**Түйіндеме**

*Берілген жұмыста рационал өрнектердің логикалық есептемесін құрастыру және меңгеру кезінде дәлелдеу, алгоритм, есептеушілік және анықталғандық туралы қазіргі заман көзқарасын қалыптастыруына әсер ететін логикалық есептемелердің дедуктивті құралдарының пропедевтика мүмкіндіктері анықталады. Және де осы мүмкіндіктерді оқушылардың логикалық сауаттылығын қалыптастыру процесінде іскі асырудың әдістемелік нұсқаулар беріледі.*

**Resume**

*The present work reveals the possibilities of propaedeutics of deductive means of logical calculus which promote the formation of modern concepts of proving, algorithm, computability and definability on the basis of construction and study of rational expressions logical calculus and gives methodological recommendations for the realization of these possibilities in the process of forming the logical competence of students.*

ӘӨЖ 004.42

# ҚАШЫҚТЫҚТАН ОҚУ КУРСТАРЫН ЖОБАЛАУДЫ КОМПЬЮТЕРЛІК ІСКЕ АСЫРУ

***Б.Ж. НҰРБЕКОВ, М.С. КАЗАНГАПОВА***

**С.Торайғыров атындағы**
**Павлодар мемлекеттік университеті**

Қашықтықтан оқыту курстарын жобалауда қатар оқытылатын курстарды реттеу үшін немесе бір курс ішінде оқытылатын тақырыптарды реттеу үшін топологиялық сұрыптау әдісінің идеяларын пайдаланамыз. Бұл үшін топологиялық сұрыптау алгоритмінің төмендегі модификацияланған түрін пайдаланамыз:

While Граф бос емес do
begin
Ізашары жоқ төбелерді анықтау.
Осы төбелер тобын жақшаға алып шығару.
Графтар деректері және инциденттік доғаларды жою
end

Егер В курсы үшін А курсы бойынша материалды білу керек болса, онда А<В деп, белгілеіміз. Топологиялық сұрыптаудың мәні, бір-де бір курс, осы курсқа тірек болатын курстардан бұрын оқылмайды [1].

Тақырыптарды топологиялық әдіспен сұрыптау

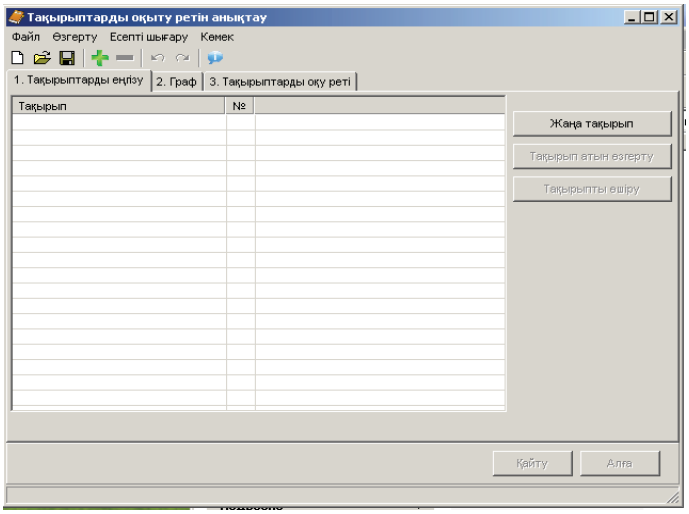
```
function TCourse.GetSortedList: Boolean;
var
a: array of Boolean;
  i, j: Integer;
function HasParent(Index: Integer): Boolean;
var
i: Integer;
begin
Result := False;
for i := 0 to Count - 1 do
if not a[i] then
if Matrix[i][Index] = 1 then begin
Result := True;
Break;
end;
end;
begin
Result := False;
if HasCycles then
Exit;
SetLength(a, Count);
SetLength(SortedList, Count);
for i := 0 to Count - 1 do begin
for j := 0 to Count - 1 do
if not a[j] then
if not HasParent(j) then begin
SortedList[i] := j;
a[j] := True;
Break;
end;
end;

Result := True;

end [2].
```

Ұсынылып тұрған CourseTopological жобасы келесі 14 модульден тұрады:

1. About.ddp
2. About.dfm

3. About.pas
4. CourseTopologicalApp.dpr
5. CourseTopologicalApp.dproj
6. CourseTopologicalApp.dproj.local
7. CourseTopologicalApp.exe
8. CourseTopologicalApp.identcache
9. CourseTopologicalApp.res
10. W2kMain.ddp
11. W2kMain.dfm
12. W2kMain.pas
13. W2kStrs.rc
14. W2kStrs.res

Бағдарламаның негізгі терезесі келесі суретте келтірілген.



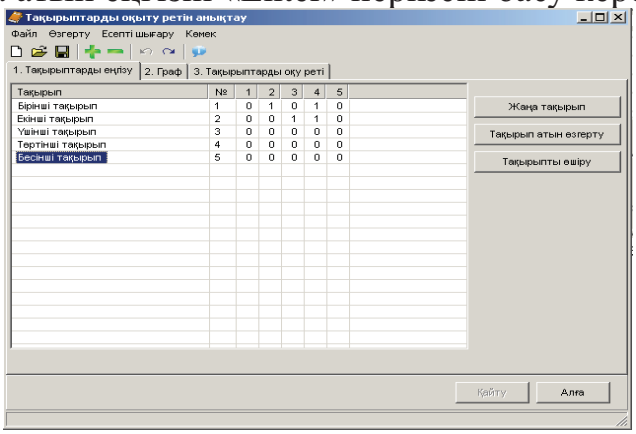

Сурет 1 - Бағдарламаның негізгі терезесі.

Негізгі әрекеттер бас менюге шығарылған:

- жана файл құру
- сақталған файлды ашу
- өзгерістерді файлға сақтау
- бағдарламадан шығу
- жаңа тақырып қосу
- тақырыптың атын өзгерту
- тақырыпты өшіру
- есепті шығарудың алдыңғы және келесі қадамдарына өту
- жоба туралы мәліметті көру.

Бағдарламамен жұмыс жасау үшін CourseTopologicalApp.exe файлды орындау керек.

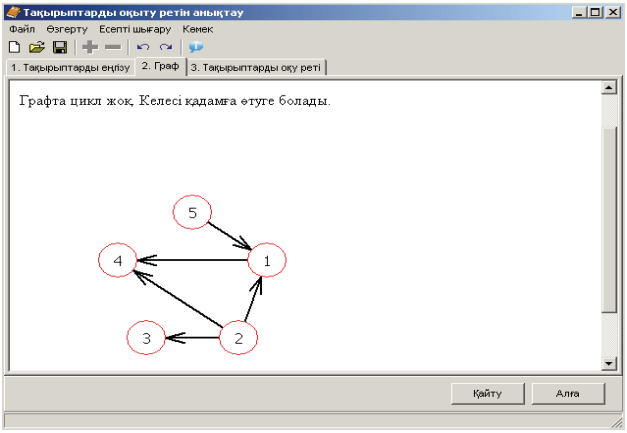
Бағдарламаға алдымен мәліметтерді еңгізу керек: тақырыптардың атауларын және байланыс матрицасын. Жаңа тақырып қосу үшін «Жаңа тақырып» батырмасын басу қажет. Тақырыптың атын өзгерту үшін оны таңдап сосын «Тақырыптың атын өзгерту» батырмасын басу керек, тақырыптың жаңа атын еңгізіп «Enter» пернесін басу керек.




Сурет 2 - тақырыптар атаулары, байланыс матрицасы.

Терезедегі «Алға» батырмасын басқанда терезеде еңгізілген мәлімет бағытталған граф түрінде көрсетіледі. Онымен қоса Тереңге іздеу әдісімен графта циклдар бар немесе жоғы анықталып есептеу нәтижесі экранға жазылады:

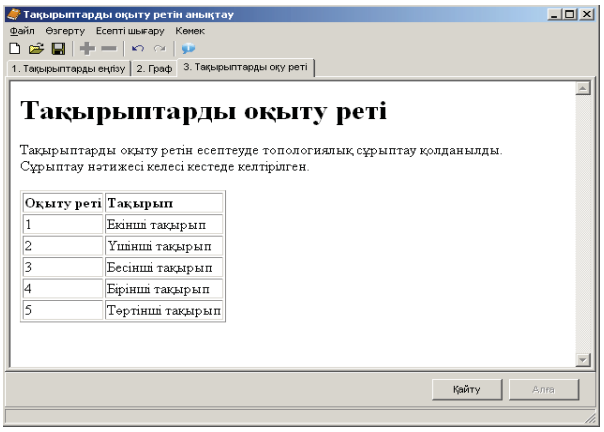
- «Графта цикл жоқ. Келесі қадамға өтуге болады.»
- «Графта циклдар бар. Есептеуді жалғастыру мүмкін емес!» - бұл жағдайда есептің келесі қадамын есептеу мүмкін емес, сондықтан мәліметтерді еңгізу қадамына қайтып, байланыстар матрицасын өзгерту қажет.




Сурет 3 - Граф циклы

Терезедегі «Алға» батырмасын басқанда тақырыптарды оқытудың реті есептеледі. Топологиялық сұрыптау әдісі, еңгізілген тақырыптардың байланыстары туралы мәліметі ескеріп қолданылады. Нәтижесінде

тақырыптарды оқыту реті кесте түрінде экранға беріледі.




Сурет 4 - тақырыптар оқыту реті.

Сонымен, қатар оқылатын курс бөлімдерін ажырата көрсетіп оқу курсының бағдарламасын оқу мазмұнына байланысты оңтайландыруды топологиялық сұрптаумен іске асыруға болады.

## ӘДЕБИЕТТЕР

1. Толкачев Ф.В. Система упражнений по императивному программированию в фундаментальной подготовке будущих учителей информатики [Электронный ресурс]: дис. канд. пед. наук: 13.00.02. – М.: РГБ, (Из фондов Российской Государственной библиотеки), 2003;

2. Вирт Н. Алгоритмы и структуры данных. Библиотека программиста. – СПб.: Невский Диалект, -2001, 235с.

***Резюме***

*В этой статье рассматривается алгоритм топологической сортировки при проектировании дистанционного обучающего курса. В данной работе также различаются отдельные части курса, приспособление программы учебного курса в соответствии с учебным содержанием показано с помощью топологической сортировки.*

***Resume***

*The algorithm of topological sorting has been considered while making remote training course in this article. In the present work separate parts of a course, adaptations, and training course programs have been differed according to the educational maintenance, which is shown by means of topological sorting.*