МАТЕМАТИКА И МАТЕМАТИЧЕСКО ОБРАЗОВАНИЕ, 2003
MATHEMATICS AND EDUCATION IN MATHEMATICS, 2003
Proceedings of the Thirty Second Spring Conference of
the Union of Bulgarian Mathematicians
Sunny Beach, April 5–8, 2003

ЕЛЕМЕНТИ ОТ ТЕОРИЯ НА ГРАФИТЕ В ИЗВЪНКЛАСНАТА РАБОТА В СРЕДНИТЕ УЧИЛИЩА

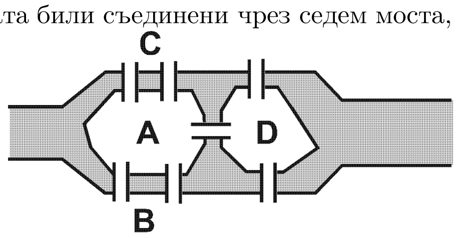
Иван А. Мирчев, Зоранчо А. Алексов

В статията е представен един от разработените от авторите модули за извънкласна работа в часовете по информатика и математика.
Редица реални практически проблеми и тяхното решаване се описват ефективно, нагледно и лесно с езика на теорията на графите – търсене на най-кратки пътища, организиране на потоци в мрежа, покриващи дървета, намиране на Ойлерови и Хамилтонови маршрути и др. Запознаването на учениците от средните училища с такава проблематика е важно и полезно най-малко поради следните три причини:
– интересът към математиката се повишава, тъй като се разглеждат приложни проблеми и се руши представата на учениците, че математиката е суха, сложна и само теоретична наука;
– разглеждането на графови проблеми не изисква допълнителни знания извън традиционната учебна програма;
– алгоритмичният подход при излагане на елементи от теория на графите дава възможност за осъществяване на междупредметни връзки от различни области – математика, информатика, икономика, биология, химия и др.

Ще формулираме няколко класически, емблематични задачи от теория на графите, свързани с търсене на Ойлерови маршрути, които са достъпни за учениците.

Задача 1. Град Кьонигсберг (Калининград) бил разположен на бреговете на река и два нейни острова. Участъците от сушата били съединени чрез седем моста, както е показано на чертеж 1.

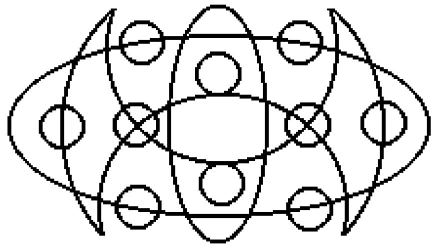
В началото на 18-ти век жителите на града се занимавали с интересна задача – може ли жител на града, тръгвайки от дома си (кой да е от четирите участъка от сушата A, B, C, D), да премине по всеки от седемте моста точно веднъж и да се върне у дома си? Можете ли вие да намерете такъв път?



Черт. 1

Задача 2. Можете ли да начертаете фигурата от чертеж 2 без да повдигате молива от хартията и без да повтаряте линии?

Задача 3. Можете ли с пръст точно по веднъж да преминете по всички ръбове на кутия кибрит, започвайки от произволен връх, вървейки по ръбовете на кутията



Черт. 2

и при това никъде да не повдигате пръст от нея? Ако не можете, колко пъти най-малко трябва да повдигнете пръста?

Такъв тип „несериозни“ на пръв поглед задачи са свързани с решаване на сериозни оптимизациони проблеми в макросистеми - организиране на оптимални маршрути на стоки, хора и информация, вземане на управленски решения, стратегии за разполагане на обекти. Примери в това отношение са дадени в края на статията.

Излагането на този материал предполага познаване само на следните основни понятия от теорията на графите: граф (свързан, ориентиран, неориентиран, теглови, s-граф), пьт (верига, маршрут), цикъл, степен на връх.

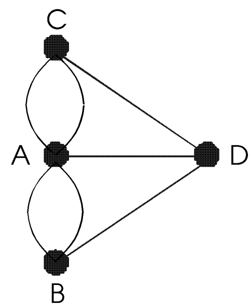
Ойлерови вериги. Да се върнем към задачата за Кьонигсбергските мостове (**Задача 1.**).

Ойлер пръв е поставил въпроса за съществуването на цикъл в неориентиран s-граф G, който преминава по всяко ребро точно веднъж. Всеки такъв цикъл се нарича *(покриващ) ойлеров цикъл.*

Когато съществува ойлеров цикъл, който покрива всички ребра на G, графът G се нарича *ойлеров граф.*

Отворена верига (път, маршрут), която минава точно веднъж по всяко ребро, се нарича *отворена ойлерова верига.*

На езика на графите задачата за мостовете лесно се представя, както е показано на чертеж 3. Ако на всеки участък от сушата A, B, C, D, съпоставим връх на граф, в който дъгите са съществуващите Кьонигсбергски мостове, поставеният въпрос „може ли жител да излезе от дома си и да се върне пак там, като мине по всички мостове точно веднъж (нито повече, нито по-малко)“, е всъщност въпрос съществува ли в графа от чертеж 3 ойлеров цикъл?



Черт. 3

В конкретната ситуация е почти очевидно, че ойлеров цикъл не съществува, тъй като всички върхове на графа от черт. 3 са от нечетна степен - броят на „излизанията“ от всеки начален връх е по-голям от броя на „влизанията”.

В общия случай е вярна следната теорема:

Теорема 1. *Свързаният неориентиран s-граф G съдържа покриващ ойлеров цикъл (е ойлеров) тогава и само тогава, когато всички върхове на графа са от четна степен.*

Доказателство: *Необходимост:* Очевидно, ако съществува покриващ ойлеров цикъл в графа G, то всички върхове на G са от четна степен (броят на „влизанията“ в един връх е равен на броя на „излизанията“).

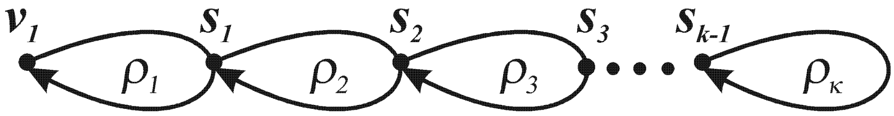
Достатъчност: Ще дадем конструктивно доказателство, което представлява алгоритъм за намиране на ойлеров цикъл в случая, когато всички върхове на свързания граф са от четна степен.

Да си изберем произволен връх ν_i в графа $G := G_i$ и да се „движим“ произволно по различни ребра на графа. Рано или късно ще достигнем отново във върха ν_i. Допускането, че се намираме някъде другаде (в друг връх) и не можем да се върнем във ν_i, противоречи на предположението, че всички върхове на свързания граф са от четна степен. С други думи намерихме цикъл $\rho_i = (\nu_i, \ldots, \nu_i)$, включващ различни ребра на графа.

1 случай. Ако ребрата на ρ_i са всички ребра на G_i, то сме намерили ойлеров цикъл.

2 случай. Цикълът ρ_i не включва всички ребра на графа G_i. „Отстраняваме“ (от разглеждане) ребрата на ρ_i от G_i. Избираме си връх s_i от цикла ρ_i, инцидентен с някое от оставащите (неотстранени) ребра на G_i. Поне един такъв връх s_i (*свързаваща точка*) със сигурност съществува (графът е свързан и всички върхове са от четна степен). Полагаме $\nu_i := s_i$ и повтаряме горе описаната процедура.

По-просто казано – „разбиваме“ графа на k на брой ребрено непресичащи се цикли по следния начин:



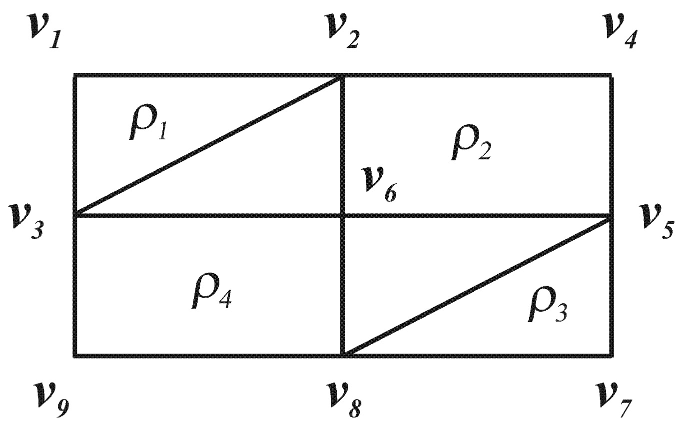
Очевидно ойлеровия цикъл в графа G можем да опишем (начертаем без вдигане на молива и повтаряне на ребра) по следния начин. Тръгваме от ν_1 и се движим в цикъла ρ_1, докато стигнем до свързваща точка s_1 (посоката на движение няма значение); от s_1 се движим в цикъла ρ_2, докато стигнем до свързващата точка s_2 и т.н., докато достигнем свързващата точка s_{k-1} и опишем изцяло цикъла ρ_k (последния цикъл); връщаме се в предпоследния цикъл и се движим по неописания му участък, докато стигнем до свързваща точка s_{k-2} и т.н., докато стигнем до свързваща точка s_1 и опишем неизвървяния участък от цикъла ρ_1, докато стигнем в ν_1.

Съгласно горе казаното, отговорът на въпроса от **Задача 2** е позитивен. Ако всичките линии се разглеждат като ребра, а всичките пресечни или допирни точки са върхове, тогава чертежът от задачата представя граф, в който всичките върхове са от четна степен.

Пример: Има ли ойлеров цикъл в графа от чертеж 4 и ако има, го задайте!

Всички върхове на свързания граф G са от четна степен и съгласно доказаната теорема графът е ойлеров, т.е. в него съществува покриващ ойлеров цикъл.

Разглеждаме $\rho_1 = (\nu_1, \nu_2, \nu_3, \nu_1)$. „Отстраняваме“ ребрата от този цикъл и в оставащия граф разглеждаме цикъла $\rho_2 = (\nu_2, \nu_4, \nu_5, \nu_6, \nu_2)$ – виж чертеж 5. „Отстраняваме“ и тези ребра от графа. Разглеждаме в оставащия граф цикъла $\rho_3 = (\nu_5, \nu_7, \nu_8, \nu_5)$. „Отстраняваме“ и тези ребра от графа. В оставащия граф разглеждаме цикъла $\rho_4 = (\nu_8, \nu_9, \nu_3, \nu_6, \nu_8)$. Отстраняваме и тези ребра от графа – край на

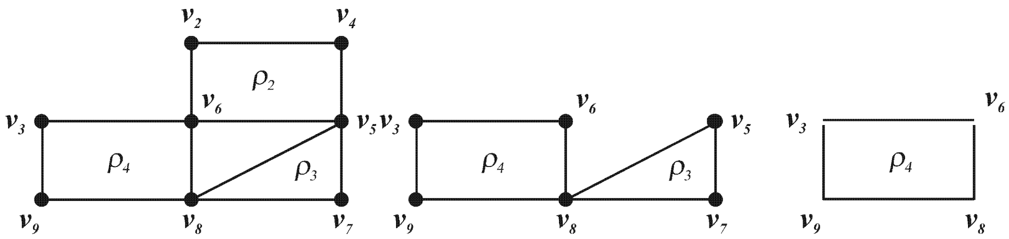


Черт. 4

процедурата за разбиване на графа (ребрата от ρ_4 бяха последните останали ребра на графа).

Един ойлеров цикъл в графа G е $(\nu_1, \nu_2, \nu_4, \nu_5, \nu_7, \nu_8, \nu_9, \nu_3, \nu_6, \nu_8, \nu_5, \nu_6, \nu_2, \nu_3, \nu_1)$.

Намерете и друг ойлеров цикъл - друга последователност на преминаване по ребрата.



Черт. 5

Ясно е, че ако графът не е свързан, ойлеров цикъл, покриващ графа, не съществува, тъй като няма верига, водеща от една негова компонента в друга.

Съществува и следният прост алгоритъм за построяване на покриващ ойлеров цикъл (когато той съществува) в неориентиран граф, който лесно може да се приложи и за ориентирани графи: *„Стартирайте от връх ν и всеки път отстранявайте преминатото ребро. Не преминавайте по ребро, ако отстраняването на това ребро води до двукомпонентно разбиване на графа (без да се броят изолираните върхове)“.*

Очевидно е, че ако съществува отворена ойлерова верига, покриваща графа, тя свързва два върха на графа, които са от нечетна степен.

Следният резултат характеризира графите, които могат да бъдат покрити с определен брой отворени ойлерови вериги:

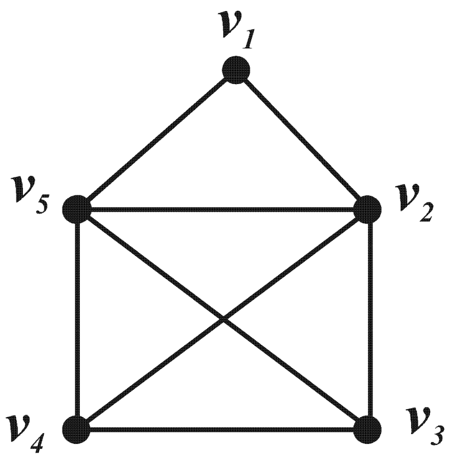
Теорема 2. *Нека G е свързан граф с $2.s > 0$ нечетни върха. Съществуват s независими (нямащи общи ребра) ойлерови вериги, които покриват G. По-малко от s вериги с това свойство няма. Във всяка система от s независими ойлерови вериги, покриващи G, веригите са отворени, две по две нямат общи крайни точки и всяка верига съединява върхове от нечетна степен.*

Няма да доказваме тази теорема.

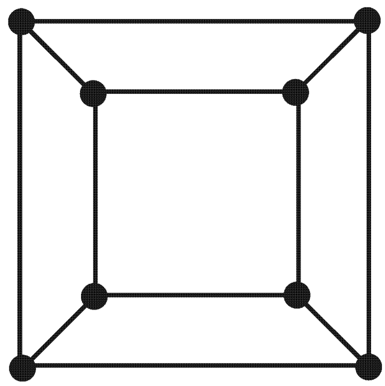
Следствие. *При $s = 1$, т.е. ако в графа G съществуват точно два върха от нечетна степен, графът G се покрива от една отворена ойлерова верига.*

Пример, илюстриращ следствието, е и графът на чертеж 6.

Задача 3, формулирана в началото на статията, може да се сведе до граф, показан на чертеж 7. Всеки връх на графа е от нечетна степен. Тъй като графът има 8=2·4 върхове от нечетна степен, графът се покрива с четири независими ойлерови вериги. За да се мине по всички ръбове на кутията точно веднъж, трябва пръстът да се повдигне от кутията три пъти.



Черт. 6



Черт. 7

Друга задача, която може да се представи с граф и да се сведе до търсене на ойлеров цикъл (верига), е следната:

Задача: Може ли шахматен кон, движейки се по шахматна дъска, да направи всеки допустим ход на коня точно по веднъж?

Упътване: Разгледайте 64-те шахматни полета като върхове на граф, в който възможните ходове на коня между две полета са ребра. Опитайте!

Отговор: Не може. В графа има осем нечетни върха.

Покриването на един граф с ойлерови цикли или вериги е една важна за практиката задача. Ще формулираме няколко оптимизационни проблеми, тясно свързани с намирането на ойлерови вериги и цикли.

(A) Задача за китайския пощальон – The Chinese Postman Problem (CPP). Проблемът е формулиран и изследван от китайския математик Kwan Mei-Ko, откъдето идва и наименованието му. *Пощальон трябва да разнесе писма, те-*

леграми, пенсии и т.н. до всички адреси от региона, който обслужва. Какъв маршрут трябва да избере пощальонът, за да мине по всяка улица ***поне веднъж,*** *така че да измине минимално разстояние и да се върне отново в пощенския офис?*

Очевидно районът, обслужван от пощальона, може да се разглежда като граф с върхове кръстовищата на улиците и ребра самите улици. Освен това, на всяко ребро в този граф може да се съпостави тегло (дължината на улицата в метри или времето за преминаване в минути). Когато в цитирания граф има покриващ ойлеров цикъл, очевидно оптималният маршрут за движение на пощальона е точно този цикъл. Ясно е, че когато има повече ойлерови цикли, има повече оптимални маршрути, всичките с дължина, равна на сумата от теглата на ребрата.

Ясно е, че в общия случай (графът не винаги е ойлеров) на пощальона може да се наложи да повтори движението по някои от улиците. Няма да разглеждаме този проблем, но може да се даде идея на учениците за това как **CPP** задачата може да се сведе до задача за намиране на ойлеров цикъл.

Полезно би било да се разгледа и дискутира този проблем в случай на ориентиран и смесен граф и най-малко да се обърне внимание върху еквивалентността на проблема при неориентиран граф и ориентиран, симетричен и четен граф (за всеки връх броят на изходните и входните дъги е еднакъв).

Проблемът **CPP** е интересен класически оптимизационен проблем, тъй като може да се отнася не само до пощальони, а до движение на превозни средства или хора, доставящи стоки, извършващи инспекции на обекти, разглеждане на музеи или панаири, събиране на отпадъци и т.н. Пример за такъв проблем е проблемът за снегорина.

(Б) Задача за снегорина. *Служба за поддръжка на пътищата при зимни условия е дислоцирана на определено място в даден град. Всяка сутрин кола на службата излиза от депото, за да почиства улиците от сняг и да разпръсква солена луга против заледяването. Известен е капацитетът на колата – например 5 тона солена луга. Да се намери маршрут за движение на снегорина при няколкократно презареждане с луга в депото, така че да бъдат обезопасени всички улици с минимални разходи.* С други думи отново трябва да се реши проблемът за покриване на граф, като условието е по всяка от улиците да се преминава поне по веднъж.

Задачата **(Б)** може да се обобщи, ако се включат и времеви ограничения – например цялата работа да се завърши за n часа. В този случай въпросът е какъв е минималният брой снегорини, достатъчен за това.

Много подобен проблем на задачата за снегорина е проблемът за тестване на софтуер. Софтуерът може да се представи с граф, чийто върхове са състоянията (states) на програмата, а дъги са възможните преминавания от едно в друго състояние (action). Целта е да се тестват всичките възможни преминавания от едно в друго състояние по най-ефикасен начин.

Добре е учениците сами да формулират (моделират) реални проблеми от практиката, свеждащи се до търсене на оптимални ойлерови маршрути.

Задачите от тип **CPP** са интересни, лесно се обясняват, но решаването им не винаги е лесно. Това могат да бъдат много сериозни практически проблеми, особено когато са включени допълнителни условия. Съществува богата литература за този тип проблеми. В Интернет лесно се намират много уеб-страници, свързани с тази проблематика. На учениците може да им се възложи сами да намерят примери и решения в Интернет.

ЛИТЕРАТУРА

[1] Мирчев, И., Графи. Оптимизационни алгоритми в мрежи, Унив. изд. „Неофит Рилски", Благоевград, 2001.
[2] Diestel, R., Graph Theory Electronic Edition 2000, Springer Verlag, New York, 1997, 2000.
[3] Robinson, H. Graph Theory Techniques in Model-Based Testing, Semantic Platforms Test Group, Microsoft Corporation, Presented at the 1999 International Conference on Testing Computer Software.
[4] Network Flow Problems, *International Journal of Industrial Engineering*, 1999.
[5] Thimbleby, H. The directed Chinese Postman Problem, School of Computing Science, Middlesex University, London.
[6] http://www.c3.lanl.gov/mega-math/workbk/graph/grbkgd.html#games
http://www2.toki.or.id/book/AlgDesignManual/BOOK/BOOK4/NODE165.HTM
http://www.math.princeton.edu/tsp/index.html
http://www-leibniz.imag.fr/GRAPH/english/definitions.html#flow
http://www.lehigh.edu/~inmath/postman1.html

Иван Асенов Мирчев
ЮЗУ Неофит Рилски
ул. Иван Михайлов 66
2700 Благоевград, България
e-mail: mirchev@aix.swu.bg

Зоранчо Ангелов Алексов
ЮЗУ Неофит Рилски
ул. Иван Михайлов 66
2700 Благоевград, България
e-mail:zoran@focus.net.mk

ELEMENTS OF GRAPH THEORY FOR ADDITIONAL WORK OUT OF REGULAR CLASSES IN SECONDARY SCHOOLS

Ivan A. Mirchev, Zorancho A. Alexov

In this paper we present one of our units for additional work out of the regular classes for the subjects informatics and mathematics.
The solution of many real, practical problems could be effectively, clearly and easily described by using the language of the Graph Theory. Such problems are: searching for the shortest paths, organizing the network flows, spanning trees, finding Eulerian and Hamiltonian routes etc. Introducing the Graph Theory to students in secondary schools is important and useful at least for the next three reasons:
– The interest for mathematics is rising because these problems are interesting, taken from real life. This means breaking the fame that mathematics is boring, complicated and only theoretical science;
– Solving graph problems does not require additional knowledge, outside the traditional curriculum;
– The algorithmic approach to analyzing the elements of Graph Theory gives the opportunity for understanding the interconnections between different disciplines - mathematics, informatics, economy, biology, chemistry and so.