Gayev Ye.A., Azarskov V.M.

Educational “Own Discoveries” Method by an easy MATLAB-Programming for Engineers

Aerospace Control Systems Department of National Aviation University (Kyiv, Ukraine),
Ye_Gayev@i.ua, Azarskov@nau.edu.ua

INTERNATIONAL CAE CONFERENCE
2018, 8th - 9th October | Vicenza, Italy

We present an innovative educational course of “Programming for Engineers” here that realizes own student’ discoveries in science. We get first year students with “shiny eyes”. Our education aims to keep their shining till the 6th year.

How to teach future engineers that are students yet?

It is a crucial question for our civilization today!
How to account for the facts

➔ today’s youth has not been a fan of mathematics and physics like it was in 1960-1970,

but

➔ they are fans of computers and the Internet, and often know them better as teacher?

We find the above are key stones to find proper education methodology.

- Why do we emphasize “Engineers” in the course title? Because they need quite different ‘Programming’ with regard to ‘programmers’, with more Mathematics and Physics, with more Algorithms.
- Dealing with modern newcomers, we need sometimes to re-switch students from music, TV and Internet to serious technologies;
- We apply methodology of “Own student Discoveries” [1,2];
- The proper way to realize “Own Discoveries” is an Easy Programming;
- There is a number of programming languages. Today’s student learn Delphy, C, C++, Java, Python etc. Indeed, they are valuable for students to become a programmer but not an Engineer. To educate Engineers we suggest a wide use of modern mathematical packages, particularly MATLAB, that are the only to lead to “Own student Discoveries” [1,2].

In this poster we demonstrate how we realize our educational approach in Aerospace Control Systems Department of National Aviation University (Kyiv, Ukraine).

Easy Programming is the way to everything!

- 1 -

Method of “Own student’ Discoveries”

Education is rather conservative field about 2 000 years old. Lectures, one of its main instruments, is a passive one. Laboratory and practical works have been somewhat active; however, students often execute them ‘automatically’ without bearing in mind the question “Why is so?”. Cinema and TV allowed to see many uncommon things by own eyes today, but it is a passive method as well. Computer technologies significantly extended what students could try and test in their Laboratory Works. Most of students, however, follow to ready hints without asking “Why is it so?”.

“Why is it so?” means that the student should repeat an original discovery in the field and understand its origin. She/he will not require 100 years for this, or several months, or even a week – a PC-environment should help to get result in hours or even in seconds. This is what we mean under “**Education by own discoveries**”.

The British genius Stephen Wolfram suggested revolutionary to learn each science through programming cellular automata [3]. Our approach is more conservative and practical: to devote one and half first years of teaching to an Easy Programming, to creation a few own educational programs, and then systematically apply it in almost all Curriculum disciplines.

We invest two semesters of the First teaching year to get students familiar with MATLAB as an Easy Programming environment. The third semester is devoted to object-oriented language Java. We try to start ‘own discoveries’ as soon as possible. Our time plan lies in the following [4,5]:

1. ABC’ of the MATLAB, use its visualization capabilities to ‘discover empirically’ how the polynomial sequence $f(0)+\frac{f'(0)}{1!}x+\frac{f''(0)}{2!}x^2+\frac{f'''(0)}{3!}x^3+...$ tends to the function $f(x)$ and another one $4\left(\frac{\sin(x)}{1}+\frac{\sin(3x)}{3}+\frac{\sin(5x)}{5}+\frac{\sin(7x)}{7}+..\right)$ to $f(x)=\begin{cases}-1/\pi, & x\in(-\pi,0)\\ +1/\pi, & x\in(0,\pi)\end{cases}$.

- 2 -

2. Philosophical difference between analytical and numerical actions like $(\cos x)' \to -\sin x$ $\frac{f(x+\Delta x)-f(x)}{\Delta x}$, and $\int_a^b f(x)dx$ and $\sum_{i=1} \frac{f(x_{i+1})+f(x_i)}{2}(x_{i+1}-x_i)$ (i.e. analytical and numerical getting derivative and integral).

3. Idea of Algorithms; start programming in MATLAB.

4. First (simple) own ‘discoveries’ lead by Instructor: finding ‘experimentally’; shutting stone over surface; probability.

5. Animation, plotting and rotation of a *N*-gon, plotting and rotation of *N*-pointed star, a Propeller.

6. Standard algorithms: *MySinus*, search of *Min* and *Max*, re-ordering numeric and text arrays.

7. Simple use of sound and image.

8. Complex data types, polymorphism in MATLAB-programs.

9. Complex algorithms: iterations, simple fractals, recursion, wavelets.

10. Graphical User Interface.

11. Complexity of algorithms, its ‘experimental’ determination by Least Square Method;

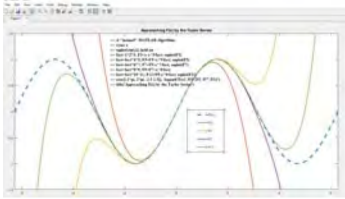
12. Term Paper on student choice to realize her/his own project. Some of programs made by students are displayed below.

The third semester is usually devoted to Java and to OOP-properties of MATLAB. It includes similar problems, approaches and algorithms. It ends with Term Paper as well.
Programming abilities gathered students apply to other disciplines studied in consequent study years, see below.

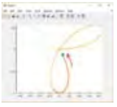
- 3 -

We start with inspiring students with powerful MATLAB ‘art’

instead of boring ‘to solve quadratic equation $ax^2+bx+c=0$’ or “Hello, World”.

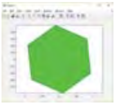


The second aim is to bring together this Programming course and student’ Mathematics and Physics.



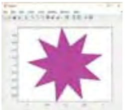
```
>> %Body moving a complex trajectory
>> N=3; t=0:pi/20000:2*pi;
>> x=sin(N*t).*cos(t); y=sin(N*t).*sin(t);
>> comet(x,y)
```

First Algorithm: create *N*-gon, create *N*-pointed star:



Students repeat such programs
(A) with GUIs controlling
(1) figure color,
(2) rotation direction,
(3) rotation speed etc.,

(B) In Java later on.



```
>> %Manual creation of an N-gon
>> N=6; %Subject to vary!
>> Fi=pi/5; Alf=(0: 2*pi/N: 2*pi)+Fi;
x=cos(Alf); y=sin(Alf);
>> fill(x,y, 'g'), axis equal
```

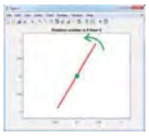
```
>> %Manual creaqtion of an N-star
>> N=18; a=0.5; %Subject to vary!
>> Fi=pi/5; Alf=(0: 2*pi/N: 2*pi)+Fi; x=cos(Alf); y=sin(Alf);
x(1:2:end)=a*x(1:2:end); y(1:2:end)=a*y(1:2:end);
>> fill(x,y, 'm'), axis equal
```

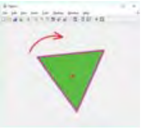
From ‘manual algorithms’ to Programming!

- 4 -

Programming

1. **No “boring Loop” *FOR***





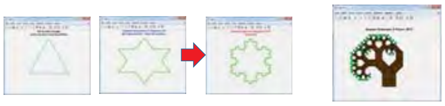
for … end
If … else … end
while … end
switch … case .. end

2. **“No boring” *IF -- ELSE***

m*-Program *Welcome.m

…. And the music at the end….

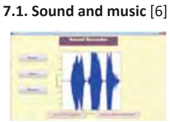
3. **Simple Fractals** (Koch snowflake, Pithaguras)

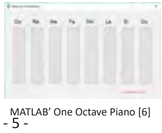


4. **Basic standard algorithms** (*max*, *min*, *Order*, *WordOrder*)
5. **Complex algorithms and Data Types**
6. **Graphical User Interface (GUI)**

7. Own student’ Discoveries for their Term Papers

7.1. Sound and music [6]



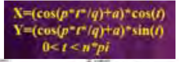


MATLAB’ One Octave Piano [6]

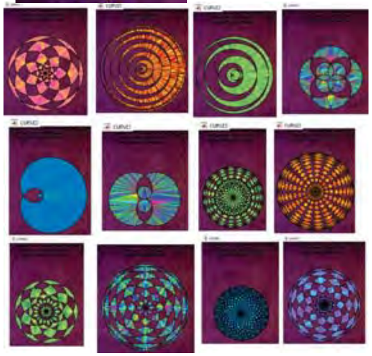
- 5 -

Term paper Discoveries; student’ educational Programs

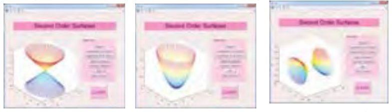
7.2. Moving (and singing) parametric curves [9]



$X=(\cos(p*t*/q)+a)*\cos(t)$
$Y=(\cos(p*t*/q)+a)*\sin(t)$
$0<t<n*pi$



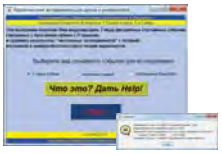
7.3. Programs that test student’ knowledge (Secondary Order Surfaces)

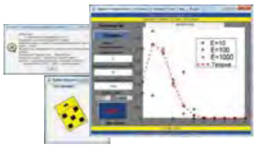


Easy Programming is the way to everything!

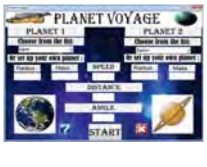
- 6 -

7.4. Probability Research [7]





7.5. Programs with physical meaning



More details see in neighboring poster.

Easy Programming is the way to everything!

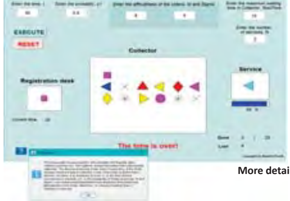
- 7 -

8. “Own Discoveries” on elder teaching years

8.1. “Information and Coding Theory”

Digital Laboratory of Information Theory [9]

8.2. “Imitational and Stochastics Modelling”



More details see in neighboring poster.

- 8 -

Conclusion

- ➔Programming governs the World of Education, [3]. Why not to use MATLAB’ Easy Programming in education of Engineers?
- ➔With MATLAB, we realize methodology of “Student’ Own Discoveries” [4-10].
- ➔This corresponds to best World methodology suggestions [3,11,12] etc.
- ➔We should save student’ curiosity and ‘shining eyes’ till the end of education.

References

1. https://en.wikipedia.org/wiki/Discovery_learning
2. https://de.wikipedia.org/wiki/Entdeckendes_Lernen (in German)
3. Wolfram S. A New Kind of Science. Wolfram Media, 2002. – 1192 p.
4. Gayev Ye.A., Nesterenko B.N. MATLAB for Math and Programming. Textbook. Zaporizhzhya: Poligraph, 2006. - 101 pp. https://www.academia.edu/19707263/Math_and_Programming_with_MATLAB
5. Gayev Ye.A., Azarskov V.M. Modern Programming: Textbook, part 1, 2014; part 2, 2017. (in Ukrainian, https://www.mathworks.com/support/books/modern-programming-gayev0.html)
6. Gayev Ye.A., Rozhok O., Ovcharchin N. Sound and music in programming curriculum. Engineering of softs, 2014, № 3(19), c. 41 - 48. (in Ukrainian, http://ecobio.nau.edu.ua/index.php/IPZ/article/view/9714/12319)
7. Gayev Ye.A., Martich M., Tarak G. Occasional events modelling programs for learning programming and mathematics. Informational technologies in Education, 2015, № 23, c. 30-42. (in Russian, http://ite.kspu.edu/webfm_send/829)
8. Gayev Ye.A. et all. Digital Laboratory of Information Processes Theory: an innovative educational approach. Матеріали XIII Міжнародної науково-технічної конференції “Avia-2017”. April 19-21, Kyiv: NAU, 2017. 9.42, p. 638--641. (http://avia.nau.edu.ua/doc/avia-2017/AVIA_2017.pdf)
9. Gayev Ye.A., Malinina D. Parametric rose as an object in mathematics, programming and aestaetics. Informational technologies in Education, №1, c.9-24, (in Russian, http://ite.kspu.edu/ru/webfm_send/929)
10. Gayev Ye.A., Kalmikov V.V. The Travelling Salesman Problem in the engineering education programming curriculum. Proc. of Nat. aviation university, 2017. (http://jrnl.nau.edu.ua/index.php/visnik/article/view/11989/16164)
11. Moler S. Experiments with MATLAB. (https://www.mathworks.com/moler/exm.html)
12. Chen K., Giblin P., Irving A. Mathematical explorations with MATLAB. – Cambridge University Press, 1999. - 322 p.

In this poster we demonstrated how we realized our educational approach in Aerospace Control Systems Department of National Aviation University (Kyiv, Ukraine). More details see in neighboring poster.

Easy Programming is the way to happy student’ Discoveries!

- 9 -