Львівський державний університет безпеки життєдіяльності ДСНС України

Кафедра прикладної математики і механіки

ТЕМА 3. МЕРЕЖІ І ПОТОКИ.
ДЛЯ ПРОВЕДЕННЯ ЛЕКЦІЙНОГО ЗАНЯТТЯ
З КУРСАНТАМИ ТА СТУДЕНТАМИ __2__ КУРСУ
З ДИСЦИПЛІНИ __СИСТЕМНИЙ АНАЛІЗ ТА ТЕОРІЯ ПРИЙНЯТТЯ РІШЕНЬ

Мета лекції:

Навчальна: вивчити задачу про максимальний потік, формалізацію задачі про максимальний потік як задачі лінійного програмування, задачу про потік найменшої вартості, формалізацію задачі про потік найменшої вартості як задачі лінійного програмування.

Виховна: виховання свідомого ставлення до вивчення предмету, самостійності, відповідальності та організованості при підготовці до занять.

Розвиткова: розвиток логічного та абстрактного мислення, розвиток просторової уяви.

План

Література

1. *Аршинова О.І., Шевченко А.В.* Системний аналіз. Навч. посібник. – К.: НАУ, 2008. – 128 с.
2. *Роїк О.М., Шиян А.А., Нікіфорова Л.О.* Системний аналіз. Навч. посібник. – Вінниця: ВНТУ, 2015. – 83 с.
3. *Кузик А.Д.* Основи системного аналізу. – Львів: ЛДУ БЖД, 2005. – 100 с.
4. *Кунда Н.Т.* Дослідження операцій у транспортних системах. – К.: Видавничий Дім “Слово”, 2008. – 400 с.
5. *Карагодова О.О., Кігель В.Р., Рожок В.Д.* Дослідження операцій. – К.: Центр учбової літератури, 2007.– 256 с.
6. *Таха Х.* Введение в исследование операций. –6-е изд.: Пер. с англ. –М.: Изд. дом "Вильямс", 2001. – 912 с.

Час проведення: 2 години.
Місце проведення: лекційний зал.
Забезпечення заняття: мультимедіа.

1. Задача про максимальний потік.

Мережа (сітьовий графік) може зображати систему, яка транспортує деякий продукт з одного пункту в інший. Цим продуктом можуть бути люди, електроенергія, природній газ тощо. Прикладом може бути мережа трубопроводів для транспортування нафти від свердловин через насосні станції до нафтопереробних заводів.

Використовуючи таке представлення розглянемо мережу як орієнтований граф, де кожній дузі (i, j) відповідає додатне дійсне число c_{ij}, яке називають ***пропускною здатністю*** у напрямку $i \to j$. Такий граф не може мати петель, оскільки розглядаються задачі для транспортування продукту тільки між різними вершинами. Орієнтований граф повинен бути зв'язним. Вхід будемо також називати ***джерелом***, а вихід – ***стоком***. Джерело не має вхідних дуг, а сток – вихідних. Таку мережу аналітично можна задавати за допомогою **матриці пропускних здатностей**. Графічно на схемі мережі ближче до вершини *i* будемо позначати c_{ij}, ближче до вершини *j* будемо позначати c_{ji} (рис.3.1).

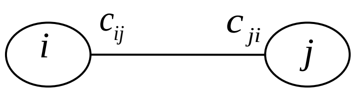


Рис. 3.1.

Потрібно знайти максимальну величину потоку (кількості машин, рідини, сировини тощо), який може увійти до мережі та вийти з неї за заданий період часу. Для розв'язання задачі про максимальний потік використовують метод Форда - Фалкерсона. В ***Maple*** задача про максимальний потік розв'язується за допомогою функції

MaxFlow (G, s, t)

де ***G*** – зважений граф;
s – вершина, що є джерелом графа;
t – вершина, що є стоком графа.

Приклад 3.1. Мережа автодоріг, що проходять через Львівську область, може забезпечити пропускні здатності (тис. автомашин за годину), які вказані на рис.3.2. Потрібно визначити максимальний потік у заданій мережі.

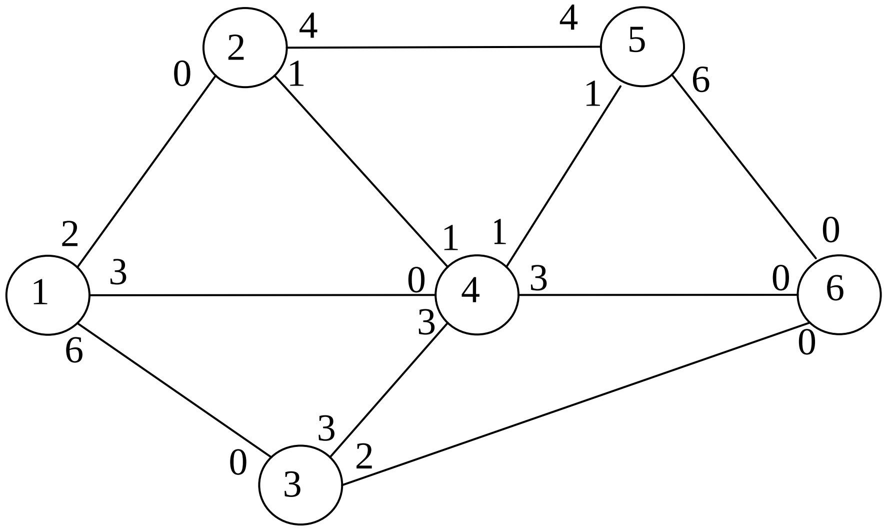


Рис. 3.2.

Розв'язання.

Розв'яжемо задачу в ***Maple***.

> *restart* :

> *with*(*GraphTheory*) :

Задаємо матрицю пропускних здатностей

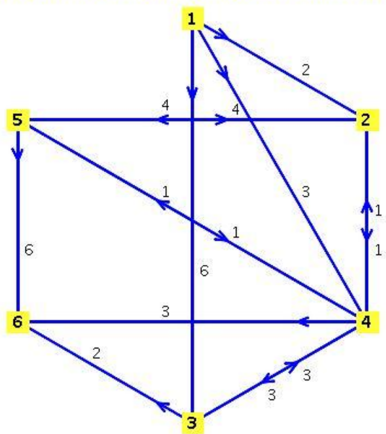
> $A := \begin{bmatrix} 0 & 2 & 6 & 3 & 0 & 0 \\ 0 & 0 & 0 & 1 & 4 & 0 \\ 0 & 0 & 0 & 3 & 0 & 2 \\ 0 & 1 & 3 & 0 & 1 & 3 \\ 0 & 4 & 0 & 1 & 0 & 6 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$;

$$A := \begin{bmatrix} 0 & 2 & 6 & 3 & 0 & 0 \\ 0 & 0 & 0 & 1 & 4 & 0 \\ 0 & 0 & 0 & 3 & 0 & 2 \\ 0 & 1 & 3 & 0 & 1 & 3 \\ 0 & 4 & 0 & 1 & 0 & 6 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{1}$$

Задаємо орієнтований зважений граф, що відповідає матриці *A* та будуємо його

> *N* := *Digraph*(*A*, *weighted*); *DrawGraph*(*N*);

N := Graph 1: a directed weighted graph with 6 vertices and 14 arc(s)



Шукаємо максимальний потік від вершини 1 до вершини 6

```
> MaxFlow(N, 1, 6);
```

$$9, \begin{bmatrix} 0&2&5&2&0&0 \\ 0&0&0&0&3&0 \\ 0&0&0&3&0&2 \\ 0&1&0&0&1&3 \\ 0&0&0&0&0&4 \\ 0&0&0&0&0&0 \end{bmatrix} \qquad (2)$$

Отримали величину максимального потоку $F_{\max} = 9$ та матрицю величин потоків по кожній дузі.

2. Формалізація задачі про максимальний потік як задачі лінійного програмування.

Припустимо, що потрібно знайти максимальний потік між джерелом *s* та стоком *t*. Позначимо:

x_{ij} – величина потоку, що проходить по дузі (i, j);

c_{ij} – пропускна здатність цієї дуги.

Для кожної проміжної вершини записується обмеження, що задає баланс потоку, який проходить через дану вершину:

загальний вхідний потік = загальний вихідний потік.

Для кожної дуги записується обмеження, що потік не перевищує пропускної здатності дуги та є невід’ємним:

0 ≤ потік по дузі ≤ пропускна здатність дуги.

Функцією мети, яку потрібно максимізувати, є величина потоку, що виходить з джерела *s* або входить у сток *t*.

Приклад 3.2. Запишемо задачу з попереднього прикладу, як задачу лінійного програмування. Обмеження для кожної проміжної вершини та функцію мети наведено у табл. 3.1. Структура коефіцієнтів, що формують обмеження, має таку особливість: в стовпці, що відповідає змінній x_{ij}, завжди в рядку, що відповідає вершині *i* стоїть – 1, а в рядку, що відповідає вершині *j* стоїть +1. Решта коефіцієнтів дорівнюють нулю. Така структура коефіцієнтів типова для мережевих моделей.

Таблиця 3.1.

Дуги / *Вершини*	x_{12}	x_{13}	x_{14}	x_{24}	x_{25}	x_{34}	x_{36}	x_{42}	x_{43}	x_{45}	x_{46}	x_{52}	x_{54}	x_{56}	*Вільні члени*
	y_1	y_2	y_3	y_4	y_5	y_6	y_7	y_8	y_9	y_{10}	y_{11}	y_{12}	y_{13}	y_{14}	
2	1	0	0	-1	-1	0	0	1	0	0	0	1	0	0	= 0
3	0	1	0	0	0	-1	-1	0	1	0	0	0	0	0	= 0
4	0	0	1	1	0	1	0	-1	-1	-1	-1	0	1	0	= 0
5	0	0	0	0	1	0	0	0	0	1	0	-1	-1	-1	= 0
Пропускна здатність d_k	2	6	3	1	4	3	2	1	3	1	3	4	1	6	
Коефіцієнти функції мети F_1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	*Max*
Коефіцієнти функції мети F_2	0	0	0	0	0	0	1	0	0	0	1	0	0	1	*Max*
Оптимальний план	2	4	3	0	3	2	2	1	0	1	3	0	0	4	

Перепозначимо невідомі величини потоків y_k, пропускні здатності дуг d_k, $k=\overline{1,14}$. Отримаємо наступну задачу лінійного програмування:

$$F_1 = y_1 + y_2 + y_3 \to \max \qquad\qquad F_2 = y_7 + y_{11} + y_{14} \to \max$$
$$A \cdot \overline{y} = \overline{0}, \qquad \text{або} \qquad A \cdot \overline{y} = \overline{0},$$
$$0 \le y_k \le d_k, \qquad\qquad 0 \le y_k \le d_k,$$

де A – матриця коефіцієнтів рівнянь - обмежень (2-5-ий рядок, 2-15-ий стовпець табл.3.1.), $\overline{y} = (y_1, y_2, \ldots, y_{14})$, $\overline{0} = (0,0,0,0)$, $k=\overline{1,14}$. Розв'язавши задачу засобами ***Maple*** отримуємо оптимальні значення величин потоків $\overline{y} = (2,4,3,0,3,2,2,1,0,1,3,0,0,4)$.

> restart;

Підключаємо пакет ***simplex***

> with(simplex) :

Задаємо функцію мети L та систему обмежень -- нерівностей ***ineq***

> $L := y_1 + y_2 + y_3$; $ineq := \{y_1 - y_4 - y_5 + y_8 + y_{12} = 0, y_2 - y_6 - y_7 + y_9 = 0, y_3 + y_4 + y_6 - y_8 - y_9 - y_{10} - y_{11} + y_{13} = 0, y_5 + y_{10} - y_{12} - y_{13} - y_{14} = 0, y_1 \geq 0, y_1 \leq 2, y_2 \geq 0, y_2 \leq 6, y_3 \geq 0, y_3 \leq 3, y_4 \geq 0, y_4 \leq 1, y_5 \geq 0, y_5 \leq 4, y_6 \geq 0, y_6 \leq 3, y_7 \geq 0, y_7 \leq 2, y_8 \geq 0, y_8 \leq 1, y_9 \geq 0, y_9 \leq 3, y_{10} \geq 0, y_{10} \leq 1, y_{11} \geq 0, y_{11} \leq 3, y_{12} \geq 0, y_{12} \leq 4, y_{13} \geq 0, y_{13} \leq 1, y_{14} \geq 0, y_{14} \leq 6\}$;

$$L := y_1 + y_2 + y_3$$

$$ineq := \{y_2 - y_6 - y_7 + y_9 = 0, y_1 - y_4 - y_5 + y_8 + y_{12} = 0, y_5 + y_{10} - y_{12} - y_{13} - y_{14} = 0, y_3 + y_4 + y_6 - y_8 - y_9 - y_{10} - y_{11} + y_{13} = 0, 0 \leq y_1, 0 \leq y_2, 0 \leq y_3, 0 \leq y_4, 0 \leq y_5, 0 \leq y_6, 0 \leq y_7, 0 \leq y_8, 0 \leq y_9, 0 \leq y_{10}, 0 \leq y_{11}, 0 \leq y_{12}, 0 \leq y_{13}, 0 \leq y_{14}, y_1 \leq 2, y_2 \leq 6, y_3 \leq 3, y_4 \leq 1, y_5 \leq 4, y_6 \leq 3, y_7 \leq 2, y_8 \leq 1, y_9 \leq 3, y_{10} \leq 1, y_{11} \leq 3, y_{12} \leq 4, y_{13} \leq 1, y_{14} \leq 6\}$$

Знаходимо максимум функції L при заданій системі обмежень -- нерівностей ***ineq***

> maximize(L, ineq)

$$\{y_1 = 2, y_2 = 4, y_3 = 3, y_4 = 0, y_5 = 3, y_6 = 2, y_7 = 2, y_8 = 1, y_9 = 0, y_{10} = 1, y_{11} = 3, y_{12} = 0, y_{13} = 0, y_{14} = 4\}$$

Знаходимо максимальне значення функції L

> assign(maximize(L, ineq)); L;

$$9$$

3. Задача про потік найменшої вартості.

Задача пошуку потоку найменшої вартості в мережі з обмеженою пропускною здатністю узагальнює задачу визначення максимального потоку за такими параметрами:

1) кожній дузі поставлена у відповідність невід'ємна вартість проходження одиниці потоку по даній дузі;

2) дуги можуть мати нижнє додатне обмеження пропускної здатності;

3) будь-яка вершина мережі може бути як джерелом, так і стоком.

В **задачі про потік найменшої вартості** потрібно знайти такий розподіл потоків по дугам, вартість якого є мінімальною. При цьому повинні враховуватись обмеження на пропускні здатності дуг і на величини попиту і пропозиції окремих (чи всіх) вершин.

Розглянемо мережу *G* з обмеженою пропускною здатністю. Позначимо:

x_{ij} – величина потоку, що проходить по дузі (i, j);

l_{ij} – нижня межа пропускної здатності дуги (i, j);

u_{ij} – верхня межа пропускної здатності дуги (i, j);

c_{ij} – вартість проходження одиниці потоку по дузі (i, j);

f_j – величина потоку, що проходить через вершину *j*.

На рис.3.3. наведено приклад позначення параметрів вершин та дуг на схемі мережі. Додатне значення $[f_j]$ вказує значення попиту, від'ємне – значення пропозиції.

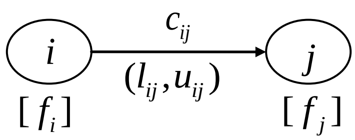


Рис. 3.3.

4. Формалізація задачі про потік найменшої вартості як задачі лінійного програмування.

Задачу про потік найменшої вартості можна подати у вигляді задачі лінійного програмування. Для кожної вершини записується обмеження, що задає баланс потоку, який проходить через дану вершину:

загальний вхідний потік – загальний вихідний потік =
= потік, що проходить через вершину.

Для кожної дуги записується обмеження, що потік є не меншим, ніж нижня межа пропускної здатності та не перевищує верхньої межі пропускної здатності:

нижня межа ≤ потік по дузі ≤ верхня межа.

Використовуючи прийняті позначення, задачу лінійного програмування для мережі з обмеженою пропускною здатністю можна записати так:

$$F = \sum_{(i,j)\in E} c_{ij} x_{ij} \to \min$$

$$\sum_{\substack{k \\ (j,k)\in E}} x_{jk} - \sum_{\substack{i \\ (i,j)\in E}} x_{ij} = f_j,$$

$$l_{ij} \le x_{ij} \le u_{ij},$$

де E – множина дуг.

Приклад 3.3. Для перевезення зерна з трьох зерносховищ до трьох агропромислових підприємств використовується залізничний та автомобільний транспорт. Можливі маршрути перевезень зображено на рис. 3.4. Пропозиція зерносховищ (пункти 1, 2, 3) становить відповідно 100, 200 та 50 тон, а попит агропромислових підприємств (пункти 4, 5, 6) – 150, 80 та 120 тон. На маршрутах, де використовується автомобільний транспорт, є нижнє та верхнє обмеження пропускної здатності. Пропускна здатність залізничного транспорту практично необмежена. Вартість транспортування однієї тони зерна на кожному маршруті (в сотнях гривень) наведено біля відповідної дуги (рис. 3.4.). Потрібно визначити план перевезення найменшої вартості.

Розв'язання.

Нехай x_{ij} – величина потоку, що проходить по дузі (i, j). Обмеження для кожної вершини та функцію мети наведено в табл. 3.2.

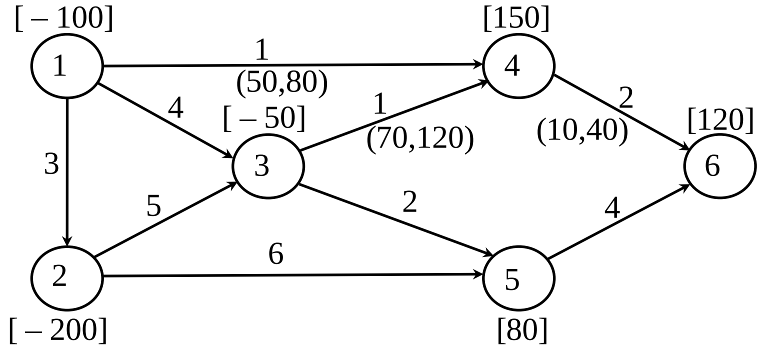


Рис. 3.4.

Таблиця 3.2.

Дуги / *Вершини*	x_{12}	x_{13}	x_{14}	x_{23}	x_{25}	x_{34}	x_{35}	x_{46}	x_{56}	*Вільні члени*
	y_1	y_2	y_3	y_4	y_5	y_6	y_7	y_8	y_9	
1	-1	-1	-1	0	0	0	0	0	0	*= -100*
2	1	0	0	-1	-1	0	0	0	0	*= -200*
3	0	1	0	1	0	-1	-1	0	0	*= -50*
4	0	0	1	0	0	1	0	-1	0	*= 150*
5	0	0	0	0	1	0	1	0	-1	*= 80*
6	0	0	0	0	0	0	0	1	1	*= 120*
Нижня межа, l_k	0	0	50	0	0	70	0	10	0	
Верхня межа, u_k	∞	∞	80	∞	∞	120	∞	40	∞	
Коефіцієнти c_k *функції мети* F	3	4	1	5	6	1	2	2	4	min

Перенумеруємо невідомі величини потоків – y_k, коефіцієнти функції мети – c_k, нижні межі пропускної здатності дуг – l_k, верхні межі пропускної здатності дуг – u_k, $k=\overline{1,9}$. Отримаємо наступну задачу лінійного програмування:

$$F=\sum_{k=1}^{9} c_k y_k \to \min$$

$$A\cdot\overline{y}=\overline{f},$$

$$l_k \le y_k \le u_k,\ k=\overline{1,9},$$

де A – матриця коефіцієнтів рівнянь-обмежень (2-7-ий рядок, 2-10-ий стовпець табл.3.2.), $\overline{y}=(y_1,y_2,\ldots,y_9)$, $\overline{f}=(-100,-200,-50,150,80,120)$ – вектор вільних членів. Розв'язавши задачу засобами ***Maple*** отримуємо оптимальні значення величин потоків $\overline{y}=(0,20,80,40,160,110,0,40,80)$. Загальна вартість перевезень становить 183 000 грн.

Розв'яжемо задачу в ***Maple***.

Утворюємо Excel-файл 97-2003 з даними таблиці 3.2. та називаємо його potik.xls (рис.3.5.), а лист перейменовуємо на 1. У комірку K12 вводимо формулу

=СУММПРОИЗВ(B11:J11;B12:J12)*100, в якій буде обчислено найменшу вартість перевезень. Файл має бути розміщений у тій самій папці, що й файл програми ***Maple***. Значення $u_k = \infty$ замінено на достатньо велике скінчене значення, в нашому випадку 1000.

	A	B	C	D	E	F	G	H	I	J	K
1	Дуги	x12	x13	x14	x23	x25	x34	x35	x46	x56	Вільні
2	Вершини	y1	y2	y3	y4	y5	y6	y7	y8	y9	члени
3	1	-1	-1	-1	0	0	0	0	0	0	-100
4	2	1	0	0	-1	-1	0	0	0	0	-200
5	3	0	1	0	1	0	-1	-1	0	0	-50
6	4	0	0	1	0	0	1	0	-1	0	150
7	5	0	0	0	0	1	0	1	0	-1	80
8	6	0	0	0	0	0	0	0	1	1	120
9	Нижня межа	0	0	50	0	0	70	0	10	0	
10	Верхня межа	1000	1000	80	1000	1000	120	1000	40	1000	
11	Коефіцієнти	3	4	1	5	6	1	2	2	4	min
12	Оптимальний план										0

Рис. 3.5.

Після цього файл potik.xls закриваємо.

Відкриваємо ***Maple***. Підключаємо необхідні пакети. Пакет ***ExcelTools*** допоможе нам отримати вхідні дані з таблиці ***Excel*** (рис.3.5.).

```
> restart :
> with(Optimization) : with(ExcelTools) :
```

Імпортуємо дані (табл. 3.2.), що задані у файлі potik.xls (рис.3.5.).

Коефіцієнти рівнянь-обмежень:

```
> A := Import("potik.xls", "1", "B3:J8") : A := convert(A, Matrix);
```

$$A := \begin{bmatrix} -1.0 & -1.0 & -1.0 & 0. & 0. & 0. & 0. & 0. & 0. \\ 1.0 & 0. & 0. & -1.0 & -1.0 & 0. & 0. & 0. & 0. \\ 0. & 1.0 & 0. & 1.0 & 0. & -1.0 & -1.0 & 0. & 0. \\ 0. & 0. & 1.0 & 0. & 0. & 1.0 & 0. & -1.0 & 0. \\ 0. & 0. & 0. & 0. & 1.0 & 0. & 1.0 & 0. & -1.0 \\ 0. & 0. & 0. & 0. & 0. & 0. & 0. & 1.0 & 1.0 \end{bmatrix} \quad (1)$$

Коефіцієнти функції мети:

> $c := Import("potik.xls", "1", "B11:J11") : c := convert(c, Vector);$

$$c := \begin{bmatrix} 3.0 \\ 4.0 \\ 1.0 \\ 5.0 \\ 6.0 \\ 1.0 \\ 2.0 \\ 2.0 \\ 4.0 \end{bmatrix} \qquad (2)$$

Вектор вільних членів:

> $f := Import("potik.xls", "1", "K3:K8") : f := convert(f, Vector);$

$$f := \begin{bmatrix} -100.0 \\ -200.0 \\ -50.0 \\ 150.0 \\ 80.0 \\ 120.0 \end{bmatrix} \qquad (3)$$

Нижні межі обмежень:

> $l := Import("potik.xls", "1", "B9:J9") : l := convert(l, Vector);$

$$l := \begin{bmatrix} 0. \\ 0. \\ 50.0 \\ 0. \\ 0. \\ 70.0 \\ 0. \\ 10.0 \\ 0. \end{bmatrix} \qquad (4)$$

Верхні межі обмежень:

> $u := Import("potik.xls", "1", "B10:J10") : u := convert(u, Vector);$

$$u := \begin{bmatrix} 1000.0 \\ 1000.0 \\ 80.0 \\ 1000.0 \\ 1000.0 \\ 120.0 \\ 1000.0 \\ 40.0 \\ 1000.0 \end{bmatrix} \qquad (5)$$

Розв'язуємо задачу лінійного програмування у матричній формі. Отримуємо оптимальне значення функції мети та оптимальний розв'язок:

$$> Rozv := LPSolve(c, [NoUserValue, NoUserValue, A, f], [l, u]);$$

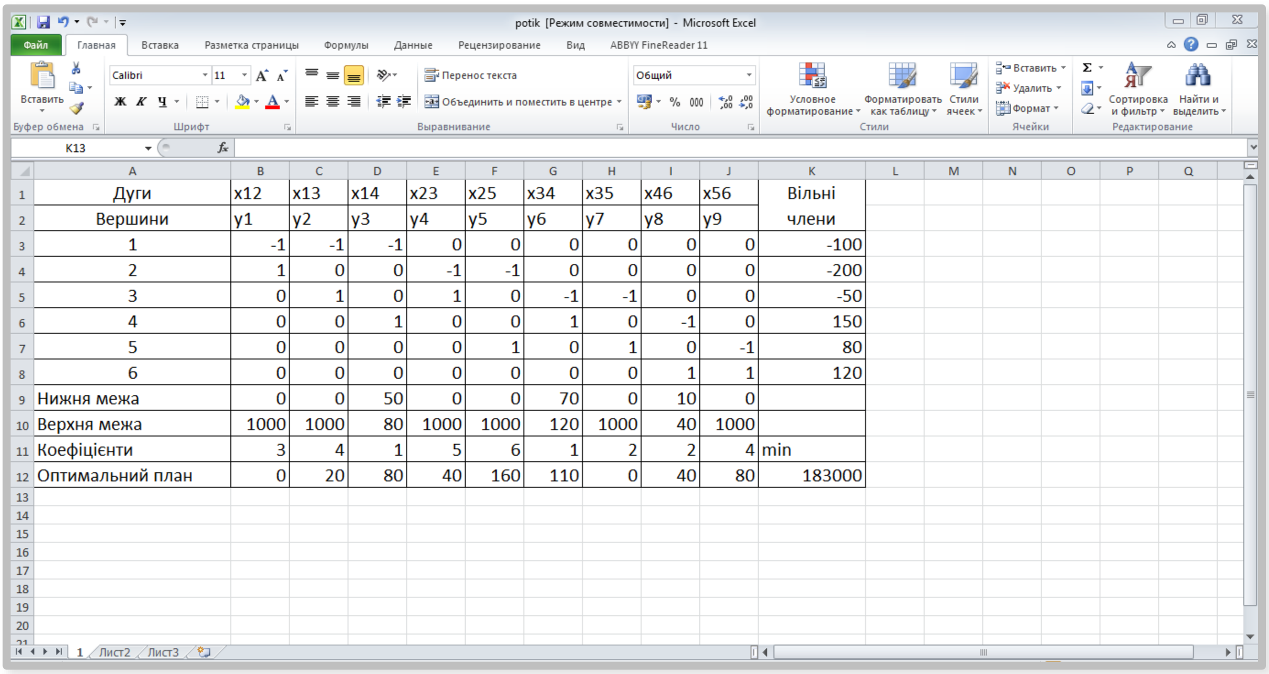
$$Rozv := \left[1830.00000000000, \begin{bmatrix} 0. \\ 19.9999999999997016 \\ 80. \\ 40.0000000000002842 \\ 159.999999999999970 \\ 110.000000000000014 \\ 0. \\ 40. \\ 79.9999999999999574 \end{bmatrix} \right] \qquad (6)$$

Експортуємо отриманий розв'язок у файл potik.xls, попередньо транспонуємо його (рис.3.6.).

$$> x := op(2, Rozv);$$

$$x := \begin{bmatrix} 0. \\ 19.9999999999997016 \\ 80. \\ 40.0000000000002842 \\ 159.999999999999970 \\ 110.000000000000014 \\ 0. \\ 40. \\ 79.9999999999999574 \end{bmatrix} \qquad (7)$$

```
> x := convert(x, Vector[row]) :
> Export(x, "potik.xls", "1", "B12:J12") :
```



Дуги	x12	x13	x14	x23	x25	x34	x35	x46	x56	Вільні
Вершини	y1	y2	y3	y4	y5	y6	y7	y8	y9	члени
1	-1	-1	-1	0	0	0	0	0	0	-100
2	1	0	0	-1	-1	0	0	0	0	-200
3	0	1	0	1	0	-1	-1	0	0	-50
4	0	0	1	0	0	1	0	-1	0	150
5	0	0	0	0	1	0	1	0	-1	80
6	0	0	0	0	0	0	0	1	1	120
Нижня межа	0	0	50	0	0	70	0	10	0	
Верхня межа	1000	1000	80	1000	1000	120	1000	40	1000	
Коефіцієнти	3	4	1	5	6	1	2	2	4	min
Оптимальний план	0	20	80	40	160	110	0	40	80	183000

Рис. 3.6.

Контрольні запитання.

1. Постановка задачі про максимальний потік.
2. Задача про максимальний потік як задача лінійного програмування.
3. Методи розв'язання задачі про максимальний потік.
4. Постановка узагальненої задачі пошуку потоку найменшої вартості.

Завдання на самопідготовку.

1. Віртуальний університет ЛДУ БЖД [Електронний ресурс]. — Режим доступу: http://virt.ldubgd.edu.ua/
2. Системний аналіз та теорія прийняття рішень [Електронний ресурс] / Чмир Оксана Юріївна. — Режим доступу: http://virt.ldubgd.edu.ua/course/view.php?id=1765

Доцент
кафедри прикладної математики і механіки Оксана Чмир

Лекція обговорена на засіданні
кафедри прикладної математики і механіки

Протокол № від “ ” серпня 20 р.