Modern Teaching Methods in Physics with the Aid of Original Computer Codes and Graphical Representations

Anisoara Ivanov[1], Andrei Neacsu[2*]

[1]*Grigore Moisil High School, Tulcea - Romania*
[2]*National Institute for Nuclear Physics and Engineering, Bucharest-Magurele, Romania*
[*]*E-mail: nandrei@nipne.ro*

(Received 07.05.2011 Accepted 10.06.2011)

Abstract

This study describes the possibility and advantages of utilizing simple computer codes to complement the teaching techniques for high school physics. The authors have begun working on a collection of open source programs which allow students to compare the results and graphics from classroom exercises with the correct solutions and further more to be able to easily interact with the source code in order to modify and create their own software. We believe that teaching and learning physics needs to benefit from the modern developments in computer sciences, thus creating a connection between textbook exercises and the use of computers is desirable. The personal experience of the authors suggests that solving a physics problem with the aid of a computer leads to a better understanding of the physical properties of the systems involved in the exercise.

Keywords: physics exercise, software, instructional techniques, high school physics

Introduction

Modern teaching-learning-assessment methods offer a beneficial opportunity of organizing a thorough teaching, based on interactive-engagement strategy. The systematic use of modern methods involves effective communication and constructive relationships, in which all who take part in discussions obtain cognitive, affective-motivational, attitudinal, social and practical application benefits.

The use of modern methods does not mean giving up traditional teaching system, but updating it. At the end of modern teaching-learning activities, school assessment results are concerned not only with the students' acquisitions in the cognitive domain, knowledge, skills, abilities, but also with the full spectrum of behaviors that contribute to student personality development: effective and psychomotor behavior, some extra-educational outcomes, some with direct influence on school results, and self-assessment skills. The teaching act, that focuses on meeting students' needs and interests, is based on the game theory. The educational process should arouse interest for the satisfaction of needs and should trigger reactions directed towards the goal.

Romanian teachers agree that science is open to new suggestions and ideas in the context in which education must respond to national and international social realities. The Romanian Educational System, considered a national priority, has undergone an intense reformation process, so that education should adapt to a democratic society and market economy. The teacher, freed from the coercive dictum MAGISTER DIXIT, should become the professional who knows how to use tools and material basis.

Modern developments in computer languages and computer hardware have enabled the use of programs, codes and libraries with minimal programming skills. It is now possible to easily create, distribute and modify source codes by educators and students for solving or illustrating textbook exercises of high school physics. The computer simulations proposed in this article are intended to complement the laboratory experiments and classroom exercises, in order to verify, confirm and clarify the acquired results. We will focus on a single program developed for the mechanics lessons, which solves and represents graphically the equations for the free fall of a body in vacuum (1). The physical quantities involved in this simple exercise are mass, length, time, speed, momentum and energy. A very easy and convenient way to program the equations which link these quantities was to write them in a friendly language for mathematical expressions such as Fortran (GNU Fortran project, 2011), allowing students to access the formulas in their natural expressions. The purpose is allowing students to see, understand and modify these expressions without difficulty. For the same reason, the graphics part is achieved through the use of Grace (Grace, 2008), which utilizes simple expressions and commands, accessible from the source code.

Description of the program

In order to develop a simple program suitable for students and teachers, a list of possibilities was taken into account. All the software needed and related to this project had to be freeware and open source, the program needed to be easy to use, modify and compile, the hardware requirements had to be as low as possible and it needed to be usable on any operating system. To comply with these intended requirements the choice for the programming language was FORTRAN using the GNU complier, the graphics port is through the library of GRACE, known also as XMGRACE and all these are available for Linux, Unix, Mac OS and Windows. When compiling in Windows using the GRACE library it is needed to have an X server (XFree86 project, 2011;X-org) running and it is best if compiled and run under Cygwin (Cygwin project, 2011).

Input

The program requires the user to enter four input values for the parameters describing mass, height, gravitational constant and the number of data points for the graphical representation of the instantaneous values. The units are relative and the student must keep in mind the scale used in order to accurately interpret the results.

Output

The default output results are the values for the final velocity, total time, average velocity, initial potential energy, final kinetic energy, and graphic which represents the instantaneous kinetic and the potential energies as a function of elapsed time. This selection of output values demonstrates the student the conservation of energy and the fact that in vacuum speed and total time are independent of mass. It is possible to choose any other output values and plots, such as momentum, instantaneous speed, ratios of energies or any other desired initial, final of instantaneous value.

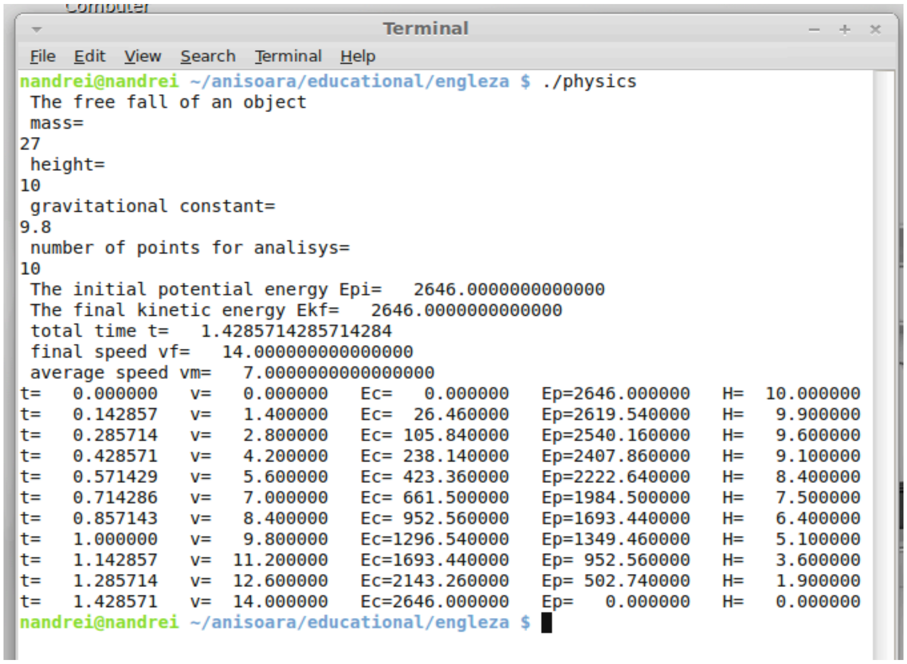
Usage

After entering the input values, the initial and the final parameters are computed. The number of requested data points generates the discretization of the time variable necessary for the instantaneous values. Because iterations are involved, the expressions for the kinetic and the potential energies are declared as functions and invoked whenever required via their input parameters. During the iterative loop, the instantaneous velocity is computed, the distance traveled, the current height and these values enter the general expressions for the kinetic and the potential energies.

For an easy use of the source code in this package and the extensions, which will follow, the project is developed in modules containing callable functions for all the physical expressions. For example we create a function “Potential energy”, another function “Kinetic energy”, a function for “Momentum”, such that these functions can be used in more complicated expressions and invoked at any time with their set of variables (mass, speed, height, gravitational constant). Further development will add more expressions and thus more capabilities to the project. Although the idea itself is not new and such computer programs have been made before, we believe that designing and developing this project brings benefits into the physics lessons. The fact that everything is as simple as possible for the student and for the teacher brings versatility to the software package and enables further development and expansion beyond what is initially provided. I may very well serve as a starting point for other projects of physics and mathematics which fit this idea of calling simpler expressions into more complicated formulas. Another interesting aspect observed from personal experience, is that presenting physics with more than one method leads to an easier understanding. Some students spend more time using computers and are attracted to programming and algorithms; allowing them to interact with the source code opens new possibilities for their curiosity in science. The whole philosophy behind the project is not teach programming or IT related subjects, but rather make use of what the students already know and to attract them towards physics by offering them a new perspective over the classroom exercises and experiments, after the textbook lesson was taught. The simple instructions, needed to compile the programs and how to run it, are included in the package distributed.

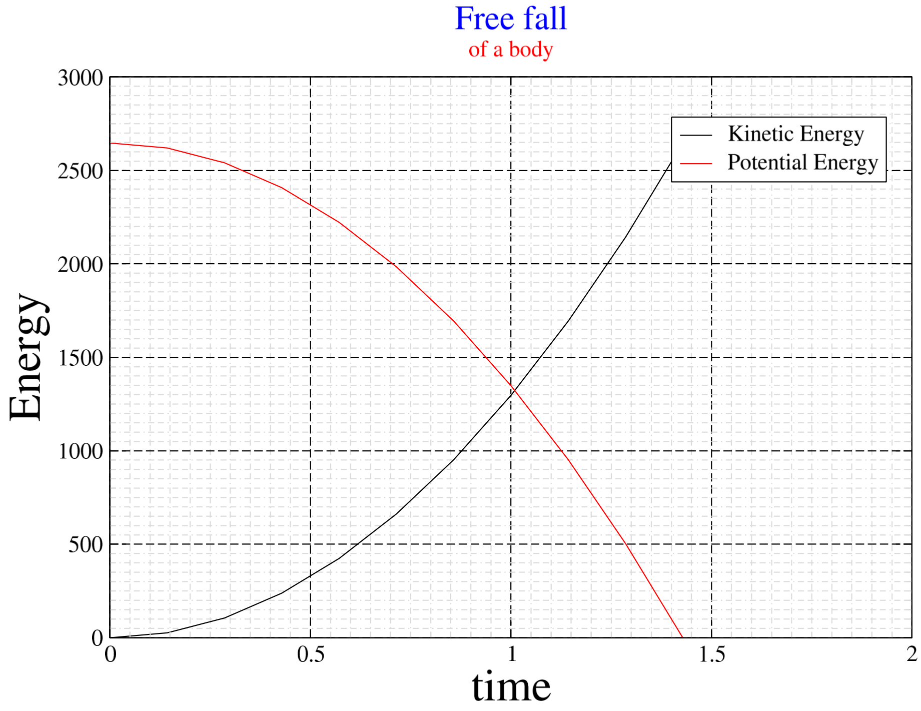
An example

The following figures show the use of the program with a set of input parameters and the obtained results.



```
nandrei@nandrei ~/anisoara/educational/engleza $ ./physics
 The free fall of an object
 mass=
27
 height=
10
 gravitational constant=
9.8
 number of points for analisys=
10
 The initial potential energy Epi=   2646.0000000000000
 The final kinetic energy Ekf=   2646.0000000000000
 total time t=   1.4285714285714284
 final speed vf=   14.000000000000000
 average speed vm=   7.0000000000000000
t=   0.000000   v=   0.000000   Ec=   0.000000   Ep=2646.000000   H=  10.000000
t=   0.142857   v=   1.400000   Ec=  26.460000   Ep=2619.540000   H=   9.900000
t=   0.285714   v=   2.800000   Ec= 105.840000   Ep=2540.160000   H=   9.600000
t=   0.428571   v=   4.200000   Ec= 238.140000   Ep=2407.860000   H=   9.100000
t=   0.571429   v=   5.600000   Ec= 423.360000   Ep=2222.640000   H=   8.400000
t=   0.714286   v=   7.000000   Ec= 661.500000   Ep=1984.500000   H=   7.500000
t=   0.857143   v=   8.400000   Ec= 952.560000   Ep=1693.440000   H=   6.400000
t=   1.000000   v=   9.800000   Ec=1296.540000   Ep=1349.460000   H=   5.100000
t=   1.142857   v=  11.200000   Ec=1693.440000   Ep= 952.560000   H=   3.600000
t=   1.285714   v=  12.600000   Ec=2143.260000   Ep= 502.740000   H=   1.900000
t=   1.428571   v=  14.000000   Ec=2646.000000   Ep=   0.000000   H=   0.000000
nandrei@nandrei ~/anisoara/educational/engleza $ 
```

Figure 1. *The input and the text output of the program*



Fri Jul 1 21:39:23 2011

***Figure 2.** The graphic output generated by the program with calls to the " libgrace_np" library [].*

Conclusions

This package was intended to be simple and easy to develop further into more complex codes for problems beyond the one presented here. The fact that it is very modular allows users and developers to choose various bits and integrate them in their own ideas. The authors intend to expand the capabilities and design more functionality until all aspects of middle school and high school physics are covered.

We believe that it is fun to solve exercises and problems with the program presented here and the students benefit from it in various ways: they can check their results with the ones obtained on the computer, can change anything in the code to suit their needs and curiosity and most importantly, they get an additional perspective on how to solve physics related calculations.

References and Links

The “Mechanics” package available at http://smile.theory.nipne.ro/nandrei/educational

GCC team. *GNU Fortran project 2011*. Retrieved from http://gcc.gnu.org/fortran

Grace package and library. The FTP home of Grace 2008. Retrieved from http://plasma.gate.weizmann.ac.il/Grace

XFree Project. XFree86. Retrieved from http://www.xfree86.org

X-org. Retrieved from http://www.x.org

Cygwin project. Cygwin suite 2011. Retrieved from http://www.cygwin.com