НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
ІМЕНІ ІГОРЯ СІКОРСЬКОГО»

Кафедра прикладної математики

МЕТОДИЧНІ РЕКОМЕНДАЦІЇ

до виконання лабораторних робіт
з кредитного модуля «АЛГОРИТМИ І СИСТЕМИ КОМП'ЮТЕРНОЇ МАТЕМАТИКИ 1.МАТЕМАТИЧНІ АЛГОРИТМИ»
для студентів
спеціальності 113 «Прикладна математика»

Затверджено на засіданні
кафедри прикладної математики

Протокол № 14

від «01» 07. 2021р.

Завідувач кафедри
прикладної математики

___________ О.Р.Чертов

«____»________ 2021 р.

Київ – 2021

Лабораторна робота №1

Розв’язання систем лінійних рівнянь

Мета роботи – вивчити правила використання програмних засобів для факторизації матриць і розв’язання систем лінійних алгебраїчних рівнянь, розв’язувати задану систему рівнянь із використанням , провести порівняльний аналіз вивчених чисельних методів розв’язання СЛР.

Методичні вказівки

Вибір у кожному конкретному випадку методу розв’язання систем лінійних алгебраїчних рівнянь визначається багатьма факторами: особливостями матриці коефіцієнтів системи, порядком системи, швидкодією, обсягом пам’яті ЕОМ і т.п.
Одним із найбільш універсальних і ефективних методів розв’язання систем лінійних алгебраїчних рівнянь є метод Гауса. Застосування його доцільне для лінійних систем загального виду з щільно заповненою матрицею коефіцієнтів. Реалізація методу вимагає біля n^2 слів оперативної пам’яті ЕОМ, що обмежує порядок розв’язуваної з його допомогою системи *(* $n < 10^2$ *)*. Число арифметичних операцій, виконуваних при розв’язанні методом Гауса порядку *n*, у загальному випадку складає приблизно $2n^3/3$.
Практична реалізація методів розв’язання систем лінійних алгебраїчних рівнянь часто виконується із застосуванням факторизації /чи трикутної декомпозиції/ матриць. У таких алгоритмах матриця коефіцієнтів перетворюється в добуток трикутних матриць. У СКМ включений ряд ефективних програм, які реалізують алгоритми факторизації. Так, у *LH-методі* матриця коефіцієнтів A представляється у вигляді добутку $A=LH$, де L нижня трикутна матриця, H верхня трикутна матриця з одиницями на головній діагоналі. Задача розв’язання системи $AX=y$ заміняється на задачу розв’язання системи $LHX=b$, де $HX=y$ і $LY=b$. Останні системи рівнянь мають трикутні матриці коефіцієнтів. При рішенні рівнянь високого порядку *(* $n \approx 10^2 - 10^5$ *)* із розрідженою матрицею коефіцієнтів часто виявляється ефективним застосування ітераційних методів. У розріджених матрицях більшість коефіцієнтів дорівнюють нулю. У пам'яті ЕОМ зберігаються тільки ненульові елементи таких матриць. Ітераційні методи використовують особливості розріджених матриць, тому вони вимагають виконання меншої кількості арифметичних операцій. Помилки округлення в ітераційних методах даються взнаки істотно менше, ніж у прямих. Багато ітераційних методів є самовиправляючимися, тобто окрема помилка, допущена при обчисленнях, не відбивається на кінцевому результаті. Важливою перевагою ітераційних методів є зручність їхнього програмування для ЕОМ, тому що вони вимагають одноманітних повторюваних операцій.

Порядок виконання роботи

1. Ознайомитися з теоретичним матеріалом і методичними вказівками до роботи.
2. Одержати варіант завдання /задану систему рівнянь, використовувані підпрограми, необхідну точність розв'язку/.
3. Скласти блок-схему алгоритму розв’язання поставленої задачі.
4. Написати, налагодити і виконати програму розв’язання систем лінійних алгебраїчних рівнянь із використанням підпрограм СКМ.
5. Розв’язати задану систему лінійних алгебраїчних рівнянь і оцінити точність отриманого розв'язку по координатах.

$\delta = max \angle\ x^i - x_i^* \angle;\ i= 1,...,N;$

де x^i координати точного розв’язання;

x_i^* координати чисельного розв’язання;

N число невідомих.

Зміст звіту

1. Вихідні дані по роботі.
2. Блок-схема алгоритму.
3. Опис програми.
4. Лістінги програми і протокол розв’язання задачі на ЕОМ.
5. Оцінка точності отриманого розв’язання.
6. Висновки по роботі.

Контрольні питання

1. У якому випадку система лінійних алгебраїчних рівнянь має єдиний розв’язок?
2. Як використовується факторизація матриць для розв’язання систем лінійних рівнянь?
3. Охарактеризуйте прямі й ітераційні методи розв’язання систем лінійних рівнянь.
4. Опишіть компактну схему методу Гауса.
5. Опишіть алгоритм ітераційного методу Зейделя.
6. Дайте визначення збіжності ітераційного процесу.
7. Які методи називають самовиправляючимися?

Варіанти завдань

Варіант	Матриця коефіцієнтів системи А				Вектор вільних членів В
1	4	3	2	1	3
	3	6	4	2	6
	2	4	6	3	4
	1	2	3	4	7
2	0,31	0,14	0,30	0,27	1,02
	0,26	0,32	0,18	0,24	1,00
	0,61	0,22	0,20	0,31	1,34
	0,40	0,34	0,36	0,17	1,27
3	1,2357	2,1742	-5,4834		-2,0735
	6,0696	-6,2163	-4,6921		-4,8388
4	2,0	1,0	-0,1	0,1	2,7

	0,4	0,5	4,0	-8,5	21,9
	0,3	-0,1	1,0	5,2	-3,9
	1,0	0,2	2,5	–1,0	9,9
5	3	1	-1	2	6
	-5	1	3	-4	-12
	2	0	1	-1	1
	1	-5	3	-3	3
6	2	–1	4	1	11
	-1	1	2	3	14
	4	2	3	-1	4
	-3	1	3	4	16
	1	3	-1	4	18
7	3,1	1,5	1,8		10,83
	1,5	2,5	0,5		9,20
	1,0	0,5	4,2		17,10
8	2,12	0,42	1,34		11,172
	0,42	3,95	1,87		0,115
	1,34	1,87	2,98		9,009
	0,88	0,43	0,46		11,172
9	2,09	1,2	2,1		21,70
	1,2	21,2	1,5		27,46
	2,1	1,5	19,8		28,76
	0,9	2,5	1,3		49,72
10	6,1	2,2	1,2		16,55
	2,2	5,5	–1,5		10,55
	1,2	–1,5	7,2		16,80
11	3,82	1,02	0,75		15,655
	1,05	4,53	0,98		22,705
	0,73	0,85	4,71		23,480
	0,88	0,81	1,28		16,110
12	0,15	2,11	30,75		-26,38
	0,64	1,21	2,05		1,01
	3,21	1,53	1,04		5,23
13	1,15	0,42	100,71		-198,70
	1,19	0,55	0,32		2,29
	1,00	0,35	3,00		-0,65
14	1,02	-0,25	-0,30		0,515
	-0,41	1,13	–0,15		1,555
	-0,25	–0,14	1,21		2,780
15	10,9	1,2	2,1		-0,7
	1,2	11,2	1,5		5,3
	2,1	1,5	9,8		10,3
	0,9	2,5	12,1		24,6

Лабораторна робота №2

Чисельне диференціювання й інтегрування

Мета роботи– ознайомитися з програмними засобами чисельного диференціювання й інтегрування функцій; практичне розв'язання задач з використанням СКМ, порівняльний аналіз методів чисельного диференціювання й інтегрування.

Методичні вказівки

Для одержання формул чисельного диференціювання часто застосовують інтерполяційний многочлен Лагранжа. Істотні зручності для обчислень забезпечують таблиці значень функцій у точках, розташованих з постійним кроком $h = x_i - x_{i-1}$, $i = i, \ldots, n$. З урахуванням цього диференціювання многочлена Лагранжа дає наступна формула перших похідних у рівно розташованих точках при n = 2 /три точки, многочлен другого степеня/:

$$\begin{cases} \frac{1}{2h}(-3y_i + 4y_{i+1} - y_{i+2}) & \text{якщо } i = 0; \\ \frac{1}{2h}(y_{i+1} - y_{i-1}) & \text{якщо } i = 1, \ldots, n-1; \\ \frac{1}{2h}(y_{n-2} - 4y_{n+1} + 3y_n) & \text{якщо } i = n. \end{cases}$$

Похибка обчислення похідної по цих формулах складається з похибок усікання похибок округлення. Перші називаються заміною функції $y=f(x)$ інтерполяційним многочленом $P(x)$, другі– неточним значенням y_i. Зі зменшенням кроку h похибка усікання зменшується, а похибка округлення збільшується. Якщо припустити, що кожна з ординат $f(x_i)$ може мати помилку Δ, то для внутрішніх точок $x_1, \ldots, x_{n-1}$ оптимальне значення $h_0 \approx 1{,}8\sqrt[3]{\Delta / M^3}$,

де $\mu_3 = max\, f^{(3)}(\xi)$, $\xi \in [x_0, x_n)$.

Величина загальної похибки

$$E_i = \begin{cases} 3{,}3\sqrt[3]{\Delta^2 M^3}, & \text{якщо } i = 0; \\ 1{,}5\sqrt[3]{\Delta^2 M^3}, & \text{якщо } i = 1, \ldots, n-1. \end{cases}$$

Аналогічним шляхом одержують розрахункові формули похідних більш високого порядку. Однак точність чисельного диференціювання істотно зменшується з ростом порядку похідної. При завданні підінтегральної функції графіком чи таблицею експериментально отриманих значень для чисельного інтегрування найчастіше застосовують програми, що реалізують формулу трапецій, формулу Сімпсона чи квадратну формулу Гауса. Оцінка похибки цих формул по виразах для залишкових членів виявляється малоефективною через труднощі оцінки похідних високого порядку. Тому похибку наближеного значення інтеграла оцінюють за правилом Рунге. Для функцій високої гладкості, тобто тих, що мають неперервні похідні досить високого порядку, при однаковому числі вузлів формула Гауса дає значно більш точні результати, ніж формула Сімпсона, а остання– більш точні результати, ніж по формулі трапецій.

Формула Гауса забезпечує високу точність при невеликому числі вузлів, тому її особливо вигідно застосовувати при інтегруванні складних функцій, на обчислення значень яких витрачається багато часу. Однак, для функцій малої гладкості, що мають лише першу і другу похідну, а також для функцій з розривами похідних прості формули інтегрування можуть давати ту ж точність, що і формули Гауса. Крім того, формули Гауса з великою кількістю вузлів досить складні, і застосовувати їх для таких функцій невигідно.

При інтегрування функцій, заданих таблицею, формулу Гауса можна використовувати тільки при відповідному розташуванні кінців інтервалу розбиття, що можна забезпечити в рідких випадках. Формула Сімпсона в достатньо точна для помірного числа вузлів, зручно програмується для ЕОМ, і тому набула широкого застосування в практичних розрахунках.

Порядок виконання роботи

1. Ознайомитися з теоретичним матеріалом і методичними вказівками до роботи.
2. Одержати варіант завдання /підінтегральну функцію, інтервал, крок, первісну функцію, метод розв’язання задачі/.
3. Скласти таблиці значень підінтегральної та первісної функцій.
4. Розробити алгоритми чисельного диференціювання й інтегрування таблично заданих функцій.
5. Скласти програму обчислення значень інтеграла і похідної, одержати чисельний розв'язок задачі з використанням підпрограм СКМ.
6. Обчислити абсолютне значення похибок чисельного диференціювання й інтегрування, використовуючи для одержання точних значень підінтегральну та первісну функції.

Зміст звіту

1. Постановка задачі, вихідні дані для роботи, таблиці значень функцій.
2. Блок-схеми алгоритмів.
3. Опис програми.
4. Лістінги програми і протокол розв’язання задачі на ЕОМ.
5. Графіки заданих функцій і оцінки точності отриманих розв’язків.
6. Висновки по роботі.

Контрольні питання

1. Як залежить похибка усікання і похибка округлення від розміру кроку при чисельному диференціюванні?

2. Оцініть похибку заданої формули чисельного диференціювання за допомогою дослідження на апроксимацію.
3. Проведіть порівняння точності формул трапецій і Сімпсона на підставі аналізу залишкових членів формул.
4. У яких випадках доцільно використовувати квадратні формули Гауса?
5. Які формули чисельного інтегрування зручніше програмувати для ЕОМ?

Варіанти завдань

Варіант	Підінтегральна функція	Проміжок інтегру-вання	Кількість частин розбиття	Крок h	Первісна функція
1	$\frac{x}{(x+3)^2}$	[0;2]	40	0,05	$\frac{3}{x+3}+\ln(x+3)$
2	$\frac{\sqrt{4-x^2}}{x}$	[0,2;1]	80	0,01	$\sqrt{4-x^2}-2\ln\frac{2+\sqrt{4-x^2}}{x}$
3	$x\sin 2x$	[0;π/4]	20	π/80	$\frac{\sin 2x}{4}-\frac{x\cos 2x}{2}$
4	2^{3x}	[0;1]	50	0,02	$\frac{2^{3x}}{3\ln 2}$
5	$\frac{\ln^2 x}{x}$	[1;5]	50	0,08	$\frac{\ln^3 x}{3}$
6	$e^{2x}\sin x$	[0;π/2]	30	π/60	$\frac{e^{2x}}{5}(2\sin x-\cos x)$
7	$\frac{x^2}{2x+3}$	[1;3]	100	0,02	$\frac{1}{8}\left(2x^2-6x+9\ln(2x+3)\right)$
8	$x^2\sqrt{x+2}$	[1;4]	50	0,06	$\frac{2(15x^2-24x+32)}{105}\sqrt{(x+2)^3}$
9	$\frac{x}{\sin^2 3x}$	[0,2;1]	25	0,04	$-\frac{x}{3}ctg3x+\frac{1}{9}\ln\sin 3x$
10	$xe^{0,8x}$	[2;3]	40	0,025	$\frac{e^{0,8x}}{0,64}(0,8x-1)$
11	$\frac{1}{x\sqrt{x^2+0,25}}$	[1;2]	50	0,02	$-2\ln\left(\frac{0,5+\sqrt{x^2+0,25}}{x}\right)$
12	$\frac{x^2}{(2x+0,3)^2}$	[1;2]	80	0,0125	$0,25x-0,075\ln(2x+0,3)-\frac{0,01125}{2x+0,3}$
13	$\frac{x}{0,5x+0,1}$	[3;5]	50	0,04	$\frac{8(0,5x+0,2)}{3}\sqrt{0,5x+0,1}$
14	$x^2\sin x$	[0;1]	40	0,025	$2x\sin x-(x^2-2)\cos x$
15	$x2^{3x}$	[1;4]	50	0,05	$\frac{x2^{3x}}{3\ln 2}-\frac{2^{3x}}{9(\ln 2)^2}$

Лабораторна робота №3
Розв’язання звичайних диференціальних рівнянь

Мета роботи: вивчення правил використання програмних засобів чисельного розв’язання звичайних диференціальних рівнянь, практичне розв’язання рівнянь на ЕОМ з використанням СКМ, порівняльний аналіз методів інтегрування ОДУ.

Методичні вказівки

Для багатьох практичних додатків застосовуються програми, розроблені на основі методів Рунге-Кутта. Часто використовуються методи аналізу і корекції.

Метод Рунге-Кутта має досить високу точність і легко програмується. Оскільки для обчислення y_{i+1} необхідно знати тільки одне значення y_i, за допомогою цього методу можна починати розв’язання рівняння. Величину кроку можна змінити на будь-якому етапі розрахунків. Одним з недоліків методу є необхідність багаторазового обчислення значення функції на кожному кроці, що призводить до витрати великої кількості машинного часу.

Можна рекомендувати наступні правила автоматичного обрання кроку.

У вузлі x_0 узяти $h=h_0$, де h_0– заданий початковий крок, знайти наближені значення розв'язку $\bar{y}$ і $\bar{\bar{y}}$, обчислені методом Рунге-Кутта в точці $x_0 + h$ із кроком відповідно h і h_0. за абсолютну похибку наближеного значення, обчисленого методом Рунге-Кутта *i-го* порядку, приймається

$$\Delta = \left| \frac{\bar{y} - \bar{\bar{y}}}{2^{\tau} - 1} \right| .$$

Якщо $\Delta \geq \varepsilon$ /ε–задана величина/, то крок h зменшується в два рази, і обчислення повторюються, виходячи з вузла x_0..

Розв'язок в наступному вузлі x_2, виходячи з вузла x_1, одержують аналогічним шляхом.

Порядок виконання роботи

1. Одержати варіант завдання: диференціальне рівняння, метод інтегрування, використовувану підпрограму СКМ.
2. Скласти блок-схему алгоритму розв’язання диференціального рівняння.
3. Скласти програму розв’язання диференціального рівняння, одержати розв'язок за допомогою розробленої програми і за допомогою підпрограми СКМ.
4. Результати обчислень оформити у вигляді графіків.

5. Визначити близькість отриманого заданим методом розв'язку до точного значення. Як значення використовувати розв'язок, отриманий за допомогою СКМ.

Зміст звіту

1. Постановка задачі.
2. Блок-схема алгоритму.
3. Протоколи виконання програм.
4. Графіки розв’язань.
5. Оцінка точності отриманих результатів.
6. Висновки по роботі.

Контрольні питання

1. Що значить вирішити диференціальні рівняння чисельним методом?
2. Охарактеризуйте метод Рунге-Кутта четвертого порядку.
3. Дайте порівняльну характеристику методів Рунге-Кутта і методів прогнозу-корекції.
4. Опишіть процедуру автоматичного обрання кроку в методі Рунге-Кутта.
5. Як оцінюють точність отриманого розв'язку в методах прогнозу і корекції?

Варіанти завдань

Метод Рунге-Кутта

1. метод 4-го порядку

$$y_{k+1} = y_k + (k_1 + 2k_2 + 2k_3 + k_4)/6,$$

де $k_1 = hf(x_k, y_k)$;

$$k_2 = hf(x_k + h/2, y_k + k_1/2);$$

$$k_3 = hf(x_k + h/2, y_k + k_2/2);$$

$$k_4 = hf(x_k + h, y_k + k_3).$$

2. метод 4-го порядку

$$y_{k+1} = y_k + (k_1 + 4k_3 + k_4)/6,$$

де $k_1 = hf(x_k, y_k)$;

$$k_2 = hf(x_k + h/4, y_k + k_1/4);$$

$$k_3 = hf(x_k + h/2, y_k + k_2/2);$$

$$k_4 = hf(x_k + 2h/3, y_k + k_1 - 2k_2 + 2k_3).$$

3. метод 3-го порядку

$$y_{k+1} = y_k + k_1(4 + 3k_3/4),$$

де $k_1 = hf(x_k, y_k)$;

$$k_2 = hf(x_k + h/3, y_k + k_1/3);$$

$$k_3 = hf(x_k + 2h/3, y_k + 2k_2/3).$$

4. метод 2-го порядку

$$y_{k+1} = y_k + (k_1 + k_2)/2,$$

де $k_1 = hf(x_k, y_k)$;

$k_2 = hf\left(x_k + h, y_k + k_1\right).$

5. метод 2-го порядку

$$y_{k+1} = y_k + \left(k_1 + k_2\right)/2,$$

де $k_1 = hf\left(x_k, y_k\right);$

$$k_2 = hf\left(x_k + h/2, y_k + k_1/2\right).$$

Варі-ант	Диференційне рівняння	Початкова умова	Проміжок інтегрування	Крок
1	$y'' - 2y' + y = 0$	$y(2) = 1; y'(2) = -2$	[2;4]	0,2
2	$y'' - 3y' + 2y - 2x + 3 = 0$	$y(0) = 1; y'(0) = 2$	[0;2]	0,2
3	$y'' + y = 4e^x$	$y(0) = 4; y'(2) = -3$	[0;1]	0,1
4	$x^2 y'' + xy' = 0$	$y(1) = 5; y'(1) = -1$	[1;1,5]	0,05
5	$y'' - 2y' = 2e^e$	$y(1) = -1; y'(1) = 0$	[1;2]	0,1
6	$y'' + 4y = \cos 3x$	$y(0) = 0{,}8; y'(0) = 2$	[0;1]	0,1
7	$y'' + 2y' + 2y = xe^x$	$y(0) = 0; y'(0) = 0$	[0;1,5]	0,1
8	$\left(1 + x^2\right)y'' + \left(y'\right)^2 + 1 = 0$	$y(0) = 1; y'(0) = 1$	[0;0,5]	0,05
9	$y'' + 4y' + 4y = 0$	$y(0) = 1; y'(0) = -1$	[0;1]	0,1
10	$y'' - 3y' = e^{5x}$	$y(0) = 2{,}2; y'(0) = 0{,}8$	[0;0,2]	0,02
11	$x^2 y'' - 2y = 0$	$y(1) = 0{,}83$	[1;2]	0,1
12	$y'' - 5y' + 6y = e^x$	$y(0) = 0; y'(0) = 0$	[0;0,2]	0,02
13	$y'' + y = 1 + e^x$	$y(0) = 2{,}5; y'(0) = 1{,}5$	[0;1]	0,1
14	$x^2 y'' + 2{,}5y'x - y = 0$	$y(1) = 2; y'(1) = 3{,}5$	[1;2]	0,1
15	$y'' + y = x^2 - x + 2$	$y(0) = 1; y'(0) = 0$	[0;1]	0,1

Лабораторна робота №4
Чисельні методи оптимізації

Мета роботи   вивчити методи нульового, першого й другого порядку, практичну мінімізацію функцій багатьох змінних з використанням СКМ.

Методичні вказівки

Детерміновані алгоритми безумовної оптимізації поділяють на класи в залежності від інформації, що використовується. Якщо на кожній ітерації використовуються лише значення мінімізованих функцій, то метод називається методом нульового порядку. Якщо, крім цього, необхідне обчислення похідних мінімізованої функції, то мають місце методи першого порядку, при необхідності додаткового обчислення похідних другого порядку   методи другого порядку.

Загальна характеристика методів нульового порядку

В цих методах для визначення напрямку спуску не потрібно обчислювати похідні цільової функції. Напрямок мінімізації у даному випадку повністю визначається послідовними обчисленнями значень функції. Необхідно зазначити, що при розв'язанні задач безумовної оптимізації методи першого та другого порядків мають, як правило, більш високу швидкість збіжності, ніж методи нульового порядку. Проте на практиці обчислення перших і других похідних функції великої кількості змінних вельми трудомістке. В ряді випадків вони не можуть бути отримані у вигляді аналітичних функцій. Крім цього, на практиці зустрічаються задачі, розв'язання яких можливе лише за допомогою методів нульового порядку, наприклад задачі мінімізації функції з розривними першими похідними, критерій оптимальності може бути заданим у неявному вигляді, а системою рівнянь, у цьому випадку аналітичне або чисельне визначення похідних стає дуже складним, а іноді неможливим. Для розв'язання таких практичних задач оптимізації можуть бути успішно застосовані методи першого порядку.

1. Метод прямого пошуку /метод Хука-Дживса/

Даний метод має наступний зміст. Задаються деякою початковою точкою x_0. змінюючи компоненти вектора x_0, досліджують окіл даної точки і визначають напрямок, у якому відбувається зменшення мінімізованої функції $f(x)$. у цьому напрямку проводять спуск до тих пір, доки значення функції зменшується. Якщо після деякої кількості рухів в обраному напрямку не вдається знайти чергову точку, в якій значення функції зменшується, то зменшують величину кроку спуску. Якщо послідовні зменшення кроку не призводять до зменшення функції, від обраного напрямку спуску відмовляються і проводять нове дослідження околу і т.д.

Перевага методу прямого пошуку   простота його застосування на ЕОМ. Він не потребує знання цільової функції в явному вигляді, а також легко враховує обмеження на окремі змінні, складні обмеження на область пошуку.
Недолік методу в тому, що у випадку сильно витягнутих, вигнутих ліній або ліній, що мають гострі кути, рівня цільової функції він може виявитися нездатним забезпечити просування до точки мінімуму.

2. Метод деформованого многогранника /метод Нелдера-Міда/

Метод складається з того, що для мінімізації функції і змінних *f(x)* в *n*- вимірному просторі будується многогранник, що містить *n+1* вершину. Кожна вершина відповідає деякому вектору *x*. Обчислюються значення цільової функції *f(x)* у кожній вершині многогранника, визначається максимальне з цих значень і відповідна вершина *xh*. Через цю вершину і центр тяжіння решти вершин проводиться проектуюча пряма, на якій знаходиться точка x_q з меншим значенням цільової функції, ніж у вершині *xh*. Після цього вершина *xh* виключається із розглядання. Із вершин, що залишилися, і точки x_q будується многогранник, з яким повторюється описана процедура. В процесі виконання даних операцій многогранник змінює свої розміри і форму, адаптується до топографії цільової функції. В результаті многогранник витягується вздовж довгих похилих поверхонь, змінює напрямок у вигнутих западинах, стискається в околі мінімуму, що визначає ефективність методу.

3. Метод координат, що обертаються /метод Розенброка/

Суть методу в обертанні системи координат у відповідності зі зміною швидкості спадання функції. Нові напрямки координатних осей визначаються так, щоб одна з них відповідала напрямку найбільш швидкого спадання цільової функції, а інші відшукуються з умови ортогональності. У загальному випадку даний метод ефективний при мінімізації “овражних” функцій, так як результуючий напрямок пошуку прямує вздовж вісі “оврагу”.
На відміну від інших методів нульового порядку алгоритм Розенброка орієнтовано на відшукання оптимальної точки у кожному напрямку, на не просто на фіксований зсув по всьому напрямку. Розмір кроку в процесі пошуку неперервно змінюється в залежності від рельєфу поверхні рівня. Сполучення обертання координат з регулюванням кроку робить метод Розенброка ефективний при розв’язання складних задач оптимізації.

4. Метод паралельних дотичних /метод Пауелла/

Цей метод використовує властивість квадратичної функції, зміст якої у тому, що будь-яка пряма, котра проходить через точку мінімуму функції, перетинає під рівними кутами дотичні до поверхні рівного рівня функції в точках перетину.
Суть методу така. Обирається деяка початкова точка x_0 і виконується одновимірний пошук вздовж довільного напрямку, що приводять в точку x_1. після цього обирається точка x_2, що не лежить на прямій x_0-x_1, й проводиться одновимірний пошук вздовж прямої, паралельної x_0-x_1. отримана в результаті точка x_3 разом з точкою x_1 визначає напрямок x_1-x_3 одновимірного пошуку. У випадку квадратичної функції й змінних оптимальне значення знаходить за *n* ітерацій. Пошук мінімуму при цьому проводиться у взаємно спряжених напрямах.

Загальна характеристика методів першого порядку

Методи безумовної оптимізації першого порядку /синонім назви   градієнтні методи/ основані на наступних властивостях вектора-градієнта: градієнт спрямован в сторону найшвидшого зростання функції в даній точці. Антиградієнт спрямован в сторону найшвидшого спадання функції. В точці максимуму градієнт функції дорівнює нулю. Усі градієнтні методи реалізують послідовність переходів від точки до точки у напрямку антиградієнта по деякому критерію досяжності оптимального значення і відрізняються один від одного тільки шляхами вибору розміру кроку переходу.
При методі з постійним кроком для всіх ітерацій обирається постійний достатньо малий розмір кроку. Однак це може призвести до необхідності проводити неприпустимо велику кількість ітерацій для досягання мінімуму.
Більш економічним по кількості ітерацій й надійності градієнтні методи зі змінним кроком, коли в залежності від результатів обчислень розмір кроку деяким шляхом змінюється. На практиці зазвичай застосовуються методи найшвидшого спуску й спряжених градієнтів.

Метод найшвидшого спуску

Розмір кроку a_k на кожній ітерації обирається з умови мінімуму функції *f(x)* у напрямку спуску, тобто рух уздовж антиградієнта відбувається до тих пір, доки значення функції *f(x)* спадає. На кожній ітерації розв’язується задача одновимірної мінімізації по $a_k : \varphi(a_k) = f\left(x_k - a_k f'(x_k)\right)$.
Для гладких опуклих функцій градієнтні методи збігаються до мінімуму з високою швидкістю. Для функцій, що мають погано обумовлені матриці других похідних /“овражні функції”/, швидкість збіжності суттєво зменшується, оскільки напрямок антиградієнта цих функцій суттєво відхиляється від напрямку точки мінімуму.
Швидкість збіжності градієнтних методів істотно залежить від точності обчислення градієнта. Втрата точності, а це зазвичай відбувається в околі точок мінімуму або в “овражній” ситуації, може взагалі порушити збіжність процесу градієнтного спуску. У наслідок перерахованих причин градієнтні методи використовують у комбінації з іншими, більш ефективними методами на початковій стадії розв’язання задачі. У цьому випадку точка x_0 знаходиться далеко від точки мінімуму, і кроки у напрямку антиградієнта дозволяють досягнути значного спадання функції.

Метод спряжених градієнтів

Розглянуті раніше градієнтні методи знаходять точку мінімуму функції в загальному випадку за нескінчену кількість ітерацій. Метод спряжених градієнтів формує напрямок пошуку, що більш відповідає геометрії мінімізованої функції. Це суттєво збільшує швидкість їх збіжності і дозволяє, наприклад, мінімізувати квадратичну функцію

$$f(x) = (x, H_x) + (b, x) + a$$

з симетричною додатньо визначеною матрицею *H* за скінчену кількість кроків *n*, рівне кількості змінних функції. Крім того, будь-яка гладка функція в околі точки мінімуму добре апроксимується квадратичній, тому методи спряжених градієнтів вдало застосовують для мінімізації й не квадратичних функцій. В такому випадку вони перестають бути скінченими і стають ітеративними.
Однією з найбільш істотних проблем у методах спряжених градієнтів є проблема ефективного побудови напрямків. Метод Глетчера-Рівса вирішує цю проблему шляхом перетворення на кожному кроці антиградієнта $f'(x_k)$ у напрямку P_{K_1}. H спряжені з раніше знайденими напрямками $P_0, P_1, ..., P_{K-1}$. Зауваження. Два *n*-вимірних вектора *x* і *y* називають *H*- спряженими

по відношенню до матриці *H* /або *H*- спряженими/, якщо скалярний добуток $(x, Hy) = 0$. *H*- симетрична додатньо визначена матриця розміром $n \cdot n$.
Методи спряжених градієнтів найбільш ефективні при розв'язанні задач мінімізації. Однак необхідно зазначити, що вони чутливі до помилок, що виникають в процесі розрахунків. При великій кількості змінних похибка може настільки зрости, процес доведеться повторювати навіть для квадратичної функції, тобто процес для неї не завжди завершується за *n* кроків.

Загальна характеристика методів другого порядку

Необхідною умовою екстремуму функції багатьох змінних *f(x)* в точці x^* є рівність нулю її градієнта в цій точці: $f'(x) = 0$.
Розклад $f'(x)$ в околі точки $x_к$ в ряд Тейлора з точністю до членів першого порядку дозволяє переписати попередні рівняння у вигляді

$$f'(x) = f'(x_k) + f''(x_k)(x - x_k) = 0.$$

Тут $f''(x_k) = H(x_k)$ матриця других похідних /матриця Гессе/ мінімізованої функції. Отже, ітераційний процес для побудови послідовних наближень до розв'язання задачі мінімізації функції *f(x)* описується виразом

$$x_{k+1} = x_k - H^{-1}(x_k) f'(x_k),$$

де $H^{-1}(x_k)$ обернена матриця до матриці Гессе, а $-H^{-1}(x_k) f'(x_k) = P_k$ напрям спуску.
Отриманий метод мінімізації називають методом Ньютона. Очевидно, що у даному методі розмір кроку вздовж напрямку P_k береться рівною одиниці. Послідовність точок $\{x_k\}$, що отримують в результаті застосування ітераційного процесу, при певних припущеннях збігається до деякої стаціонарної точки x^* функції *f(x)*. Якщо матриця $H(x_к)$ додатньо визначена на кожному кроці, то послідовність $\{x_k\}$збігається до точки x^*, яка буде точкою строгого локального мінімуму.
Якщо функція *f(x)* є квадратичною, то за допомогою методу Ньютона його мінімум знаходиться за один крок, незалежно від початкового наближення x_0. Це обумовлено тим, що напрям спуску $P_k = -H^{-1}(x_k) f'(x_k)$ в довільних точках x_0 завжди співпадає з напрямком в точку мінімуму x^*.
Якщо ж функція *f(x)* не квадратична але опукла, метод Ньютона гарантує її монотонне спадання від ітерації до ітерації. При мінімізації "овражних" функцій швидкість збіжності методу Ньютона більш висока у порівнянні з градієнтними методами. У цьому випадку вектор $P_к$ не вказує напрям до точки мінімуму функції *f(x)*, однак має більшу складову вздовж вісі "оврагу" і значно ближче до напрямку на мінімум, ніж антиградієнт.
Суттєвим недоліком методу Ньютона є залежність збіжності для не опуклих функцій від початкового наближення x_0. Якщо x_0 знаходиться далеко від точки мінімуму, то метод може розійтися. Збіжність методу, незалежно від початкового наближення, забезпечується вибором напрямку спуску $P_k = -H^{-1}(x_k) f'(x_k)$ і розміром кроку *a* вздовж цього напрямку. Відповідний алгоритм називають методом Ньютона з регулюванням кроку. Ітераційний процес в такому випадку описується виразом

$$x_{k+1} = x_k - a_k H^{-1}(x_k) f'(x_k).$$

Розмір кроку a_k обирається з умови мінімуму функції *f(x)* по *a* у напрямку руху, тобто в результаті розв'язання задачі одновимірної мінімізації:

$$f\left(x_k - a_k H^{-1}(x_k) f'(x_k)\right) = \min\left(f\left(x_k - a_k H^{-1}(x_k) f'(x_k)\right)\right).$$

У наслідок накопичення помилок в процесі рахунку матриця Гессен на деякій ітерації може стати від'ємно визначеною або її буде неможливо обернути. В таких випадках в підпрограмах

оптимізації кладеться $H^{-1}(x_k) = E,$ де E одинична матриця. Очевидно, що ітерація при цьому буди виконуватися по методу найшвидшого спуску.
Кількість обчислень для виконання однієї ітерації по методу Ньютона значно більша, ніж у градієнтних методах. Це пояснюється необхідністю обчислення й обернення матриці других похідних цільової функції. Однак на отримання розв'язку з достатньо високим ступенем точності за допомогою методу Ньютона зазвичай потрібна значно менша кількість ітерацій, ніж при використанні градієнтних методів. В силу цього метод Ньютона суттєво ефективніший.
У певних випадках доречно комбіноване використання градієнтних методів і метода Ньютона. На початку процесу мінімізації, коли точка x_0 знаходиться далеко від точки екстремуму x^* можна застосовувати який-небудь варіант градієнтних методів. При зменшенні швидкості збіжності градієнтного методу буде доречним перехід до методу Ньютона.

Порядок виконання роботи

1. Ознайомитися з методичними вказівками і методичними вказівками до роботи.
2. Отримати варіант індивідуального завдання.
3. Розробити блок-схему алгоритму розв’язання задачі.
4. Написати, налагодити й виконати програму. В програмі передбачити опис усіх використаних масивів.
5. Обчислити точності оцінки алгоритмів по критерію оптимальності

$\varepsilon = \left| f(x_k) - f(x^*) \right|$

або по координатам $\delta = \left\| x_k - x^* \right\|,$

де k задана кількість ітерацій;

x^* точка мінімуму функції.

6. Оформити звіт по роботі.

Зміст звіту

1. Постановка завдання, вихідні дані для розрахунку.
2. Блок-схема алгоритму розв’язання завдання.
3. Опис програми.
4. Лістінги програми і протоколи результатів розв’язання задачі.
5. Висновки по роботі.

Контрольні питання

1. Загальна характеристика методів нульового порядку:
a. прямого пошуку;
b. деформованого многогранника;
c. координат, що обертаються;
d. паралельних дотичних.

2. Загальна характеристика градієнтних методів:
a. метод найшвидшого спуску;
b. метод спряжених градієнтів.
3. Метод Ньютона, збіжність метода.
4. Особливості мінімізації “овражних ” функцій. Швидкість збіжності методів першого і другого порядків.

Варіанти завдань

Варі-Ант	Функція	Початковий вектор	Точка мінімуму	Значен-ня
1	$(1-x_1)^2+(x_1-x_2)^2+(x_2-x_3)^2+(x_3-x_4)^2$	[-2;3;-4;5]	[1;1;1;1]	0
2	$(x_1-1)^2+10(x_2-1)^2+100(x_3-1)^2+1000(x_4-1)^2$	[-1;-2;-3;-4]	[1;1;1;1]	0
3	$100(x_2-x_1^2)^2+(1-x_1)^2$	[3;4]	[1;1]	0
4	$100(x_2-x_1^3)^2+(1-x_1)^2$	[-1;2;1]	[1;1]	0
5	$(x_1+10x_2)^2+5(x_3-x_4)^2+(x_2-x_3)^2+10(x_1-x_4)^2$	[3;-1;0;1]	[0;0;0;0]	0
6	$x_1^2+x_2^2+x_3^2+x_4^2+16x_1^2x_2^2+8x_2^2x_3^2+x_3^2+x_4^2+2$	[1;2;3;4]	[0;0;0;0]	2
7	$10(x_1-\sin x_2)^2+0{,}1x_2^2$	[1;2;3]	[0;0]	0
8	$(1{,}5-x_1(1-x_2))^2+(2{,}25-x_1(1-x_2^2))^2+(2{,}625-x_1(1-x_3^2))^2$	[0;0]	[3;0,5]	0
9	$100(x_2-x_1^2)^2+(1-x_1)^2+90(x_4-x_3^2)^2+(1+x_3)^3+$ $+10{,}1((x_2-1)^2+(x_4-1)^2)+19{,}8(x_2-1)(x_4-1)$	[-3;-1;-3;-1]	[1;1;1;1]	0
10	$-x_1-2x_3-x_2x_3+x_1^2+x_2^2+x_3^2$	[0;0;0]	[1/2;2/3;4/3]	19/12
11	$x_1^3+x_2^3-3x_1x_2$	[-1;3]	[1;1]	-1
12	$2x_1x_2x_3-4x_1x_3-2x_2x_3+x_1^2+x_2^2+x_3^2-2x_1-4x_2+4x_3$	[-3;-3;-3]	[1;2;0]	-5
13	$x_1^4+x_2^4-(x_1+x_2)^2$	[0;0]	[1;1],[-1;-1]	-2
14	$x_1^2+x_2^2+x_3^2-x_1x_2+x_1-2x_3$	[2;2;2]	[2/3;1/3;1]	-4/3
15	$3x_1^3-x_1+x_2^3-3x_2^2-1$	[-1;-1]	[1/3;2]	-47/9
16	$6x_1+2x_1^2-2x_1x_2+2x_2^2$	[-1;-1]	[-2;-1]	-6
17	$x_1+x_2^2+((x_1+x_2-10)/3)^2$	[-1;-1]	[5;0,5]	7,5
18	$(x_1-1)^2+100(x_1-x_2)^2$	[3;4]	[1;1]	0
19	$5(x_1-3)^2+(x_2-5)^2$	[0;0]	[3;5]	0
20	$x_1^2-x_1x_2+x_2^2$	[1;2]	[0;0]	0
21	$9x_1^2+16x_2^2-90x_1-128x_2$	[0;3]	[5;4]	-481
22	$2x_1^2+2x_2^2+2x_1x_2-4x_1-6x_2$	[1;1]	[1/3;3/4]	-14/3
23	$x_1^2-x_1x_2+x_2^2-2x_1+x_2$	[3;5]	[1;0]	-1
24	$5x_1+4x_1x_2+x_2^2-16x_1-12x_2$	[1;1]	[-4;14]	-152
25	$2x_1^2+2x_2^2+x_1x_2-11x_1-8x_2$	[-3;5]	[2;3]	-23
26	$x_1-x_2+2x_1^2+2x_1x_2+x_2^2$	[1;1]	[-1;1,5]	-1,25
27	$x_1^2+x_2^2+x_1x_2$	[1;1]	[0;0]	0
28	$x_1^2+16x_2^2$	[2;2]	[0;0]	0
29	$(1-x_1)^2+(x_1-x_2)^2$	[-5;-8]	[1;1]	0
30	$x_1^2+4x_2^2+1$	[3;5]	[0;0]	1

Лабораторна робота №5

Лінійне програмування

Мета роботи – побудова математичних моделей лінійного програмування, практичне розв'язання задач лінійного програмування з використанням СКМ.

Методичні вказівки

У загальному випадку задача лінійного програмування формулюється в такий спосіб:

$$f(x) = c_1x_1 + \ldots + c_nx_n \to min\ (max)$$

при обмеженнях

$$\begin{aligned}
&a_{1,1}x_1 + \ldots + a_{1,n}x_n \le b_1;\\
&\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\\
&a_{m,1}x_1 + \ldots + a_{m,n}x_n \le b_m;\\
&a_{m+1,1}x_1 + \ldots + a_{m+1,n}x_n \ge b_{m+1};\\
&\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\\
&a_{k,1}x_1 + \ldots + a_{k,n}x_n \ge b_k;\\
&a_{k+1,1}x_1 + \ldots + a_{k+1,n}x_n = b_{k+1};\\
&\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\\
&a_{1,1}x_1 + \ldots + a_{1,n}x_n = b_1;\\
&x_i \ge 0;\ i = \overline{1,n}.
\end{aligned}$$

Задача може бути розв'язана або повним перебором многогранника, утвореного перетинаючимися гіперплощинами, що задаються умовами обмеженнями /для більшості практичних задач чисто теоретичний спосіб/, або з використанням симплекс-методу, що реалізує спрямований перебір.

Розглянемо кілька різновидів задачі лінійного програмування.

1. Задача про використання ресурсів. Для виготовлення n продукції $P_1,\ldots,P_n$ підприємство використовує m видів ресурсів $S_1,\ldots,S_m$. Запаси кожного виду обмежені і рівні $b_1,\ldots,b_m$. На виготовлення одиниці продукції j-го виду ($j=\overline{1,m}$) витрачається a_{ij} одиниць i-го ресурсу ($i=\overline{1,n}$). При реалізації j-ї продукції підприємство одержує c_j одиниць прибутку. Необхідно скласти такий план випуску продукції, щоб при її реалізації отримати максимальний прибуток.

2. Задача про суміші. До цього типи відносяться різноманітні задачі на складання раціону харчування, сумішей з кількох компонентів для одержання кінцевого продукту з заданими властивостями.

Маємо n продуктів $P_1,\ldots,P_n$, що вміщують m поживних речовин $S_1,\ldots,S_m$. Нехай a_{ij}, $j=\overline{1,m}$, $i=\overline{1,n}$ – кількість одиниць j-ї поживної речовини в одиниці i-го продукту; c_i – вартість одиниці i-го продукту. Потрібно вибрати такий раціон харчування /тобто призначити кількість продуктів $P_1,...,P_n$, що входять у нього/, щоб умови по поживних речовинах були виконані, а вартість раціону була мінімальною.

3. Задача про завантаження устаткування. Розглянемо два варіанти цієї задачі.

3.1. Задача про максимальний прибуток.

Підприємство випускає n видів виробів $P_1,...,P_n$, кожний з яких проходить послідовну обробку на верстатах типів $T_1,...,T_m$. Запас потужності верстатів, тобто робочий час верстата, складає відповідно $b_1,...,b_m$ одиниць часу. Виріб P_i обробляється на верстаті Tj-типу a_{ij} одиниць часу. При реалізації один виріб P_i приносить підприємству c_i одиниць прибутку. Потрібно скласти такий план завантаження верстатів, при якому підприємство дістане максимальний прибуток.

3.2. Задача про мінімальну собівартість.

Підприємству необхідно випустити n видів виробів $P_1,...,P_n$ у кількості $N_1,...,N_n$ одиниць. Для цієї мети використовують m типів верстатів $T_1,...,T_n$, кожний з яких може обробляти усі вироби P_1. продуктивність кожного верстата /кількість виробів, оброблюваних за одиницю часу/ має величину a_{ij}. Собівартість кожного виробу при обробці його на тому чи іншому верстаті складає величину c_{ij}. Запас потужності верстатів /робочий час верстата/ складає відповідно $b_1,...,b_m$ одиниць часу. Потрібно скласти такий план завантаження верстатів, при якому собівартість продукції буде мінімальною.

Порядок виконання роботи

1. Ознайомитися з теоретичним матеріалом і методичними вказівками до роботи.
2. Одержати варіант індивідуального завдання /вихідні дані для задачі, метод розв’язання задачі/. Формалізувати задачу лінійного програмування: скласти математичну модель у загальному вигляді; використовуючи конкретні числові дані перетворити вихідну математичну модель до вигляду, що допускає застосування підпрограм СКМ.
3. Розробити блок-схему алгоритму розв’язання задачі.
4. Написати, налагодити і виконати програму. У програмі передбачити опис усіх використовуваних масивів і введення значень їх елементів, вивід вихідних даних, результатів розрахунків і значення ознаки результату.
5. Розв’язати задачу лінійного програмування.
6. Оформити звіт по роботі.

Зміст звіту

1. Постановка задачі, вихідні дані.
2. Блок-схема алгоритму розв’язання задачі.
3. Опис програми.
4. Лістінг програми.
5. Висновки по роботі.

Контрольні питання

1. Формулювання задачі лінійного програмування.
2. Еквівалентні форми задачі лінійного програмування, шляхи переходу від однієї форми до іншої.
3. Геометрична інтерпретація області допустимих значень.
4. Сутність сімплекс-методу.
5. Модифікований сімплекс-метод.
6. Сімплекс-метод зі штучним базисом.

Варіанти завдань

Задача на використання ресурсів

Варіант	Види ресурсів	Витрати ресурсів на одиницю продукції			Запаси ресурсів	Прибуток від реалізації одиниці продукції		
		P_1	P_2	P_3		C_{P_1}	C_{P_2}	C_{P_3}
1	S_1	2	1	1	25	6	5	5
	S_2	1	1	1	14			
	S_3	0	4	2	19			
	S_4	3	0	1	24			
2	S_1	2	5	-	300	5	8	-
	S_2	4	5	-	400			
	S_3	3	0	-	100			
	S_4	0	4	-	200			
3	S_1	2	5	-	20	50	40	-
	S_2	8	5	-	40			
	S_3	5	6	-	30			
4	S_1	2	3	-	19	7	5	-
	S_2	2	1	-	13			
	S_3	0	3	-	15			
	S_4	3	0	-	18			
5	S_1	4	2	1	150000	100	150	200
	S_2	6	0	2	170000			
	S_3	0	2	4	100000			
	S_4	8	7	0	200000			

Задача о сумішах

Варіант	Види ресурсів	Кількість одиниць поживних речовин на одиницю продукції				Максимальна норма поживних речовин	Вартість одиниці продукції			
		P_1	P_2	P_3	P_4		C_{P_1}	C_{P_2}	C_{P_3}	C_{P_4}
6	S_1	3	1	1	-	9	5	6	-	-
	S_2	1	2	-	-	8				
	S_3	1	6	-	-	12				

7	S_1 S_2 S_3	1,2 1,4 0,8 - 80 280 240 - 5 5 100 -	1,6 200 10	3 4 5 -
8	S_1 S_2 S_3	26,5 7,8 0 0 51 26 45,7 0 0 0 5 72,5	21 30 500	14,4 16 12,8 10,5
9	S_1 S_2 S_3 S_4 S_5	1 5 - - 3 2 - - 2 4 - - 2 2 - - 1 0 - -	10 12 16 10 1	2 3
10	S_1 S_2 S_3	0,18 0,24 1,2 - 10 8 200 - 15 1 1,5 -	12 1000 450	1 1,1 7,5 -

Задача на максимальний прибуток

Варіант	Види станків	Тривалість обробки виробу на станку	Прибуток від реалізації одиниці продукції	Запас потужності станків
		P_1 P_2 P_3	C_{P_1} C_{P_2} C_{P_3}	
11	T_1 T_2 T_3	12 10 9 15 18 20 6 4 4	30 32 29	13200 24000 6000
12	T_1 T_2 T_3 T_4	2 5 - 2 1 - 5 6 - 1 10 -	1 1 -	50 20 60 90
13	T_1 T_2 T_3	3 8 4 2 3 2 7 9 5	16 25 20	6048 6048 3932
14	T_1 T_2 T_3	2 3 - 2 1 - 0 3 -	11 9 -	20 37 30
15	T_1 T_2 T_3	2 0 - 1 2 - 1 4 -	6 6 -	20 37 30

адача на мінімальну собівартість

Варі-ант	Типи станків	Продуктивність станків				Собівартість продукції				План випуску продукції				Запас потужності станків
		P1	P2	P3	P4	C_{P_1}	C_{P_2}	C_{P_3}	C_{P_4}	N_{P_1}	N_{P_2}	N_{P_3}	N_{P_4}	
16	T_1	30	20	-	-	6	12	-	-					120
	T_2	20	14	-	-	8	10	-	-	4000	3000	-	-	100
	T_3	15	25	-	-	11	7	-	-					160
17	T_1	6	24	-	-	4	47	-	-	30	96	-	-	6
	T_2	13	13	-	-	13	26	-	-					6
18	T_1	30	50	30	20	2	1	0,5	1,2					240
	T_2	60	100	60	40	0,8	1,2	0,9	0,8	3	15	4,5	1,5	150
	T_3	18	30	18	12	0,5	1	0,6	0,9					150
19	T_1	8	4	2	-	4	6	2	-	160	100	100	-	60
	T_2	4	2	1	-	5	4	2	-					70
20	T_1	5	10	20	-	6	3	1,5	-					40
	T_2	1,7	3,3	5	-	6	3	2	-	300	500	100	-	60
	T_3	5	10	2,5	-	4	2	8	-					30

Лабораторна робота №6

Транспортна задача

Мета роботи – вивчити методи розв'язання транспортної задачі, практичне розв'язання транспортної задачі лінійного програмування на ЕОМ за допомогою СКМ.

Методичні вказівки

Вихідна інформація:

a_i – кількість одиниць вантажу в i-м пункті відправлення, $i = \overline{1,m}$;

b_j – потреба в j-м пункті призначення $j = \overline{1,n}$;

c_{ij} – вартість перевезення одиниці вантажу з i-го пункту в j-й;

x_{ij} – запланована кількість одиниць вантажу для перевезення з i-го пункту в j- й;

$\sum_{i=1}^{m}\sum_{j=1}^{n} c_{ij}x_{ij}$ – загальна вартість перевезень;

$\sum_{j=1}^{n} x_{ij}$ – кількість вантажу, що вивозиться з i-го пункту;

$\sum_{i=1}^{m} x_{ij}$ – кількість вантажу, що доставляється в j-й пункт.

Математичне формулювання транспортної задачі:

$$min \sum_{i=1}^{m}\sum_{j=1}^{n} c_{ij}x_{ij} \text{ за умови } \sum_{j=1}^{n} x_{ij} = a_i, \sum_{i=1}^{m} x_{ij} = b_j, x_{ij} \geq 0 \, (i = \overline{1,m}, j = \overline{1,n}).$$

Ця задача має назву замкнутої транспортної моделі, а умова $\sum_{i=1}^{n} a_i = \sum_{j=1}^{m} b_j$ є умовою можливості розв'язання замкнутої транспортної задачі.

Більш загальним випадком транспортної задачі є так звана відкрита транспортна модель: знайти min $\sum_{i=1}^{m}\sum_{j=1}^{n} c_{ij}x_{ij}$ за умови $\sum_{j=1}^{n} x_{ij} \leq a_i$, $\sum_{i=1}^{m} x_{ij} = b_j$, $x_{ij} \geq 0 \, (i = \overline{1,m}, j = \overline{1,n})$ чи за умови

min $\sum_{i=1}^{m}\sum_{j=1}^{n} c_{ij}x_{ij}$ за умови $\sum_{j=1}^{n} x_{ij} = a_i$, $\sum_{i=1}^{m} x_{ij} \leq b_j$, $x_{ij} \geq 0 \, (i = \overline{1,m}, j = \overline{1,n})$.

Розв'язання відкритої транспортної задачі зводиться до розв'язання замкнутої транспортної задачі шляхом введення фіктивного пункту відправлення чи пункту призначення з нульовою вартістю перевезень.

Як додаткові умови при розв'язанні транспортної задачі можуть бути поставлені обмеження пропускну здатність ліній перевезення вантажів з пунктів відправлення в пункти призначення.

$$0 \leq x_{ij} \leq a_{ij}$$

Вводиться матриця $D = \|\alpha_{ij}\|$ – матриця обмежень на пропускну здатність. Якщо на пропускну здатність якої-небудь лінії перевезень немає обмежень, то відповідний елемент матриці $\alpha_{ij}=\infty$.
Якщо при розв'язанні транспортної задачі враховується матриця $D = \|\alpha_{ij}\|$ і виконується хоча б одна з нерівностей

$$\sum_{i=1}^{n}\alpha_{ij} < a_i \quad \text{чи} \quad \sum_{j=1}^{m}\alpha_{ij} < b_{ij},$$

то транспортна задача не має розв'язку.
Розв'язання транспортної задачі починається з побудови опорного /припустимого/ плану перевезень. Опорний розв'язок завжди існує, більш того, завжди можна одержати оптимальний план, тому що $\sum_{i=1}^{m}\sum_{j=1}^{n}c_{ij}x_{ij} \geq 0$ /ціла функція обмежена знизу, мінімум існує завжди/. Опорний план звичайно складається методом "північно-західного кута" чи методом мінімального елемента. Надалі виконується ітераційне поліпшення плану за критерієм вартості перевезень. Для цього використовується алгоритм угорського методу або методу потенціалів.
Опорний план для транспортної задачі з обмеженнями на пропускну здатність також може бути складений на основі методу мінімального елемента, але вже з урахуванням матриці $D = \|\alpha_{ij}\|$.
Оптимізація плану перевезень проводиться, як правило, за допомогою модифікованого методу потенціалів.

Порядок виконання роботи

1. Ознайомитися з теоретичним матеріалом і методичними вказівками до роботи.
2. Одержати варіант індивідуального завдання /вихідні дані транспортної задачі, метод розв'язання задачі/.
3. Розробити блок-схему алгоритму розв'язання задачі.
4. Написати, налагодити і виконати програму. У програмі описати усі масиви і значення, що вводяться, здійснити вивід вихідних даних і результатів, передбачити перевірку замкненості задачі. Для транспортної задачі з обмеженнями на пропускну здатність виконати перевірку на існування розв'язання.
5. Розв'язати транспортну задачу.
6. Оформити звіт по роботі.

Зміст звіту

1. Постановка задачі, вихідні дані.
2. Блок-схема алгоритму розв'язання задачі.
3. Опис програми.
4. Лістінг програми і протоколи розв'язання задачі.
5. Висновки по роботі.

Контрольні питання

1. Формулювання транспортної задачі. Визначення рангу системи рівнянь.
2. Постановка транспортної задачі з обмеженнями на пропускну здатність. Умови можливості розв'язання.

3. Розв’язання задач з надлишком запасів чи заявок.
4. Методи знаходження опорного плану.
5. Сутність угорського методу.
6. Метод потенціалів.

Варіанти завдань

(транспортна здача)

Варіант	Транспортні витрати (матриця С)	Об’єм виробництва (вектор а)	Об’єм потреб (вектор b)
1	16 30 17 10 16 30 27 26 9 23 13 4 22 3 1 3 1 5 4 24	4 6 10 10	7 7 7 7 2
2	15 1 22 19 1 21 18 11 4 3 26 29 23 26 24 21 10 3 19 27	20 20 20 20	19 19 19 19 4
3	20 26 24 26 29 15 20 29 26 23 4 10 27 30 7 9 16 29 20 3	13 17 17 13	12 12 12 12 12
4	10 17 9 20 30 15 4 24 26 26 22 24 30 27 29 25 12 11 24 23	15 15 19 11	9 24 9 9 9
5	5 15 3 10 10 23 8 13 9 12 27 14 14 3 18 8 26 7 4 9	9 11 15 16	8 9 13 8 12
6	30 2 5 6 15 5 29 9 4 7 16 24 14 6 26 13 28 4 25 8	16 15 14 15	6 6 13 20 15

7	2 24 4 2 7 20 10 15 27 9 15 15 12 25 4 2 6 3 5 29	28 13 15 30	27 16 25 11 7
8	30 22 2 13 7 26 10 4 24 9 27 16 25 5 4 6 11 17 10 29	18 12 17 13	8 8 8 8 28
9	30 24 11 12 25 26 4 29 20 24 27 14 14 10 18 6 14 28 8 2	21 19 15 25	15 15 15 15 20
10	9 17 29 28 8 13 21 27 16 29 20 30 24 7 26 11 19 30 6 2	22 13 17 18	7 7 7 7 42
11	12 11 25 17 21 22 18 14 8 1 9 13 2 28 15 26 21 3 4 27	17 14 21 43	19 22 23 17 14
12	22 6 25 11 12 13 14 20 27 30 16 7 19 10 21 26 29 23 25 18	9 18 23 26	11 22 31 6 6
13	22 24 25 23 29 1 21 10 7 19 2 26 18 30 27 22 10 29 26 23	24 14 19 17	22 9 12 13 18
14	7 10 16 27 19 30 18 8 29 15 3 18 28 19 18 9 12 2 25 21	17 19 11 13	5 15 11 9 20
15	5 3 24 10 25 30 2 22 16 7 30 24 27 29 10 15 17 21 2 3	24 15 16 24	12 13 14 31 9

16	25 28 20 15 7 27 5 11 23 10 1 25 14 16 16 8 6 4 16 18	16 12 14 18	7 8 4 11 30
17	8 1 19 1 15 8 27 30 7 7 9 20 19 26 20 23 28 25 7 22	18 23 17 22	21 21 9 9 20
18	6 11 20 17 8 1 25 3 18 17 9 39 16 30 31 23 15 4 3 28	12 17 18 13	10 8 12 14 16
19	4 21 17 8 1 20 8 25 15 23 17 1 11 5 3 18 10 24 6 5	21 21 23 23	22 22 22 11 11
20	21 19 11 12 12 26 29 14 1 26 39 1 22 8 25 53 23 40 26 28	24 12 18 13	11 13 26 10 10
21	14 25 18 19 23 2 17 26 24 2 29 3 7 15 22 5 20 17 23 10	33 25 25 17	33 11 11 11 34
22	30 20 27 15 26 25 6 28 20 5 19 24 11 29 23 1 4 6 6 8	33 33 33 11	22 22 22 22 22
23	11 10 15 8 7 12 14 29 20 20 18 7 5 25 28 24 4 30 24 26	16 15 24 15	15 15 15 15 10
24	29 6 29 19 21 14 3 30 10 10 15 27 28 11 24 1 23 25 15 13	13 27 16 14	14 14 14 14 14

25	29 4 7 6 16 21 13 25 21 7 20 10 12 6 2 17 7 4 6 19	14 14 14 18	12 12 12 12 12
26	17 29 2 8 18 14 8 25 15 21 29 11 12 13 20 27 15 4 8 14	32 8 13 27	15 15 15 15 20
27	30 17 26 14 3 18 14 27 7 20 8 24 17 17 26 1 18 21 16 12	24 8 12 16	11 11 11 11 16
28	29 53 39 29 22 15 33 16 3 3 16 27 16 3 5 35 50 39 20 23	33 18 32 17	20 20 20 20 20
29	28 26 12 22 26 20 23 25 22 28 23 15 11 22 26 1 26 10 11 7	24 27 16 13	16 16 16 16 16
30	20 5 27 10 26 7 7 18 21 28 27 12 9 23 26 1 24 17 23 7	15 25 5 15	7 8 13 12 20
31	14 5 27 29 23 17 7 16 19 2 20 12 15 29 5 14 24 18 7 13	18 14 16 12	8 11 11 9 21
32	16 10 7 5 13 12 28 25 9 10 14 15 18 9 28 25 16 21 12 8	34 18 6 12	10 10 10 10 30

(транспортна задача з обмеженнями на пропускну здатність)

Варі-ант	Транспортні витрати	Обмеження на перевезення (матриця D)	Об'єм виробництва (вектор a)	Об'єм потреб (вектор b)

1	14 10 2 5 10 11 5 4 11 3 9 8 12 1 18 1 4 9 7 18	15 35 14 10 5 8 4 20 12 18 10 7 32 20 14 8 4 16 20 17	50 20 30 40	15 30 65 20 10
2	16 1 10 16 12 12 13 7 10 14 1 19 14 13 3 13 8 15 8 19	3 5 6 4 8 23 12 15 16 16 12 20 21 10 2 1 6 21 4 3	10 70 60 30	40 40 60 25 5
3	5 3 20 7 25 8 12 6 12 5 3 1 12 2 19 6 3 3 1 18	9 1 16 5 11 20 19 17 3 6 10 14 30 18 7 18 16 11 13 17	10 48 27 46	17 33 38 21 22
4	7 19 7 12 18 17 11 7 13 11 1 13 19 18 12 8 14 11 3 11	20 6 15 22 25 2 5 2 3 4 20 1 3 15 8 40 5 6 2 10	80 12 38 45	75 10 20 40 30
5	10 16 3 8 15 3 14 12 9 1 2 20 4 11 5 7 17 13 8 15	8 2 3 2 2 15 3 3 4 5 10 20 15 5 10 11 20 7 4 21	14 25 56 45	40 40 20 10 30
6	16 12 3 9 10 4 16 1 11 10 19 10 18 20 19 12 4 11 18 19	5 3 5 4 10 7 5 5 10 4 12 7 6 4 3 10 17 15 35 20	15 17 23 75	30 27 16 33 24
7	10 9 12 7 1 16 4 9 19 10 20 17 11 1 12 13 14 15 7 8	10 4 15 8 10 20 21 23 16 15 5 10 8 25 12 4 5 6 7 10	30 70 50 20	35 30 35 45 25
8	5 19 12 5 9 17 20 11 10 9 2 15 12 13 6 18 3 8 5 14	2 6 18 5 25 3 4 1 7 8 9 5 3 3 7 15 7 21 22 6	50 20 10 30	5 15 35 15 40
9	5 4 20 5 14 19 20 2 4 5 9 6 8 19 1 7 15 20 6 11	5 20 14 25 13 19 22 15 4 12 6 8 15 5 10 14 7 2 23 20	60 50 40 20	10 50 40 50 20

10	17 18 7 20 2 5 16 3 9 19 15 14 17 7 5 11 15 9 8 16	7 10 4 5 8 5 4 6 5 9 11 6 12 8 7 3 2 1 6 10	10 13 28 17	11 10 14 16 17
11	14 13 6 12 1 18 20 4 2 3 17 5 15 10 8 4 12 18 13 16	7 6 5 5 3 10 8 11 2 4 20 15 38 9 7 11 13 41 20 14	20 30 70 71	37 26 91 24 13
12	3 17 6 19 2 1 15 7 6 1 5 13 8 11 17 18 13 17 1 8	5 7 6 5 4 20 3 20 4 2 24 13 7 11 8 9 21 16 13 25	20 40 52 73	45 38 40 28 34
13	14 20 7 17 8 18 4 14 9 7 17 15 16 12 20 19 2 1 11 7	15 14 20 7 3 12 8 16 15 4 15 10 13 11 5 3 5 17 4 20	29 39 28 42	25 13 60 15 25
14	5 8 13 14 9 11 20 18 19 17 1 15 2 8 3 9 16 15 11 17	20 25 10 20 2 4 7 6 10 15 5 6 12 20 30 10 30 15 25 5	62 26 27 65	30 50 13 70 17
15	2 8 17 13 11 5 2 3 16 1 5 9 7 14 18 4 11 2 14 15	30 11 10 25 10 4 19 20 20 17 15 13 10 20 25 10 11 2 20 12	71 69 58 32	50 30 25 80 45
16	17 1 3 5 2 4 7 20 19 13 8 18 3 11 12 15 17 16 11 10	7 13 12 7 7 12 10 14 10 4 6 19 10 1 3 10 8 22 24 5	33 44 24 55	20 36 50 40 10
17	1 3 18 7 2 9 4 25 20 8 19 17 12 8 15 4 3 20 2 14	10 26 23 8 9 6 18 30 5 5 4 2 25 3 10 5 4 11 2 1	53 45 38 15	21 30 75 10 15
18	9 1 8 7 8 8 1 6 9 5 20 8 15 14 19 12 6 10 10 12	14 10 15 20 18 19 8 15 18 13 11 23 18 30 16 9 10 11 24 32	26 28 39 75	3 46 25 49 55

19	10 18 12 8 13 17 2 18 13 7 4 3 1 6 12 12 16 8 14 15	6 5 7 3 2 20 4 10 30 4 8 15 2 41 22 32 12 10 3 23	11 64 35 75	55 30 20 33 44
20	5 13 10 5 19 1 10 16 15 7 13 17 20 18 9 16 10 5 6 16	5 5 4 6 7 20 10 9 13 8 4 12 4 11 3 30 15 8 13 19	15 40 14 76	55 30 11 30 19
21	18 12 7 15 12 14 17 11 2 7 19 6 11 12 6 10 8 20 1 13	10 9 10 6 8 4 31 5 5 17 11 2 8 15 16 5 10 3 6 4	19 52 34 15	20 40 17 10 33
22	5 16 3 6 10 6 4 20 1 11 19 6 12 19 7 15 10 18 17 8	10 10 16 17 9 5 11 30 7 4 5 2 10 3 2 10 5 16 1 11	50 42 17 36	19 20 66 18 22
23	16 9 2 3 17 12 4 19 10 5 6 12 11 13 2 5 19 10 4 1	17 7 12 10 4 3 15 10 10 5 10 3 27 4 11 30 14 30 10 12	27 35 49 94	55 30 77 20 23
24	18 14 6 8 7 8 19 9 16 17 18 10 14 7 19 1 6 10 5 13	12 20 30 20 10 20 4 3 2 8 13 10 30 10 15 23 5 6 4 6	72 29 68 26	65 24 50 30 26
25	11 12 17 15 13 1 4 6 2 5 7 11 9 16 19 2 10 5 18 6	11 12 17 12 15 4 13 17 13 6 15 27 13 41 22 20 3 25 20 12	13 44 77 62	36 40 50 35 35
26	14 19 17 12 20 16 1 4 7 5 11 3 9 7 19 12 16 11 5 13	27 17 4 8 10 20 5 12 11 20 20 4 20 12 22 7 20 7 5 10	37 61 72 27	67 30 37 23 40
27	14 8 10 6 11 17 1 19 16 17 12 11 15 8 16 10 1 5 14 17	14 15 9 10 6 4 20 15 4 32 10 4 3 7 10 24 35 20 5 7	29 66 17 82	44 50 40 23 37