Dmytro Pokryshen
Associate Professor, PhD, Head of the Department of Natural and Mathematical Disciplines and ICT in Education
K. D. Ushynskyi Chernihiv Regional Institute of Postgraduate Pedagogical Education, Chernihiv, Ukraine
pokryshen@ukr.net
ORCID: 0000-0001-9572-413X

FINISHED SOFTWARE PRODUCTS IN TRAINING SPECIALISTS OF IT INDUSTRY

Abstract. The article is devoted to the problem of training specialists in the IT industry, the experience of learning the basics of programming students. The purpose and tasks of studying programming courses are determined. The purpose of this article is to identify and select fragments of developed software products and applied technologies that contribute to the formation and development of professional competencies of IT professionals. Modern application interfaces use objects and events with them. Everything about the frontend has a nice interface design. That is why modern training programs for computer science teachers and software engineers are focused on the study of object-oriented programming. Thanks to modern ICT tools, all subjects of the educational process have the opportunity to join modern scientific research, see the results and determine the directions of their own research to improve educational programs for engineers and educators. The article reviews software applications and fragments of their software code that contribute to the formation and development of professional competencies of IT professionals. Among the software applications selected Optics (for building models of optical phenomena), GRAPH (for calculating electrical circuits and graphs), PhysHelper (demonstration of models of physical phenomena: conservation law, kinematics, dynamics and others). The Optics example demonstrates how to describe an object, create an array, process it, and one way to implement dynamic constructions in Pascal in Delphi. GRAPH uses the Dijkstri algorithm. The essence of this algorithm is to find the shortest paths from the initial vertex to all the others. The program forms an array of distances to each of the vertices. The distance to vertices that are in another component of connectivity is defined as infinity. PhysHelper was created using basic tools such as JavaScript (Electron framework and p5 and KaTeX libraries), HTML and CSS. The selection of selected technologies and approaches is substantiated.

Keywords: STEM; scientific and technical creativity; programming; computer code; education

Statement of the problem in general. At the time of the reforming process in Ukraine, the educational sphere is not apart this process; it is the source from which all begin. We already have the first results of the implementation of the concept of "New Ukrainian School", the formation of a new structure of general secondary education (primary, gymnasium, lyceum). One of the main innovation is students' preparation for the future social and professional life, their preparedness for performing non-standard tasks and the formation of a competent employee. Thus, the system of training and retraining of specialists (0 14.09 Informatics, 121 Software Engineering, 122 Computer Science and Information Technology, 123 Computer Engineering) faces new requirements.

For many years, objects and events with them have been using in application interfaces. Today there are almost no environments and programming languages that use console mode. That is why current training programs for computer science teachers and software engineers are focused on the study of object-oriented programming.

Computer and information sciences teachers, who underwent training courses only fifteen years ago, already need retraining and advanced training to understand modern programming techniques and better prepare students for further study and professional activities.

Analysis of recent research and publications. In the works of V. Yu. Bykov (Bykov, 2009)*,* M.I. Zhaldak (Zhaldak, 2015, pp. 3-27), S.O. Semerikov (Semerikov, 2009), Y.V. Trius, Y.S. Ramsky, G.Y. Tsybko, it was examined the issues of preparation of future computer and information sciences teachers for database design and use of database management systems. The research of A.K. Atamanchenko and A.A. Davydenko is devoted to the problem of

preparing students for participation in tournaments and Olympiads (Davydenko, 2019, pp.34-37), and others. Gerasymchuk N.O., D.A. Pokryshen and others regard the problems of participants preparing for taking part in computer science and information technologies competitions.

Modlo E., Semerikov S., Markova O. propose to use mobile Internet devices for the formation of competencies for solving professional problems by means of information and communication technologies (Modlo, 2018). The example of using the services Scilab, SageCell, Google Sheets, Xcos on Cloud shows their effectiveness for the formation and development of general professional competence of an electromechanical engineer.

The authors Tkachuk V.O., Yechkalo Y., Semerikov S.O., Kislova M., Khotskina V. (Tkachuk, 2020), in their research works, propose to use services ClassMarker, EasyTestMaker, Google Forms, iSpring QuizMaker, Kahoot!, MyTestXpro, Plickers, ProProfs for conducting surveys and controls measures.

How to use computer simulation of the tournament on game algorithms in the process of learning the basics of algorithmization and programming by pre-service IT specialists is discussed in Liudmyla E. Gryzun, Oleksandr V. Shcherbakov and Svitlana H. Lytvynova (Gryzun, 2021).

Due to the modern means of ICT, all subjects of the educational process have the opportunity to join modern scientific researches, look at the results and determine the directions of their own researches to improve the training educational programs for engineers and educators. In our opinion, good examples that can inspire for future activities are such resources as Inria (Inria, 2021), NASA (NASA, 2021).

The Inria project is dedicated to the study of the perspective areas such as: Quantum Algorithms and Computing, High Performance Computing, Digital Education, Artificial Intelligence, Software, Frugal digital innovation, Digital health, Data Sciences, Digital Security (Inria, 2021).

The works of Andriy Davydenko and Dmytro Pokryshen are devoted to the development of applied software products that correspond to the STEM-education concept as the authors have been occupied in the field of the development of students' creative abilities for many years. The work (Davydenko, 2019, pp.34-37) is about the program's research tool of the rays passing in the various optical devices (software Optics) is overviewed the process of the creation of a device for studying the mechanical motion of bodies using the computer-manipulator sensor (software Dynamic Mouse). Also the issue of introduction of STEM-axis by means of ICT was considered in the research Natalia Lukychova, Nataliia Osypova and Galina Yuzbashva (Lukychova, 2011, pp.39-55).

Selection of previously unsolved parts of the general problem to which the article is devoted. The article continues the research of previous works (Davydenko, 2019, pp.34-37) and (Pokryshen, 2019, p.55-62). The main purpose of this work is to review the developed software products and technologies that are used as tools for the formation of professional IT competencies of computer and information sciences teachers and future software engineers.

Formulation of the goals of the article. The identification and selection of the made pieces of software products and applied technologies which contribute to the formation and development of professional competencies of IT-industry specialists.

Presenting of the main material. The educational program for the training of IT specialists has some components among which is the study of different programming languages of low and high levels, corresponding to the paradigm of object-oriented programming, SQL request language and tools for integrated use of various technologies (LAMP , XAMP, etc.).

The purpose of studying such courses is forming a modern level of programming competencies, gaining practical skills in the developing of software application for solving various problems in practice use, understanding of the internal software engineering,

developing a creative approach to solving non-trivial problems, improving of professional skills, learning new technologies and means.

The main tasks of studying such courses are:

- the development of information competencies necessary for qualified and effective use of modern information and communication technologies in educational, cognitive and professional activities;
- the development of skills of independent work on mastering of educational material, purposeful search and systematization of necessary data;
- mastering practical skills of creating, designing of the application and training software, integration of objects with databases;
- the formation of skills for the object-oriented interfaces development of application programs and databases.

During the study of the material various simplified examples are used and practical projects are developed. As practice shows, the quality of project implementation depends on the choice of topic and tasks. The ability to anticipate the problem, work through all its aspects and find the right algorithms for its solving requires high intellectual and creative levels.

The same problems arise in preparing pupils and students for intellectual IT competitions. The problem of students' preparing for the participation in national competitions and tournaments has always been and is still relevant nowadays, as it is necessary to go beyond the school curriculum, methods and forms of teaching, encourage pupils' creativity, use heuristic approaches and direct cognitive processes to maximum effectiveness.

As it was testified by the experience of the carrying out students' Olympiad and distance tournament in Information Technology, the participants have had the most difficulties in solving problems of databases, and precisely: not so much with knowledge of functions and parameters, but with the process of designing and developing structure. In case of incorrect development of the database structure, the system of queries, reports, forms and macros will not be accurate.

Basing on the analysis of the tasks for Information Technology tournaments and competitions it was proposed the methodology of designing and developing the structure of information-analytical systems.

The need of fundamental approach to the content of teaching the computer science disciplines in pedagogical higher educational institutions in order to develop the professional mobility of Computer Science and Information Technology teachers is discussed in the works (Zhaldak, 2015, pp.3-27), (Semerikov, 2009). Researchers emphasize the need to teach more general and deeper topics, due to the rapid pace of development of Information Technologies and the constant renewal of the education domain. However, in our opinion, it is also reasonable to use ready-made software products for studying and explanation the topics during the trainings of IT-industry specialists.

We propose to look at some examples of fragments of the code applications developed by the author and under his guidance of students from secondary schools at different times.

Software tool Optics developed in the environment of Delphi 10.1 Berlin.

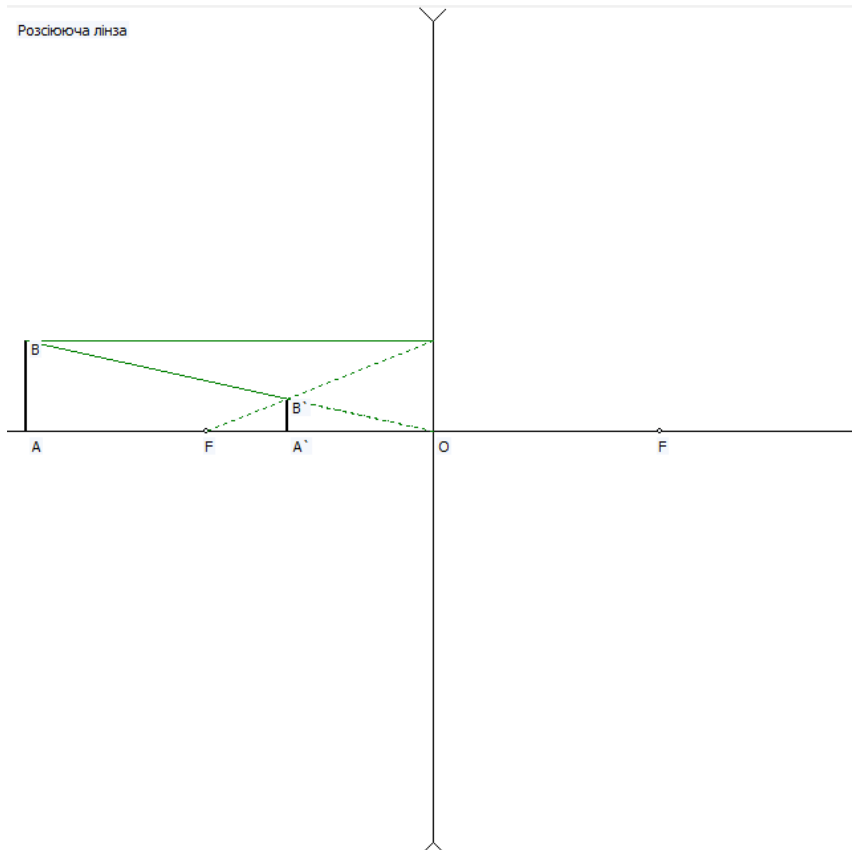Examine the path of rays along with Dissemination init lens (Figure 1):

Fig. 1 . Construction of the image in the lens

When constructing an image, two rules are used:

1. A beam passing through the optical center of the lens (O) that does not refract.

2. The beam passing parallel to the main optical axis of the lens, after refraction will begin to move away from the main optical axis; with continued refracted beams passes through the main focus (F).

Formulas we need:

$$\frac{1}{d}+\frac{1}{f}=\frac{1}{F} \quad (1); \qquad f=\frac{F \cdot d}{d-F} \quad (2); \qquad H=\frac{h \cdot f}{d} \quad (3).$$

1. Lens formula; 2. Distance to the image; 3. Image height.

For descriptions of objects (objects) used type of **record** .

```
type
    objects=record
    name:string[20];
    focus,vidstan,vusota,vidstanpisla,vusotapisla :real;
    n1,n2, kytpad, kytzal, kytpruzm, kytvuhid: real;
        end;
var
 Form1: TForm1;
 obj: array [1..100] of objects;
 fobj: file of objects;
 filename: string;
 x0,y0:integer;
 i:byte;
 h,f,zoom,zoomh,max,maxh:real;
```

```
krok:real;
```

Procedures and creating and deleting objects:

```
procedure TForm1.CreateObject(Sender: TObject);  // Create objects
begin
  if listbox1.ItemIndex=0 then linz.show;
  if listbox1.ItemIndex=1 then dzerkal.show;
  if listbox1.ItemIndex=2 then mezha.show;
  if listbox1.ItemIndex=3 then prizm.show;
end;
procedure TForm1.DeleteObj(Sender: TObject);  // Delete objects
var i, n:byte;
begin
  if listbox2.itemindex>=0 then begin
  n:=listbox2.items.count;
  if n<>0 then begin
  for i:=listbox2.itemindex+1 to (listbox2.items.count-1) do obj[i]:=obj[i+1];
  listbox2.Items.clear;
  for i:=1 to (n-1) do   ListBox2.Items.add(IntToStr(i)+': '+obj[i].name);
          end;

                  end;
end;
```

A fragment of the procedure of dynamic image construction

```
  zoom:=1;
   InvalidateRect(0, nil, true);
  if listbox2.items.count=0 then ShowMessage('No object selected for construction') else
                      begin
   Image1.Picture.Graphic:=nil;
       sitka;    // сітка побудови
   if obj[i].name='Лінза' then begin
                     // construct an image in the lens
                     form1.Canvas.Pen.Color:=clblack;
                     form1.Canvas.Pen.Style:=pssolid;

                     max:=abs(obj[i].vidstan);
                     maxh:=abs(obj[i].vusota);
                      if abs(obj[i].focus)>=max then max:=abs(obj[i].focus);
                      if abs(obj[i].vusota)>=max then max:=abs(obj[i].vusota);

                      if abs(obj[i].vusotapisla)>=maxh then maxh:=abs(obj[i].vusotapisla);
                      if abs(obj[i].vidstanpisla)>=max then max:=abs(obj[i].vidstanpisla);
                      if maxh>=max  then zoom:=(image1.Height/2-15)/max
                              else zoom:=(image1.Width/2-15)/max;

                     while (maxh*zoom)>=(image1.Height/2-15) do zoom:=zoom*9/10;     //
autozoom

                     // Побудова лінзи
```

```
            form1.Canvas.Pen.Style:=psSolid;
            form1.Canvas.Pen.width:=1;
            form1.Canvas.moveto(0,y0);        // gorizont
            form1.Canvas.lineto(image1.Width,y0);
            form1.Canvas.moveto(x0,10);       // vertical
            form1.Canvas.lineto(x0,image1.Height-10);
               // focus
            f:=obj[listbox2.ItemIndex+1].focus;
            form1.Canvas.ellipse(x0-round(f*zoom)-2,y0-2,x0-round(f*zoom)+2,y0+2);
            form1.Canvas.ellipse(x0+round(f*zoom)-2,y0-
2,x0+round(f*zoom)+2,y0+2);
   //
   form1.Canvas.Pen.width:=2;
   form1.Canvas.moveto(x0-round(obj[i].vidstan*zoom),y0);
   form1.Canvas.lineto(x0-round(obj[i].vidstan*zoom),y0-round(obj[i].vusota*zoom));
  //
   form1.Canvas.TextOut(x0-round(obj[i].vidstan*zoom-5),y0-
round(obj[i].vusota*zoom),'B');
   form1.Canvas.TextOut(x0-round(obj[i].vidstan*zoom-5),y0+5,'A');
   form1.Canvas.TextOut(x0+5,y0+5,'O');
   form1.Canvas.TextOut(x0+round(obj[i].focus*zoom),y0+5,'F');
   form1.Canvas.TextOut(x0-round(obj[i].focus*zoom),y0+5,'F');

   form1.Canvas.Pen.width:=1;
   form1.Canvas.Pen.Color:=clgreen;
   for j := round(obj[i].vidstan*zoom) downto 1 do begin
                            Pause;
                            form1.Canvas.moveto(x0-round(obj[i].vidstan*zoom),y0-
round(obj[i].vusota*zoom));
                            form1.Canvas.Lineto(x0-j,y0-round(obj[i].vusota*zoom));
                            form1.Canvas.moveto(x0-round(obj[i].vidstan*zoom),y0-
round(obj[i].vusota*zoom));
                            form1.Canvas.Lineto(x0-j,y0-
round(j*obj[i].vusota/obj[i].vidstan));

                            end;
```

The example of this software product demonstrates a way to describe an object, create an array, process them, as well as one of the ways to implement dynamic constructions in Pascal in the Delphi environment.

The software basis of the GRAPH software was the Qt environment and the C ++ language. The files created in GRAPH have different extensions. *. praph - for electrical circuits and *. graph - for graphs.
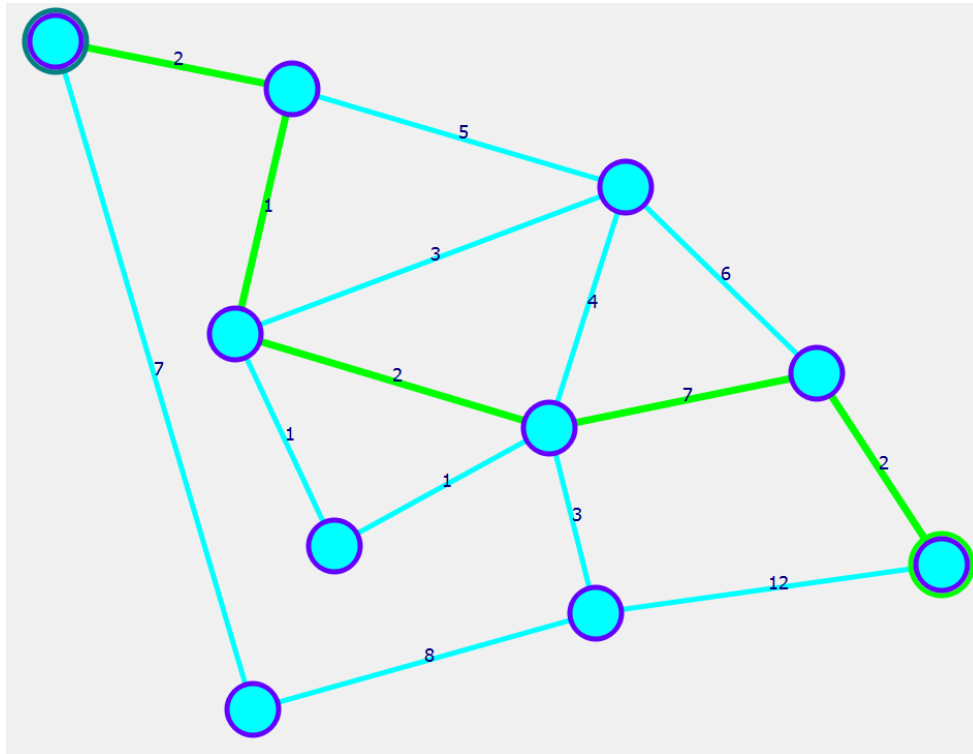
Fig. 2 . GRAPH software (shortcut)

A piece of code that defines the shortest path between vertices with a load on the edges.

```
weight = (double*)malloc(n*sizeof(double));
    if(!weight){ qDebug() << "Memory error!\n"; return;}
    for(i=0;i<n;i++)
      weight[i] = h;
    vector<int> call;
      weight[x3]=0;
      call.push_back(x3);
      j=0;
      for(;;){
        a=call[j];
        for(i=0;i<n;i++){
          if(i==a||mus[a][i]==-1)
            continue;
          if(weight[i]>weight[a]+mus[a][i]){
            weight[i]=weight[a]+mus[a][i];
            call.push_back(i);
          }
        }
        if(j==(int)call.size()-1)
          break;
        j++;
      }
      if(weight[y]!=h){
        ui->label_4->setText(QString :: number(weight[y]));
      }
      else{
        ui->label_4->setText(tr("There are no way to this point"));
```

```
    return;
}
ways.clear();
euler_path.clear();
search_way_from(y,{});
ui->label_17->setText(QString :: number(ways.size()));
//que.clear();
way_pos = 0;
qDebug() << "Weight : " << weight[y] << " " << weight[x3] << "\n";
qDebug() << "Ways : " << ways.size() << "\n";
free(mus);
free(weight);
drawing::update();
```

In GRAPH uses are Dijkstra's algorithm . The essence of the algorithm is to find the shortest paths from the starting vertex to all others. The program forms an array of distances to each of the vertices. The distance to the vertices that are in another component of connectivity is defined as infinity.
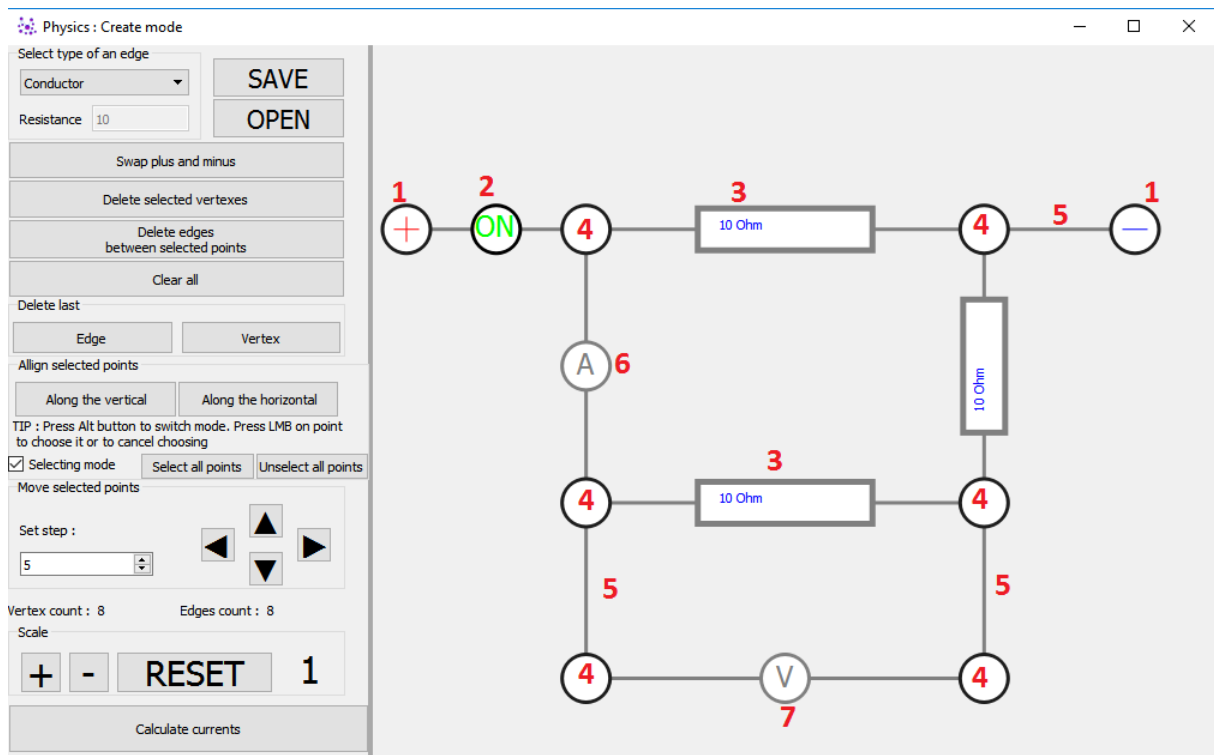

Fig. 3. Elements of an electric circuit in GRAPH

GRAPH software works in two modes: graphs and electrical circuits. The specified code snippet can be used in both modes.

The developed PhysHelper program was created using such basic tools as JavaScript (Electron framework and p5 and KaTeX libraries), HTML and CSS. This framework was chosen because of its possibility to be used for the applications developing; and it is compatible with platforms like MacOS, Windows and Linux (Jantjies, 2014).

Electron includes Node.js for work with back-end and rendering library from Chromium. There are two types of processes in this framework. The main process starts the script main from the package.json, this script can display the GUI creating web pages. The application

Electron always has one main process. Each web page runs in its own process, called the renderer process. Chromium multiprocessor architecture is used to display them. The main process creates web pages by instantiating Browser Window. Each instance of Browser Window launches a web page in its own process. When destroying an instance, the corresponding the renderer process is stopped. The main process controls all web pages and the corresponding renderer processes. Instead, each rendering process is isolated and applies only to the web page that is running in it. In web pages, calling the native GUI API is prohibited, so to perform GUI operations on web pages, the renderer process "communicates" with the main to request the execution of these operations of the main process. The API Node.js can also be used on web pages allowing interacting with the operating system at a low level. Using HTML and CSS markup language allows creating a user-friendly interface that would be more difficult to achieve by other means. P5 provides a flexible interface for creating and managing animated graphics, and also contains convenient mathematical primitives that are often used to create physical simulations. KaTeX is a library for computer printing to display formulas. The **npm** package manager and an electron-packager were also used. Git was chosen as the version of the control system.

Implementation of the experiment "Body thrown at an angle to the horizon"

This experiment considers the motion of bodies along a parabola under the action of gravity. The experiment considers the motion of a body thrown horizontally and the motion of a body thrown at an angle to the horizon, and from any height. The program ignores the air resistance and considerates an idealized model, that's why a body moves uniformly along the axis Ox and along the axis Oy - with the acceleration g, given to the body by the force of gravity. So, the coordinates of the body are: $x(t)=x0+ v0xt$, $y(t)= y0+ v0yt+ gyt22$, where $v0x=v0cos$, $v0y=v0sin$, – the angle at which the cast body. The entire flight time can be calculated by the formula

$$t=v0sin +v02 +2h0gg.$$

The user can pause conveniently the experiment to set the balls to the desired parameters and running it again to observe how they move at the same time and compare the parameters of the bodies' movement by using the plate in the upper right corner of the workspace.

Additional classes and methods have been created to implement this fragment of the program. The code snippet of the *class Ball* methods is shown in the listing:

```
 update() {
     this.vel.add(gravity);
     this.pos.add(this.vel);
     if (this.pos.y > ground-this.radius) {
       this.pos.y = ground-this.radius;
       this.vel.set(0, 0);
       return true;
     } else return false;
   }
   kick(x, y) {
     let mouse = createVector(x, y);
     mouse.sub(this.pos);
     mouse.div(20);
     this.vel = mouse;
     if (first) {
       vel1 = this.vel.copy();
       vel1.mult(rate/scl);
     } else {
       vel2 = this.vel.copy();
```

```
        vel2.mult(rate/scl);
    }
}
```

Implementation of the experiment "Law of conservation of momentum"

In this experiment, the laws of energy and momentum conservation of are considered. It is also neglected the friction force; the collision is quite elastic. For the case when the impact is non-central, we turn to the frame of reference, the axis Ox of which is directed along the line of the centres of the considered balls. During the impact, the internal forces will be directed only along the axis Ox, so the Y-component of both speeds will not change: $v1y'=v1y$, $v2y'=v2y$. Considering the mentioned above, writing the law of conservation of kinetic energy and the law of conservation of momentum, moving to the projections on the axis Ox, we obtain
$$v1x'=m1-m2v1x+2m2v2xm1+m2, \quad v2x'=m2-m1v2x+2m1v1xm1+m2.$$

Additional classes and methods have been created to implement this fragment of the program. The code snippet of the *class Ball* methods is shown in the listing:

```
collision(k) {
    for (let i = k+1; i < balls.length; i++) {
        let elem = balls[i];
        let dist = p5.Vector.sub(this.pos, elem.pos);
        if (dist.mag() < this.radius + elem.radius) {
            dist.setMag(this.radius + elem.radius);
            this.pos = p5.Vector.add(dist, elem.pos);
        }
        if (dist.mag() == this.radius + elem.radius) {
            let newVel1 = createVector();
            let newVel2 = createVector();
            let direct = p5.Vector.sub(this.pos, elem.pos);
            let angle = -direct.heading();
            this.vel.rotate(angle);
            elem.vel.rotate(angle);
            newVel1.y = this.vel.y;
            newVel2.y = elem.vel.y;
            newVel1.x = ((this.mass - elem.mass)*this.vel.x +
2*elem.mass*elem.vel.x)/(this.mass + elem.mass);
            newVel2.x = ((elem.mass - this.mass)*elem.vel.x +
2*this.mass*this.vel.x)/(this.mass + elem.mass);
            newVel1.rotate(-angle);
            newVel2.rotate(-angle);
            this.vel = newVel1;
            elem.vel = newVel2;
        }
    }
}
```

Conclusions. The applied character of these or those technologies, the formation of the domain specific process and necessary competences are presented through using ready-made complex software products and their fragments of the code. Future IT specialists scope to see the implementation area of the acquired knowledge and will be able to apply it immediately. The use of simplified abstract examples does not allow demonstrating the existing potential of modern software environments and applications. Moreover, using ready-made software products, students receive the necessary motivation to develop their own projects and start-ups.

Among the prospects for further research we see the analysis of software products developed by students. Based on the results obtained, we see the possibility of continuing education already on the basis of student projects.

REFERENCES

Bykov, V. (2009). Electronic pedagogy and modern tools of open education. Information technologies and teaching aids, 2009, № 5 (13). http://journal.iitta.gov.ua/index.php/itlt/article/view/177#.U-StDfl_tic (in Ukrainian)

Zhaldak, M. (2015). Problems of fundamentalization of the content of teaching information disciplines in pedagogical universities. Scientific journal of NPU named after M.P. Dragomanov, Series № 2: Computer - oriented learning systems: collection of scientific works, 17, 3-23 (in Ukrainian)

Semerikov, S. (2009). Theoretical and methodical bases of fundamentalization of training of computer science disciplines in higher educational establishments. Kyiv: NPU named after M.P. Dragomanov (in Ukrainian)

Davidenko, A. & Pokryshen, D. (2019). Software tool for studying the course of rays in various optical devices, Scientific Journal of NPU named after MP Drahomanov. Series 2: computer-based learning systems, 2019, Issue 21 (28), p.34-37 (in Ukrainian)

Lukychova, N., Osypova, N. & Yuzbashva, G. (2021). ICT and current trends as a path to STEM education: implementation and prospects. CTE 2021: 9th Workshop on Cloud Technologies in Education, December 17, 2021, Kryvyi Rih, Ukraine. April 16, 2022. http://ceur-ws.org/Vol-3085/paper16.pdf

Modlo, Ye., Semerikov, S. & Shmeltzer, E. (2018). Modernization of Professional Training of Electromechanics Bachelors: ICT-based Competence Approach. Proceedings of the 1st International Workshop on Augmented Reality in Education, Kryvyi Rih, Ukraine, October 2, 2018. CEUR Workshop Proceedings 2257, 148–172. February 25, 2022.

http://ceur-ws.org/Vol-2257/paper15.pdf

Tkachuk V., Yechkalo Yu., Semerikov S., Kislova, M. & Khotskina, V. (2020). Exploring Student Uses of Mobile Technologies in University Classrooms: Audience Response Systems and Development of Multimedia: ICT- based Competence Approach. CEUR Workshop Proceedings, Vol. 2732. http://elibrary.kdpu.edu.ua/bitstream/123456789/4126/1/20201217.pdf

Gryzun, L., Shcherbakov, O. & Lytvynova, S. (2021). Computer modeling of the tournament of game algorithms in the process of learning of basics of algorithmization and programming by pre-service IT-specialists. CTE 2021: 9th Workshop on Cloud Technologies in Education, Kryvyi Rih, Ukraine. April 16, 2022. http://ceur-ws.org/Vol-3085/paper14.pdf

Inria, the French research institute for digital sciences (2021). February 05, 2022. https://www.inria.fr/en

NASA. National Aeronautics and Space Administration (2021). February 05, 2022. https://www.nasa.gov

Pokryshen, D. & Oleksienko, S. (2019). The role and place of information-analytical systems in the training of computer science teachers, Open educational e-environment of modern University, № 6 (2019) Kyiv. 2019. p.55-62 https://openedu.kubg.edu.ua/journal/index.php/openedu/article/view/2414-0325.2019.6.5562

Jantjies, E. (2014). A framework to support multilingual mobile learning: A South African perspective. Dissertation, University of Warwick

ГОТОВІ ПРОГРАМНІ ПРОДУКТИ У ПІДГОТОВЦІ ФАХІВЦІВ ІТ-ГАЛУЗІ

Покришень Дмитро Анатолійович
Кандидат педагогічних наук, доцент, завідувач кафедри природничо-математичних дисциплін та інформаційно-комунікаційних технологій в освіті
Чернігівський обласний інститут післядипломної педагогічної освіти імені К.Д. Ушинського, Чернігів, Україна
pokryshen@ukr.net
ORCID: 0000-0001-9572-413X

Анотація. Стаття присвячена висвітленню проблеми підготовки фахівців ІТ-галуззі, досвіду навчання основ програмування здобувачів освіти. Визначено мету та завдання вивчення курсів з програмування. Метою цієї статті є визначення та добір фрагментів розроблених програмних продуктів та застосованих технологій, які сприяють формуванню та розвитку професійних компетентностей фахівців ІТ-галузі. У сучасних інтерфейсах прикладних програм використовують об'єкти та події з ними. Все, що стосується frontend має приємний дизайн інтерфейсу. Саме тому сучасні навчальні програми підготовки вчителів інформатики та інженерів-програмістів зорієнтовані на вивчення об'єктно-орієнтованого програмування. Завдяки сучасним засобам ІКТ всі суб'єкти освітнього процесу мають змогу долучитись до сучасних наукових досліджень, подивитись результати та визначити напрями власних досліджень для вдосконалення освітніх програм підготовки інженерів та освітян. У статті зроблено огляд програмних застосунків та фрагментів їх програмного коду, що сприяють формуванню та розвитку професійних компетентностей фахівців ІТ-галузі. Серед програмних застосунків обрано Optics (для побудови моделей оптичних явищ), GRAPH (для розрахунку електричних кіл та графів), PhysHelper (демонстрація моделей фізичних явищ: закон збереження, кінематика, динаміка та інші). На прикладі Optics демонструється спосіб опису об'єкту, створення масиву, їх опрацювання, а також один із способів реалізації динамічних побудов мовою Pascal у середовищі Delphi.У GRAPH використовується алгоритм Дейкстрі. Суть даного алгоритму полягає у пошуку найкоротших шляхів від початкової вершини до всіх інших. Програма формує масив відстаней до кожної з вершин. До вершин, які знаходяться у іншій компоненті зв'язності відстань визначається як нескінченність. PhysHelper була створена за допомогою таких основних засобів як JavaScript (фреймворк Electron та бібліотеки p5 та KaTeX), HTML та CSS. Обґрунтовано добір обраних технологій та підходів.

Ключові слова: STEM; науково-технічна творчість; програмування; комп'ютерний код; освіти