be step by step changed by a “rather small” increment dx . A change of this parameter can be achieved by using so-called *iterative calculation* of values in the cell. Other variables needed for the animation of the dynamic objects depends on this x coordinate

and they are recalculated directly in the Excel worksheet.

B. Iteration

In general, iteration means repetition of certain activities - the word comes from the Latin *iteretur* - repeat. The iteration in mathematics and numerical methods is a process of applying a function f repeatedly, using the output value x_{n+1} of the function as input argument x_n of the function in the next step. This can be written by mathematical formula

$$x_{n+1} = f(x_n) \tag{8}$$

A typical example of the use of iterations is the numerical solution of nonlinear equations by the general iterative method - see e.g. [10]. A typical example of a nonlinear equation is the transcendental equation

$$x = \cos x \tag{9}$$

To find the approximate solution, one has to properly set the initial value of x (e.g. $x_0 = 1$), to provide the iteration ($x_{i+1} = \cos x_i$) and to see how the value of x is changing. The termination of the iterations is determined by the required accuracy - ε . The calculation will be terminated if the following condition is fulfilled:

$$|x_{i+1} - x_i| < \varepsilon \tag{10}$$

Apart from solving of the nonlinear equations, the iteration calculus can be used e.g. for determination of values of *recursively given sequences*.

As mentioned e.g. in [4] or [8] the sequences are defined either by explicit formulas for the n -th terms or recursively which means that the previous values of terms of the sequence are used to generate the value of the next term of the sequence. In other words one needs to know the initial value x_0 and the recurrence construction to be able to find subsequent terms of the sequence.

Let us briefly remind a basic mathematical formulation of recursively defined sequences. Given a real function $f: N_0 \times R \rightarrow R$ and an initial value $x_0 \in R$, consider the sequence of iterates of x_0 under the given function f :

$$\begin{aligned} &x_0, \\ &x_1 = f(0, x_0), \\ &x_2 = f(0, x_1), \\ &x_3 = f(0, x_2), \\ &\dots \end{aligned} \tag{11}$$

Such a sequence is called a recursively defined sequence. It means that for a given x_0 , we can compute an arbitrary term of sequence x_t , where $t = 0, 1, 2, \dots$, according to the relation

$$x_{t+1} = f(0, x_t) \tag{12}$$

This relation, together with an initial value x_0 is called a first

order recurrence equation. The solution to this equation is any sequence x_t generated by this equation.

The *termination condition* in the case of recurrently given sequences is given by number of sequence terms – t_{max} .

Example of such sequence can be sequence given by following formula:

$$\begin{aligned} x_0 &= 0, \\ x_{t+1} &= x_t + 2 \\ t_{max} &= 5 \end{aligned} \tag{13}$$

with the terms 0, 2, 4, 6, 8.

C. Iteration in algorithm development and programming

Iteration in algorithm development and programming means a repeated call of the function whereas the argument of the function is the result of the previous function call. In this sense, the iteration corresponds to the aforementioned case of recursively given sequences. The iteration in algorithm development and programming is therefore called recursion.

An important part of recursion is a terminating condition which determines when the recursion is to be stopped. The recursion in programming is quite often used, as well as e.g. the assignment statement $A:=A+2$ can be understood as a form of recursion based on (13) with the number of repetition $t_{max} = 1$. This expression means that the new variable value equals to the original value increased by 2.

D. Iteration /recursion in the MS Excel

The circular reference can occur in the formula of the cell in the Excel sheet. This means that the formula in a cell refers to the value of the same cell, eventually that the formula of the cell refers to the value of another cell, which has a formula linking back to the original cell.

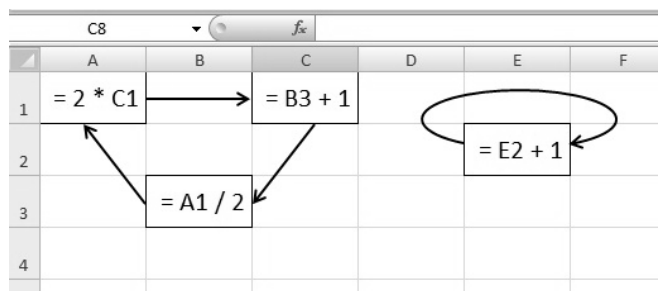


Fig. 8 Examples of circular reference in MS Excel

The value of such cells can be determined only by the iterative calculation. If such an iterative calculation is not possible in MS Excel option tools (default is disabled), the calculation will be executed and an error message will be shown.

The iterative calculation in version of MS Excel 2007 can be enabled in the "Excel Options" dialog box (which is reached by clicking to the Excel option button in the main menu) in the folder "Formulas" – see Figure 9. The "Enable iterative calculation" checkbox has to be checked.

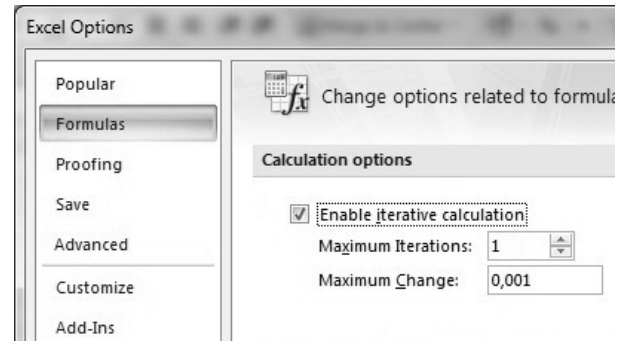


Fig. 9 Setting of iterative calculation parameters in MS Excel

The item "Maximum Iterations" specifies the maximum number of iterations, repetitions, which Excel provides in an iterative calculation. The item "Maximum Change" sets the accuracy of the iterative calculation - the value of the cell before and after the iteration is compared. If the absolute difference of the both values is smaller than the accuracy of the calculation, the iteration is stopped. The default value of "Maximum Iterations" is set to 100 and the default value of "Maximum Change" is set to 0.001.

The iterative calculation and its termination can be expressed in the form of the algorithm flowchart in Figure 10, where the following variables are used:

- NI - the number of the already done iterations;
- X_{old} - the value of the cell before the iteration;
- X_{new} - the value of the cell after the iteration;
- MI - the maximum iterations;
- MC - the maximum change.

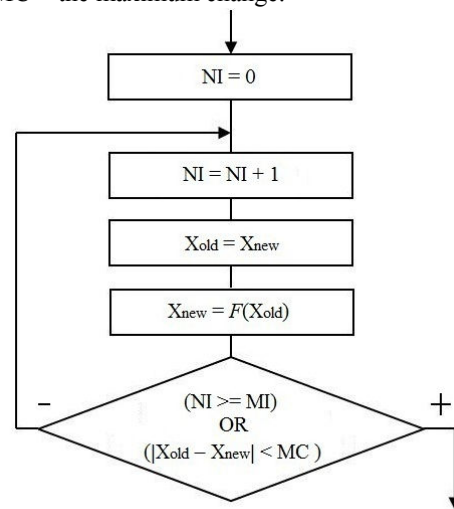


Fig. 10 Flowchart of iterative calculation in MS Excel

It is clear that the terminating condition is given either by the maximum of iterations (13) or by condition (10).

E. Setting of iterative calculation for an animated chart

In the following passage it will be shown how to use the iterative calculation in animated charts. Specifically, as it has been mentioned above, the iterative calculation will be used to

change the x -coordinate of the center of the cylinder. The value of this parameter is entered in cell X3.

The circular reference formula

$$=X3+W3,$$

is entered into cell X3. This formula expresses that the value of cell X3 after one iterative recalculation step increases by the increment value entered in cell W3 - iteration (animation) step. Iterative recalculation starts by pressing of F9 key. The number of the iteration steps is set by the value of "Maximum Iterations". To make only one step of the iteration, defined by formula in cell X3, the value of "Maximum Iterations" has to be equal to 1, as is evident from the Figure 9.

By repeated pressings, eventually by holding down the key F9, the value of cell X3 will step by step increase by the value of the animation step – value in cell W3.

If only the formula =X3+W3 was entered to the cell X3, the value of X3 would after each iteration step still increase. It is therefore necessary to determine the terminating condition, i.e. minimum and maximum limits, given by the values in the cells U3, V3 respectively, so that only within this interval the value of cell X3 will increase.

This can be done by adding the following condition to formula in cell X3:

$$=IF(X3>=V3,U3,X3+W3).$$

This condition compares the value in cell X3 with the value in cell E3. If the condition is true ($X3 \geq V3$) the value of X3 changes to the minimum given by value in cell U3. Otherwise, the iterative calculation is carried out and the value of X3 cells increases by the value of the iteration step (value of the cell W3). The situation is shown on the Figure 11.

Font		Alignment		Number			
fx =IF(X3>=V3;U3;X3+W3)							
S	T	U	V	W	X	Y	
		Interval of x_c change		Iterative (animated)	Coordinates of the cylinder center		
		Min	Max	Step	x_c	y_c	
		-20	20	0,5	18,50	-4,91	
						Coordinates of points of the circle	
				α	x	y	
				0	23,5000	-4,9075	
				5	23,4810	-4,4717	
				10	23,4240	-4,0392	
				15	23,3296	-3,6134	
				20	23,1985	-3,1974	

Fig. 11 Solution of the iterative calculation of the cylinder center and circle coordinates

The iterative recalculation of the above defined parameter – x coordinate of the cylinder center will be within the range from a minimum value (value of the cell U3) to a maximum value (value of the cell V3) with the iteration step (the value of the cell W3) by repeated pressings or by holding of key F9. Within this interval, the center of cylinder will “move” from left to right in the XY scatter chart.

Simultaneously, the y -coordinate of the cylinder center is

recalculated using the formulas (5) and (6). The formulas are entered to cell Y3.

F. Animation of movement of cylinder in XY scatter chart

The above mentioned principle of the iterative recalculation of the x and y coordinates of the cylinder center can be advantageously used as a base for creation of an “animated” chart.

MS Excel spreadsheet has implemented tools for creation of graphs. These tools allow users to create different types of charts, and among others, e.g. a XY scatter chart, which allows to present a set of $[x, y]$ points with given coordinates x and y in the plane. A different type of the curve can also be plotted in this chart, if the “sufficient” number of the $[x, y]$ points forming the curve is used.

In this case, on the chart they are shown both the static baselines (the horizontal plane, the inclined plane and the trajectory of the cylinder center given by equations (4) and (5)) and the moving objects (the point representing the cylinder center and a circle around this center, showing the surface of the cylinder). The coordinates of the cylinder center ($[x_c, y_c]$) are given by values in the cells X3 and Y3, respectively.

The values of the set of the points $[x, y]$ needed for drawing of the circle depend on the coordinates of the cylinder center ($[x_c, y_c]$) by following formulas:

$$\begin{aligned} x &= x_c + r \cos \alpha, \\ y &= y_c + r \sin \alpha \end{aligned} \tag{14}$$

where α is the angle, the change is from 0° to 360° with increment 5° . This set of points is also recalculated during each step of the iteration of cell X3. The solution in MS Excel is shown in Figure 11.

The result of the above mentioned is this: when the F9 key is repeatedly pressed or held, the center of the cylinder and a circle (drum surface) “move” from the left to the right on the chart and the chart appears as animated.

Figure 12 shows the XY scatter chart, created directly using the Excel tools, of the trajectories of the mass point for values $r = 5, v = 10$ and $\alpha = 30^\circ$.

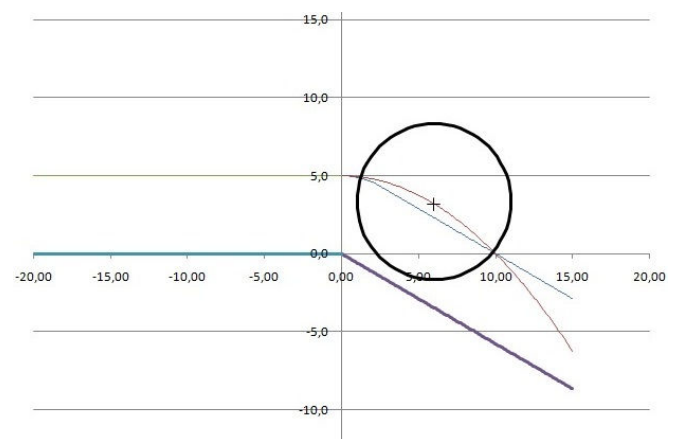


Fig. 12 XY scatter chart of the solution in MS Excel graph tool

VIII. CONCLUSION

There are various approaches how to provide training in algorithms, how to introduce and develop basic algorithmic thinking of students.

A new kind of the teaching / learning strategy using the system approach was presented in the paper.

Algorithms as the general basis of a structured problem solving can be understood as the basis of the computer science. The formation of the structure of the problem in separate steps develops creative thinking while applying a system approach to different types of problems. The paper emphasizes the fact that the algorithm is neither a mere mathematics, nor a mere programming.

The system approach can be set as the default paradigm for a wide integration of the principles of the algorithm development into education.

ACKNOWLEDGMENT

This research has been partially supported by the Specific research project of the Faculty of Science of University of Hradec Kralove No. 2129.

REFERENCES

- [1] A. A. Markov, *Theory of algorithms*, Moscow: Imprint, 1954.
- [2] E. Milkova, "Multimedia Applications and their Benefit for Teaching and Learning at Universities", *WSEAS Trans. on Information Science & Applications*, Issue 6, vol. 5, June 2008, pp. 869-879.
- [3] E. Milkova, "Algorithms: The Base of Programming Skills", in *Proc. 29th International Conferences on Information Technology Inter-Faces*, University of Zagreb, Croatia, 2007, pp. 765 – 770.
- [4] P. Prazak, "Recursively Defined Sequences and CAS", in *Proc. 6th WSEAS/IASME International Conference on Educational Technologies (EDUTE'10)*, WSEAS Press, Kantoui, Sousse, Tunisia, 2010, pp. 58-61.
- [5] B. Wilson, *Soft Systems Methodology: Conceptual Model Building and its Contribution*. Hoboken: Wiley, 2001.
- [6] W. Dettmer, *The Logical Thinking Process: A Systems Approach to Complex Problem Solving*, Visconsin: Quality Press, 2007.
- [7] S. Hubalovsky, E. Milkova, "Modelling of a real situation as a method of the algorithmic thinking development", in *Proc. 6th WSEAS/IASME International Conference on Educational Technologies (EDUTE'10)*, WSEAS Press, Kantoui, Sousse, Tunisia, 2010, pp. 68–72.
- [8] S. Hubalovsky, E. Milkova, P. Prazak, "Modeling of a Real Situation as a Method of the Algorithmic Thinking Development and Recursively Given Sequences", *WSEAS Trans. on Information Science & Applications*, Issue 8, vol. 7, Aug. 2010, pp.1090-1100.
- [9] S. Hubalovsky. "Use of application MS Excel for calculation of the gravity centre of a system of right parallelepipeds". *Mathematics, physics, informatics*, vol. 19, 2010, pp. 551-558. (in Czech)
- [10] V. Jehlicka, *Numerical methods, Solution of non-linear equations* [Online]. 2006. Available: <http://lide.uhk.cz/pdf/ucitel/jehliv1/>, (In Czech).

Stepan Hubalovsky was born in Trutnov, Czech Republic in 1970, he obtained master degree in education of mathematics, physics and computer science in 1995 and doctor degree in theory of education in physics in 1998 both in Faculty of Mathematics and Physics, Charles university in Prague, Czech Republic.

He worked 5 years as master of mathematics, physics and computer science on several secondary schools. He works as assistant professor on University of Hradec Kralove from 2006. He interested in algorithm development, programming and computer modelling.

RNDr. Stepan Hubalovsky, Ph.D. is member of Union of Czech Mathematicians and Physicist.