

ПАДРЫХОЎКА БУДУЧЫХ НАСТАЎНІКАЎ ІНФАРМАТЫКІ ДА РАСПРАЦОЎКІ КАНТРАЛЮЮЧЫХ ПРАГРАМ З ДАПАМОГАЙ МОВЫ ПРАГРАМІРАВАННЯ

У наш час у педагагічнай навуцы і практыцы праблема кантролю ведаў з дапамогай камп'ютэра надаецца вельмі вялікая ўвага. Таму задача падрыхтоўкі будучага настаўніка інфарматыкі да стварэння ім кантралюючых праграм з дапамогай мовы праграміравання з'яўляецца надзвычай актуальнай. У сваёй практыцы мы разглядаем гэту задачу як адну з тыповых прафесійных вучэбна-прадметных праблем.

Існуюць розныя метадыкі навучання той ці іншай дзейнасці, якія суадносяцца з рознымі фармуліроўкамі мэт навучання. Мы выкарысталі адзін з найбольш распаўсюджаных варыянтаў метадыкі фарміравання дзейнасці ў вучэбным працэсе [1]. Ён адносіцца да тых выпадкаў, калі неабходна навучыць дзейнасці, абагульнены змест якой навучэнцу невядомы. У гэтым выпадку асноўныя этапы навучання наступныя:

- матывацыйны этап;
- этап знаёмства навучэнца з узорам дзейнасці ў канкрэтнай сітуацыі або з узорам канчатковага прадукту дзейнасці;
- трэніроўка навучэнца ў выкананні дзейнасці ў канкрэтных сітуацыях;
- кантрольны этап.

Матывацыйны этап існуе для стварэння ў студэнтаў жадання ці патрэбы ў выкананні пэўных дзеянняў (выкананні пэўнага задання). У нашым выпадку мы абапіраліся на веды студэнтаў па курсах педагагікі, на іх вучэбны вопыт, на вынікі анкетавання школьных настаўнікаў. Акцэнт пры гэтым рабіўся на тое, што набытыя веды, уменні і навыкі павінны сістэматычна кантралявацца і рэгулявацца. Рашаючае значэнне пры гэтым адводзіцца бягучаму кантролю.

Далей студэнты знаёміліся з узорамі дзейнасці або яе канечнага прадукту і самой дзейнасці. Для гэтага выкладчык сам паказваў, як выконваць заданне ў пэўнай сітуацыі, ці адсылаў студэнтаў да вучэбнай літаратуры, дзе ёсць апісанне працэсу распрацоўкі кантралюючай праграмы для дадзенай сітуацыі.

Для трэніроўкі выкладчык прапаноўваў выканаць шэраг заданняў. Апорай пры выкананні дзейнасці пры гэтым з'яўляліся ўзоры тэкстаў кантралюючых праграм.

Кантроль сфарміраванасці дзейнасці складаўся з выканання заданняў студэнтамі ўжо без выкарыстання ўзораў. Само заданне змяшчала канкрэтную сітуацыю, знаёмую студэнту або незнаёмую – такую, з якой ён не сутыкаўся на папярэднім этапе. Калі студэнт паспяхова спраўляўся з незнаёмай сітуацыяй, то можна было пераканана гаварыць, што ён засвоіў дзейнасць. Ацэнку якасці распрацоўкі праграм праводзілі выкладчык і студэнты. Студэнты выконвалі кантралюючыя праграмы сваіх таварышаў і ацэньвалі іх.

Мы разглядалі два спосабы прад'яўлення інфармацыі карыстальніку пры распрацоўцы кантралюючых праграм.

Альтэрнатыва. У гэтым варыянце інфармацыя прад'яўляецца карыстальніку такім чынам, што ён павінен выбраць правільны адказ, які абавязкова прысутнічае на экране, з мноства магчымых альтэрнатыв, прад'яўляемых у кадры. Пры гэтым разглядаюцца розныя варыянты ажыццяўлення неабходнага выбару.

Стварэнне. Асаблівасцю прад'яўляемай інфармацыі ў гэтым выпадку з'яўляецца тое, што адсутнічае эталон для выбару і карыстальніку неабходна поўнасцю або часткова сфармуляваць правільны адказ.

Мы абмяркоўвалі розныя варыянты ўводу правільных адказаў, калі аналізавалі прапануемыя настаўнікам ў школах гатовыя кантралюючыя праграмы. У большасці кантралюючых праграм увод правільных адказаў ажыццяўляецца з дапамогай клавіятуры (увод нумара правільнага адказу, увод аднаго сімвала, увод фразы і г. д.), з дапамогай камп'ютэрнага прыстасавання «мыш» (адзначыць пстрычкай мышы правільны адказ, правесці курсорам мышы па правільным адказе і г. д.) або з дапамогай спалучэння працы гэтых прыстасаванняў (паставіць курсор «мышы» ў патрэбны пункт

і ўвесці сімвал і інш.).

Рэакцыя праграмы на ўведзены адказ бы- вае рознай: толькі ў канцы праграмы падво- дзіцца агульны вынік або выстаўляецца адзна- ка; толькі пасля кожнага адказу паведамля- ецца, правільны ён ці не; акрамя паведамлен- ня аб няправільнасці адказу паказваецца правільны варыянт і інш.

Асабліва варта адзначыць, што для рас- працоўкі дастаткова складаных камп'ютэрных праграм для школы падключаюцца высокапра- фесійныя праграмісты. Аднак у сучасных умо- вах навыкамі распрацоўкі прасцейшых кантра- луючых праграм павінны валодаць і будучыя настаўнікі інфарматыкі. Для рашэння гэтай задачы на фізічным факультэце БДПУ распра- цаваны спецыяльныя заданні да лабаратор- ных работ, створаны новыя лабараторныя ра- боты па фарміраванні ведаў і ўменняў рас- працоўкі прасцейшых кантралюючых праграм сродкамі моў праграмавання. Гэтыя лабар- торныя работы выкарыстоўваюцца як пры вывучэнні курса «Асновы праграмавання», так і на факультатывах і спецкурсах.

Прыклад прывядзём прыклад лабараторнай работы па распрацоўцы прасцейшай кантралюючай праграмы тыпу «Альтэрнатыва» на мове прагра- мавання Delphi для студэнтаў другога курса.

Мэта: вывучыць магчымасць распрацоўкі прасцейшай кантралюючай праграмы з выба- рам з мноства магчымых варыянтаў аднаго правільнага адказу.

Распрацуйце прасцейшую кантралюючую праграму з 7 пытанняў і з трыма варыянтамі адказу, адзін з якіх правільны. У прапануемай

Пытанне 1

Якую вадкасць выкарыстоўваюць для вадкасных тэрмометраў?

Варыянты адказаў

азот

вадку

спірт або ртуць

Было пытанняў 0

Усяго памылак 0

Рыс. 1

StaticText1 (St1)

StaticText2 (St2)

StaticText3 (St3)

StaticText4 (St4)

RadioGroup1 (R.G1)

Label1

Label2 Label3

Label4 Label5

Timer1

Рыс. 2

праграме пытанні і варыянты адказаў прад'яў- ляюцца выпадковым чынам. Пасля выбару ва- рыянта правільнага адказу пстрычкай «мышы» на экран выводзіцца паведамленне аб пра- вільнасці адказу, агульная колькасць прад'яў- леных пытанняў і агульная колькасць памылак (рыс. 1).

Для стварэння праграмы ў асяроддзі Delphi на полі формы неабходна размясціць наступ- ныя аб'екты (рыс. 2) з адпаведнымі ўласцівас- цямі (Properties), якія паказаны ў табліцы.

Табліца

Уласцівасці формы і яе аб'ектаў

Элементы	Уласцівасці	Значэнні
Form 1	Caption WindowState	ТЭСТ ПА ФІЗІЦЫ wsMaximized
RadioGroup (панка Standart)	Caption Coiums Item Name	Варыянты адказаў 1 (ўвесці 3 пустыя радкі) Rg1
Timer (панка System)	Enabled Interval Name	Faise 1500 Timer1
Label1 (панка Standart)	Caption	(Выдаліць усе сімвалы)
Label2	Caption	Было пытанняў
Label3	Caption	0
Label4	Caption	Усяго памылак
Label5	Caption	0
StaticText1 (панка Additional)	Caption Name	(Выдаліць усе сімвалы) St1
StaticText2	Caption Name	(Выдаліць усе сімвалы) St2
StaticText3	Caption Name	(Выдаліць усе сімвалы) St3
StaticText4	Caption Name	(Выдаліць усе сімвалы) St4

Для напісання тэкстаў працэдур неабходна для адпаведных аб'ектаў вызначыць падзею (Events):

Імя працэдур	Імя аб'екта	Падзея
FormShow	Form1	OnShow
FormCreate	Form1	OnCreate
Rg1click	Rg1	OnClick
Timer1timer	Timer1	OnTimer

Разгледзьце тэкст праграмы (без уступных аўтаматычных аб'яў), прычым звярніце ўвагу на аб'яву ўласных канстант і пераменных у раздзелах const і var.

implementation

```
{ $R *.DFM }
```

```
const k=8; var n,osh,c, pr,i:byte; z:set of byte;  
v,v1,v2,o1,o2,o3, t:string ;
```

```
procedure vopros;
```

```

begin {Спіс пытанняў, варыянтаў адказаў і
нумар правільнага адказу}
  {Пераменныя v1 і v2 выкарыстоўваюцца ў
выпадку доўгага пытання}
  v:='';v1:=''; v2:='';
  Case n of
  1:begin
  v:=' Адзначце адзінку вымярэння
тэмпературы ў сістэме СІ.';
  o1:=' Градус'; o2:=' Кельвін'; o3:=' Джоуль';
  pr:=2;
  end;
  2:begin
  v:'Якую вадкасць выкарыстоўваюць для
вадкасных тэрмометраў?';
  o1:='азот'; o2:='ваду'; o3:='спірт або ртуць';
  pr:=3;
  end;
  //... і г. д. 6 пытанняў
  end; { для каманды case }
end;{працэдуры фарміравання пытанняў
vopros}
procedure TForm1.FormShow(Sender: TObject);
begin {Вывад першага кадра на экран}
  i:=i+1; str(i,t); t:='Вопрос '+t; st1.Caption:=t;
  n:=random(k)+1; z:=z+[n];
  vopros;
  st2.Caption:=v; st3.Caption:=v1;
st4.Caption:=v2;
  rg1.items.Strings[0]:=o1;
  rg1.items.Strings[1]:=o2; rg1.items.Strings[2]:=o3;
end;
procedure TForm1.RG1Click(Sender: TObject);
begin
  c:=rg1.ItemIndex+1;
  if c=pr then label1.caption:='Правільна'
  else begin osh:=osh+1;
  label1.caption:='Памылка'
  end;
  str(i,t); label3.caption:=t; str(osh,t);
label5.caption:=t; rg1.ItemIndex:=-1;
  timer1.Enabled:=true;
end;
procedure TForm1.Timer1Timer(Sender: TObject);
begin
  label1.caption:=''; timer1.Enabled:=false;
  {Вызначэнне нумара правільнага адказу
і яго праверка}
  rg1.ItemIndex:=-1; i:=i+1;
  if i>k then begin
  st1.Caption:=''; st2.caption:='';
  st3.caption:=''; st4.caption:='';
  rg1.Visible:=false;
  label1.Caption:='Вашы вынікі:';
  end
  else begin

```

```

st1.SetFocus; str(i,t); t:='Пытанне '+t;
st1.Caption:=t;
  {выбар нумара выпадковага
непаўтараючага пытання}
  repeat n:=random(k)+1;
  until not (n in z);
  z:=z+[n]; vopros;
  st2.Caption:=''; st3.Caption:='';
  st4.Caption:='';
  st2.Caption:=v; st3.Caption:=v1;
  st4.Caption:=v2; {вывад пытання}
  {вывад варыянтаў адказаў на экран}
  rg1.items.Strings[0]:=o1;
  rg1.items.Strings[1]:=o2;
  rg1.items.Strings[2]:=o3;
end;
end;
procedure TForm1.FormCreate(Sender: TObject);
begin osh:=0; randomize; z:=[];i:=0;
end;
END.

```

Заданне для самастойнай працы

Унясіце змены ў праграму так, каб:

- у канцы работы выстаўлялася адзнака,
- варыянты адказаў змяняліся выпадковым чынам.

Літаратура

- Педагогика / под ред. П. И. Пидкасистого. М., 2001.

SUMMARY

Basic stages of teaching of future teachers of computer science to development of supervising programs with the help of programming languages and an example of laboratory work are suggested.