

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА»

Кафедра інженерії програмного забезпечення

**КУРСОВИЙ ПРОЕКТ**  
(ПОЯСНЮВАЛЬНА ЗАПИСКА)

з дисципліни: «Математичні методи дослідження операцій»

на тему:

**«РОЗВ’ЯЗАННЯ ЗАДАЧІ МАТЕМАТИЧНОГО ПРОГРАМУВАННЯ»**

студента III курсу групи ІПЗ-20-4  
спеціальності 121 «Інженерія програмного  
забезпечення»

Васьківського Віталія Юрійовича  
(прізвище, ім'я та по-батькові)

Керівник старший викладач кафедри  
ІПЗ Локтікова Т.М.

Дата захисту: " \_\_\_\_ " \_\_\_\_\_ 20\_\_ р.  
Національна шкала \_\_\_\_\_  
Кількість балів: \_\_\_\_\_  
Оцінка: ECTS \_\_\_\_\_

Члени комісії

_____	<u>Т.М. Локтікова</u>
(підпис)	(прізвище та ініціали)
_____	<u>Н.О. Кушнір</u>
(підпис)	(прізвище та ініціали)
_____	<u>В.П. Полторак</u>
(підпис)	(прізвище та ініціали)

Міністерство освіти і науки України  
Державний університет «Житомирська політехніка»  
Факультет інформаційно-комп'ютерних технологій  
Кафедра інженерії програмного забезпечення

ЗАТВЕРДЖУЮ

В.о. зав. каф. ІПЗ, к.т.н., доц.

Андрій МОРОЗОВ

(підпис)

«\_\_\_» \_\_\_\_\_ 2023 р.

## ЗАВДАННЯ

### до курсової роботи

студенту 3-го курсу групи ІПЗ-20-4 ВАСЬКІВСЬКОМУ Віталію

1. **Тема роботи:** Розв'язання задачі математичного програмування (МП).

2. **Термін здачі** студентом закінченої роботи: червня 2023 р.

3. **Вихідні дані** до роботи: підприємство може працювати по п'яти технологічних процесах ( $T_1$ ,  $T_2$ ,  $T_3$ ,  $T_4$ ,  $T_5$ ), причому кількість одиниць продукції, що випускається, по різних технологічних процесах за одиницю часу відповідно дорівнює 300, 260, 320, 400 і 450 шт. У процесі виробництва враховуються такі виробничі чинники: сировина, електроенергія, зарплата і накладні витрати. Витрати відповідних чинників при роботі по різних технологічних процесах протягом одиниці часу вказані в табл. 1. Потрібно скласти програму максимального випуску продукції.

Таблиця 1

Виробничі фактори	Витрати при різних технологіях					Ліміт
	$T_1$	$T_2$	$T_3$	$T_4$	$T_5$	
сировина	15	20	12	14	18	2000
електроенергія	0,2	0,3	0,15	0,25	0,3	300
накладні витрати	4	5	6	3	2	1000
зарплата	5	3	4	6	3	1600

4. **Перелік питань, які мають бути розроблені:** 1) Вступ, 2) Побудова математичної моделі задачі, 3) Вибір й обґрунтування методу розв'язання задачі МП, 4) Розв'язання задачі МП вручну за допомогою вибраного методу, 5) Розробка алгоритму розв'язання задачі МП, 6) Вибір й обґрунтування інструментальних засобів для реалізації програми розв'язання задачі МП, 7) Написання програми розв'язання задачі МП, 8) Тестування програмного продукту, 9) Висновки, 10) Джерела інформації, 11) Додатки.

5. **Перелік графічного матеріалу:** алгоритм розв'язання задачі МП, скріншоти роботи програми.

6. **Дата видачі завдання:** «\_\_\_» квітня 2023 р.

Завдання прийняв до виконання студент групи ІПЗ-20-4  
Віталій ВАСЬКІВСЬКИЙ

Керівник курсової роботи, ст. викл. каф. ІПЗ

Тамара ЛОКТИКОВА

## РЕФЕРАТ

Пояснювальна записка до курсового проекту на тему «Розв’язання задач математичного програмування» складається з вступу, трьох розділів, висновків, списку використаної літератури та додатку.

Текстова частина викладена на 24 сторінках друкованого тексту.

Пояснювальна записка має 11 сторінок додатків. Список використаних джерел містить 7 найменувань і займає 1 сторінку. В роботі наведено 9 рисунків та 6 таблиць. Загальний обсяг роботи – 31 сторінка.

У додатку представлений лістинг розробленого програмного продукту.

Ключові слова: ЛІНІЙНЕ ПРОГРАМУВАННЯ, МАТЕМАТИЧНЕ ПРОГРАМУВАННЯ, МЕТОД ШТУЧНОГО БАЗИСУ, C#, WINDOWS FORMS.

					ДУ «Житомирська політехніка».23.121.03.000 - ПЗ			
Змн.	Арк.	№ докум.	Підпис	Дата	Розв’язання задач лінійного програмування	Літ.	Арк.	Аркушів
Розроб.		Васьківський В.						
Перевір.		Локтікова Т.М.					3	40
Керівник						ФІКТ Гр. ІПЗ-20-4[1]		
Н. контр.								
Зав. каф.								

## ЗМІСТ

ВСТУП.....	5
1. ПОБУДОВА МАТЕМАТИЧНОЇ МОДЕЛІ ЗАДАЧІ.....	6
Висновки до першого розділу.....	7
2. РОЗВ'ЯЗАННЯ ЗАДАЧІ МАТЕМАТИЧНОГО ПРОГРАМУВАННЯ .....	8
2.1 Вибір й обґрунтування методу розв'язання задачі.....	8
2.2 Розв'язання задачі МП метод штучного базису .....	10
Висновки до другого розділу .....	12
3. РОЗРОБКА ПРОГРАМНОГО ПРОДУКТУ .....	13
3.1. Розробка алгоритму розв'язання задачі МП .....	13
3.2. Вибір інструментальних засобів для реалізації програми .....	16
3.3. Написання програмного продукту .....	17
3.4. Тестування програмного продукту .....	19
Висновки до третього розділу .....	22
ВИСНОВКИ.....	23
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ .....	24
ДОДАТКИ.....	25

## ВСТУП

У житті кожної людини рано чи пізно виникає потреба з'ясувати певну величину, враховуючи відомі фактори, що на неї впливають. І в такий момент лінійне програмування стає незамінним інструментом. Лінійне програмування - це методом досягнення оптимального результату (наприклад, максимізація прибутку або мінімізація витрат) в математичній моделі, де вимоги виражені лінійними залежностями. Воно є окремим випадком математичного програмування (математичної оптимізації).

Існує безліч методів для вирішення таких задач, але одним з найуніверсальніших є метод штучного базису. Саме цей метод становить предмет дослідження даного курсового проекту. Основна мета дослідження - використання методу штучного базису для розв'язання задач лінійного програмування та створення програмної реалізації даного методу.

		Васьківський В.			ДУ «Житомирська політехніка».23.121.03.000 - ПЗ	Арк.
		Локтікова Т.М.				
Змн.	Арк.	№ докум.	Підпис	Дата		5

## 1. ПОБУДОВА МАТЕМАТИЧНОЇ МОДЕЛІ ЗАДАЧІ

Для дослідження розв'язання задач лінійного програмування методом штучного базису було запропоновано наступну задачу:

Підприємство може працювати по п'яти технологічних процесах ( $T_1, T_2, T_3, T_4, T_5$ ), причому кількість одиниць продукції, що випускається, по різних технологічних процесах за одиницю часу відповідно дорівнює 300, 260, 320, 400 і 450 шт. У процесі виробництва враховуються такі виробничі чинники: сировина, електроенергія, зарплата і накладні витрати. Витрати відповідних чинників при роботі по різних технологічних процесах протягом одиниці часу вказані в табл. 1. Потрібно скласти програму максимального випуску продукції.

Таблиця 1.1

Виробничі фактори	Витрати при різних технологіях					Ліміт
	$T_1$	$T_2$	$T_3$	$T_4$	$T_5$	
сировина	15	20	12	14	18	2000
електроенергія	0,2	0,3	0,15	0,25	0,3	300
накладні витрати	4	5	6	3	2	1000
зарплата	5	3	4	6	3	1600

Складемо математичну модель для даної задачі. Позначимо технологічний процес №1 через  $x_1$ , технологічний процес №2 -  $x_2$ , технологічний процес №3 -  $x_3$ , технологічний процес №4 -  $x_4$  та технологічний процес №5 -  $x_5$ . Тоді цільова функція матиме наступний вигляд:

$$F(x_1, x_2, \dots, x_5) = 300x_1 + 260x_2 + 320x_3 + 400x_4 + 450x_5 \rightarrow \max.$$

Тепер сформуємо систему нерівностей з обмежень задачі.

Виходячи з фізичного сенсу змінних  $x_1, x_2, x_3, x_4$  та  $x_5$  - часу, на значення, які можуть приймати змінні накладемо умову їх невід'ємності.

Так як, технологічний процес  $x_1$  і в одиниці міститься 15 сировини, то загальна кількість сировини буде  $15x_1$ . У технологічному процесі №2 загальна кількість сировини складе  $20x_2$ . У технологічному процесі №3 загальна кількість

		Васьківський В.			ДУ «Житомирська політехніка».23.121.03.000 - ПЗ	Арк.
		Локтікова Т.М.				6
Змн.	Арк.	№ докум.	Підпис	Дата		

сировини складе  $12x_3$ . У технологічному процесі №4 загальна кількість сировини складе  $14x_4$ . У технологічному процесі №5 загальна кількість сировини складе  $18x_5$ . А так як норма не повинна бути більшою за 2000, отримуємо наступну нерівність-обмеження:

$$15x_1 + 20x_2 + 12x_3 + 14x_4 + 18x_5 \leq 2000.$$

Аналогічні кроки здійснюємо для електроенергії, накладним витрат та зарплати. Тоді отримуємо наступну систему обмежень:

$$\begin{cases} 15x_1 + 20x_2 + 12x_3 + 14x_4 + 18x_5 \leq 2000 \\ 0,2x_1 + 0,3x_2 + 0,15x_3 + 0,25x_4 + 0,3x_5 \leq 300 \\ 4x_1 + 5x_2 + 6x_3 + 3x_4 + 2x_5 \leq 1000 \\ 4x_1 + 3x_2 + 4x_3 + 6x_4 + 3x_5 \leq 1600 \\ x_1, x_2, x_3, x_4, x_5 \geq 0 \end{cases}$$

Отже, математична модель задачі, яку необхідно розв'язати, матиме вигляд:

$$F(x_1, x_2, \dots, x_5) = 300x_1 + 260x_2 + 320x_3 + 400x_4 + 450x_5 \rightarrow \max$$

$$\begin{cases} 15x_1 + 20x_2 + 12x_3 + 14x_4 + 18x_5 \leq 2000 \\ 0,2x_1 + 0,3x_2 + 0,15x_3 + 0,25x_4 + 0,3x_5 \leq 300 \\ 4x_1 + 5x_2 + 6x_3 + 3x_4 + 2x_5 \leq 1000 \\ 4x_1 + 3x_2 + 4x_3 + 6x_4 + 3x_5 \leq 1600 \\ x_1, x_2, x_3, x_4, x_5 \geq 0 \end{cases}$$

### Висновки до першого розділу

В даному розділі було отримано задачу лінійного програмування – і цільова функція, і умови обмеження задачі, що необхідно розв'язати, є лінійними. Згідно завдання було складено математичну модель задачі.

Було отримано достатній обсяг інформації для розв'язання заданої задачі.

		Васьківський В.			ДУ «Житомирська політехніка».23.121.03.000 - ПЗ	Арк.
		Локтікова Т.М.				7
Змн.	Арк.	№ докум.	Підпис	Дата		

## 2. РОЗВ'ЯЗАННЯ ЗАДАЧІ МАТЕМАТИЧНОГО ПРОГРАМУВАННЯ

### 2.1 Вибір й обґрунтування методу розв'язання задачі

Для того щоб обрати певний метод розв'язання задачі потрібно провести детальний аналіз математичної моделі нашої задачі, отриманої у попередньому розділі.

Дослідивши цільову функцію можна сказати що вона є лінійною, тобто будь-яка змінна не має степенів. Отже наша задача представляє з себе типову задачу лінійного програмування.

Загалом розглянемо три способи розв'язання задач лінійного програмування, а саме: метод потенціалів, симплекс-метод, двоїстий симплекс-метод. Розглянемо кожен з них.

#### - Метод потенціалів:

Ефективний для певних типів задач, має високу адаптивність під конкретні випадки, проте є досить обмеженим у застосуванні, чутливим до великих значень та має потенційну складність обчислень при великій кількості змін та обмежень.

#### - Симплекс-метод:

Є досить універсальним і має широкий спектр застосування у задачах лінійного програмування, непогано показує себе при розв'язанні задач із великою кількістю змінних та обмежень, має високу ефективність і швидкість. Серед недоліків його відносна складність реалізації, велика кількість ітерацій та вразливість до деяких видів задач.

#### - Двоїстий симплекс-метод:

Має високу ефективність коли кількість обмежень перевищує кількість змінних, має перевагу у розв'язанні стандартних задач лінійного програмування, має кращу ітераційну збіжність. В свою чергу цей метод має високу складність реалізації, високу чутливість до початкового базису і має обмежену застосовність.

		Васьківський В.			ДУ «Житомирська політехніка».23.121.03.000 - ПЗ	Арк.
		Локтікова Т.М.				
Змн.	Арк.	№ докум.	Підпис	Дата		8



Було прийнято рішення розв'язати задачу за допомогою симплекс-методу, через його швидкість та універсальність в порівнянні із іншими двома методами. Оскільки виконання курсової роботи передбачає створення програмного продукту, то це стало ключовим фактором у здійсненому виборі.

Проте сам по собі симплекс-метод має певні обмеження стосовно задачі лінійного програмування, тому було прийнято рішення використовувати метод штучного базису (Big M method) для вирішення цієї проблеми. Метод розширеного базису має в своїй основі симплекс-метод і розширює його великими константами. Так ми отримуємо більшу вірогідність правильного і комфортнішого розв'язку задачі.

Загальний алгоритм роботи методу штучного базису виглядає так:

1. Формування початкової задачі.
2. Додавання штучних змінних із константою  $M$ , де  $M$  – велике позитивне число.
3. Створення початкової симплекс-таблиці.
4. Виконання ітерацій симплекс-методу і формування наступних симплекс-таблиць до моменту знаходження оптимального розв'язку.
5. Виключення штучних змінних.

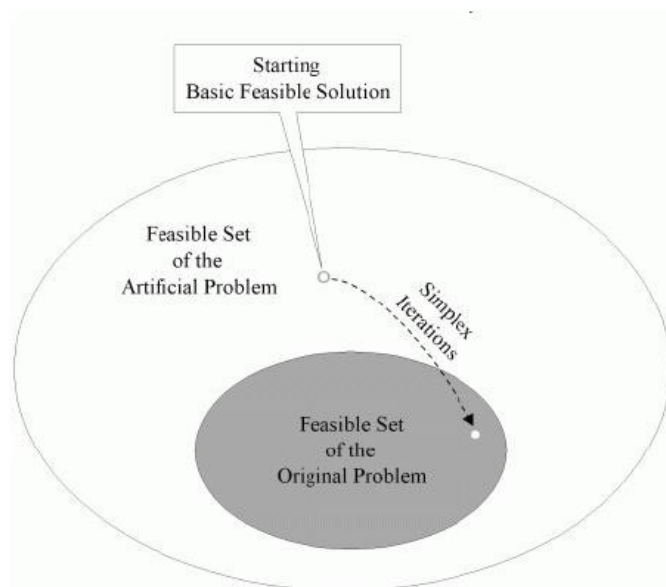


Рисунок 2.1 – відношення методу симплекс-таблиць до методу штучного базису

## 2.2 Розв'язання задачі МП методом штучного базису

Маємо наступну задачу для розв'язання методом штучного базису:

$$F(x_1, x_2, \dots, x_5) = 300x_1 + 260x_2 + 320x_3 + 400x_4 + 450x_5 \rightarrow \max$$
$$\begin{cases} 15x_1 + 20x_2 + 12x_3 + 14x_4 + 18x_5 \leq 2000 \\ 0,2x_1 + 0,3x_2 + 0,15x_3 + 0,25x_4 + 0,3x_5 \leq 300 \\ 4x_1 + 5x_2 + 6x_3 + 3x_4 + 2x_5 \leq 1000 \\ 4x_1 + 3x_2 + 4x_3 + 6x_4 + 3x_5 \leq 1600 \\ x_1, x_2, x_3, x_4, x_5 \geq 0 \end{cases}$$

Маємо привести математичну модель задачі лінійного програмування до канонічного вигляду.

В умовах-обмеженнях обернемо нерівності в рівності через введення вільних змінних  $s_1, s_2, s_3, s_4$ :

$$F(x_1, x_2, \dots, x_5) = 300x_1 + 260x_2 + 320x_3 + 400x_4 + 450x_5 + 0s_1 + 0s_2 + 0s_3 + 0s_4 \rightarrow \max$$
$$\begin{cases} 15x_1 + 20x_2 + 12x_3 + 14x_4 + 18x_5 + s_1 \leq 2000 \\ 0,2x_1 + 0,3x_2 + 0,15x_3 + 0,25x_4 + 0,3x_5 + s_2 \leq 300 \\ 4x_1 + 5x_2 + 6x_3 + 3x_4 + 2x_5 + s_3 \leq 1000 \\ 4x_1 + 3x_2 + 4x_3 + 6x_4 + 3x_5 + s_4 \leq 1600 \\ x_1, x_2, x_3, x_4, x_5 \geq 0 \end{cases}$$

Тоді ми можемо отримати початковий базисний розв'язок задачі:

$$x_1 = 0, x_2 = 0, x_3 = 0, x_4 = 0, x_5 = 0, s_1 = 2000, s_2 = 300, s_3 = 1000, s_4 = 1600$$

Заповнимо вхідну симплекс таблицю:

		Васьківський В.			ДУ «Житомирська політехніка».23.121.03.000 - ПЗ	Арк.
		Локтікова Т.М.				10
Змн.	Арк.	№ докум.	Підпис	Дата		

Таблиця 2.1

	C	-	300	260	320	400	450	0	0	0	0
	B	$x_0$	$x_1$	$x_2$	$x_3$	$x_4$	$x_5$	$s_1$	$s_2$	$s_3$	$a_4$
0	$s_1$	2000	15	20	12	14	18	1	0	0	0
0	$s_2$	300	1/5	3/10	3/20	1/4	3/10	0	1	0	0
0	$s_3$	1000	4	5	6	3	2	0	0	1	0
0	$s_4$	1600	5	3	4	6	3	0	0	0	1
	$\Delta$	0	-300	-260	-320	-400	-450	0	0	0	0

Визначаємо напрямний стовпець -  $x_5$ , напрямний рядок -  $s_1$ , напрямний елемент = 18. Розраховуємо наступну симплекс-таблицю:

Таблиця 2.2

	C	-	300	260	320	400	450	0	0	0	M
	B	$x_0$	$x_1$	$x_2$	$x_3$	$x_4$	$x_5$	$s_1$	$s_2$	$s_3$	S
450	$x_5$	1000/9	5/6	10/9	2/3	7/9	1	1/18	0	0	0
0	$s_2$	800/3	-1/20	-1/30	-1/20	1/60	0	-1/60	1	0	0
0	$s_3$	7000/9	7/3	25/9	14/3	13/9	0	-1/9	0	1	0
0	$s_4$	3800/3	$\frac{1}{2}$	-1/3	2	11/3	0	-1/6	0	0	0
	$\Delta$	50000	75	240	-20	-50	0	25	0	0	0

Визначаємо напрямний стовпець -  $x_4$ , напрямний рядок -  $x_5$ , напрямний елемент = 7/9. Розраховуємо наступну симплекс-таблицю:

Таблиця 2.3

	C	-	300	260	320	400	450	0	0	0	M
	B	$x_0$	$x_1$	$x_2$	$x_3$	$x_4$	$x_5$	$s_1$	$s_2$	$s_3$	$a_2$
400	$x_4$	1000/7	15/14	10/7	6/7	1	9/7	1/14	0	0	0
0	$s_2$	1850/7	-19/280	-2/35	-9/140	0	-3/140	-1/56	1	0	0
0	$s_3$	4000/7	11/14	5/7	24/7	0	-13/7	-3/14	0	1	0

0	$s_4$	$5200/7$	$-10/7$	$-39/7$	$-8/7$	0	$-33/7$	$-3/7$	0	0	1
	$\Delta$	$400000/7$	$900/7$	$2180/7$	$160/7$	0	$450/7$	$200/7$	0	0	0

Отримали оптимальний розв'язок задачі:

$$x_1 = 0, x_2 = 0, x_3 = 0, x_4 = 1000/7, x_5 = 0, F_{min} = 400000/7.$$

### Висновки до другого розділу

В даному розділі було розв'язано задачу лінійного програмування методом штучного базису.

До переваг методу штучного базису належать: універсальність для будь-яких умов і наочність розв'язання. До недоліків – час на розрахунки.

		Васьківський В.			ДУ «Житомирська політехніка».23.121.03.000 - ПЗ	Арк.
		Локтікова Т.М.				12
Змн.	Арк.	№ докум.	Підпис	Дата		

### 3. РОЗРОБКА ПРОГРАМНОГО ПРОДУКТУ

#### 3.1. Розробка алгоритму розв’язання задачі МП

Перед розглядом самих алгоритмів роботи програми слід зазначити що проект буде реалізовано у вигляді десктоп-додатку. Тож перед початком роботи складемо загальну схему роботи додатку.

Спочатку програма має ініціалізувати графічний інтерфейс і відобразити його користувачу. Тепер користувач вводить дані задачі і запускає алгоритм обрахунку. Дані потрапляють до програми, де відбувається розв’язок самої задачі. Результат обрахувань задачі відправляються назад до користувача для формування сторінки з візуалізацією процесу обрахунку. Після завершення візуалізації отриманих даних робота загального алгоритму закінчується і може бути повторена спочатку.

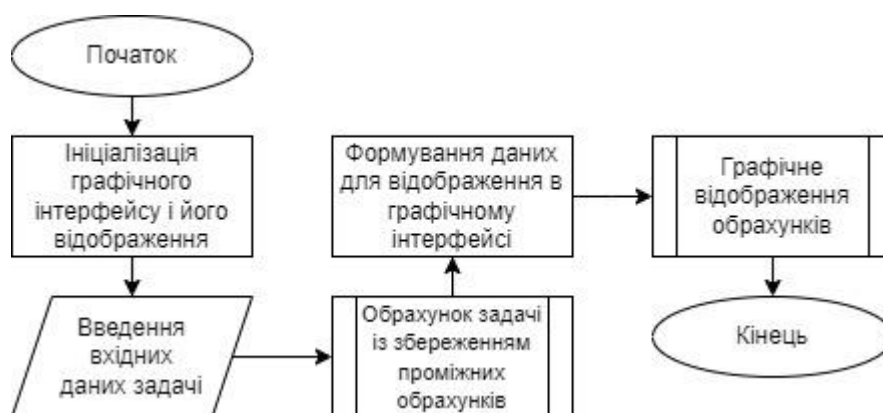


Рисунок 3.1 – загальна схема роботи додатку

В процесі роботи програми дані декілька разів передаються від користувача на сервер і навпаки. В той час як веб-сторінка лише отримує і візуально відображає дані, сервер займається обробкою цих даних, формуванням нових і обрахунком самої задачі.

Розглянемо детальніше сам алгоритм розв’язання задачі із збереженням проміжних результатів. Перш за все ми отримані дані від користувача ми валідуємо на коректність. Якщо дані коректні то починаємо процес обрахунку. Перш за все задача зводиться до пошуку мінімуму. Далі додаємо в задачу вільні змінні, а після них – штучні. Формуємо початкову симплекс-таблицю. Якщо

рядок оцінок вказує що розв’язок не доведений до кінця то на основі попередньої таблиці формуємо наступну, де повторюємо ці дії. Так триватиме доки не буде знайдено розв’язок або доведено що він відсутній. Крім того усі проміжні обрахунки записуються. Отриманий результат передаємо далі для відображення на веб-сторінці.



Рисунок 3.2 – схема роботи алгоритму розв’язання задачі

Окремою частиною програми є запис усіх проміжних кроків обрахунків. Фактично ці кроки будуть являти собою симплекс-таблиці з даними, що в них. Тому розглянемо детальніше процес створення симплекс-таблиці на основі минулої.

В якості вхідних даних буде минула симплекс-таблиця. Перед тим як почати формувати нову таблицю знайдемо напрямний рядок, напрямну колонку та напрямний елемент в минулій відповідно до алгоритму методу штучного базису. Тепер замінюємо в базисі змінну, що є напрямним рядком, на змінну, що є напрямним стовпцем. Обраховуємо значення елементів нового рядка, а після – і всі інші. Коли вже заповнили усі елементи таблиці, приступаємо до обрахунку рядку оцінок. Обрахунок кожного елемента таблиці відбувається циклічно по рядках в таблиці.

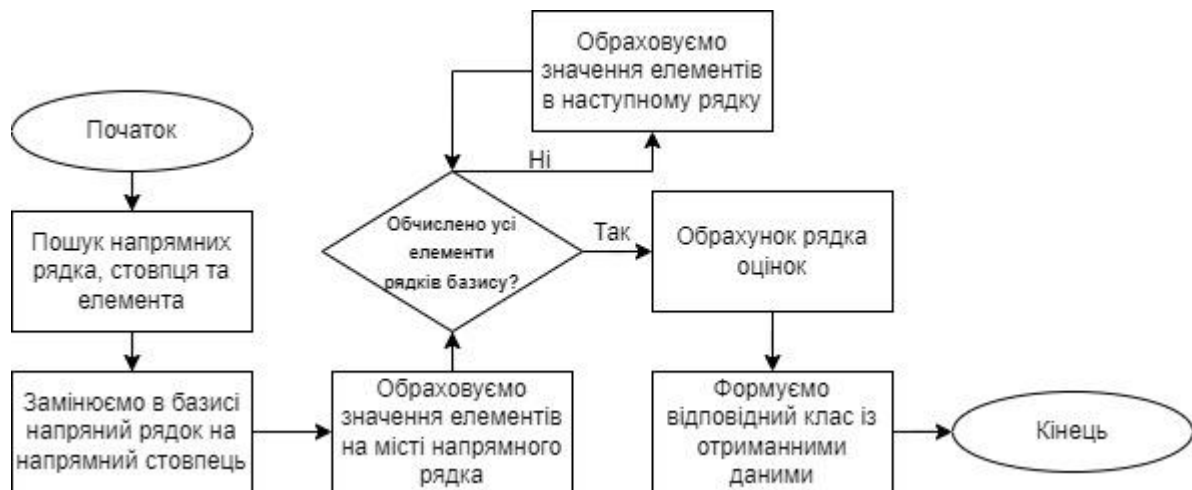


Рисунок 3.3 – схема роботи алгоритму обчислення наступної симплекс-таблиці

Розглянемо як само буде відбуватися графічне відображення результатів обрахунку.

Спочатку записуємо задачу вже з доданими вільними та штучними змінними до цільової функції та умов-обмежень. Виводимо першу вихідну симплекс-таблицю з умовами задачі. Перевіряємо чи дані з обрахунками містять ще симплекс-таблицю. Якщо вони є то графічно відображаємо і їх. Коли усі таблиці буде виведено, то виводимо фінальний результат обрахунків.

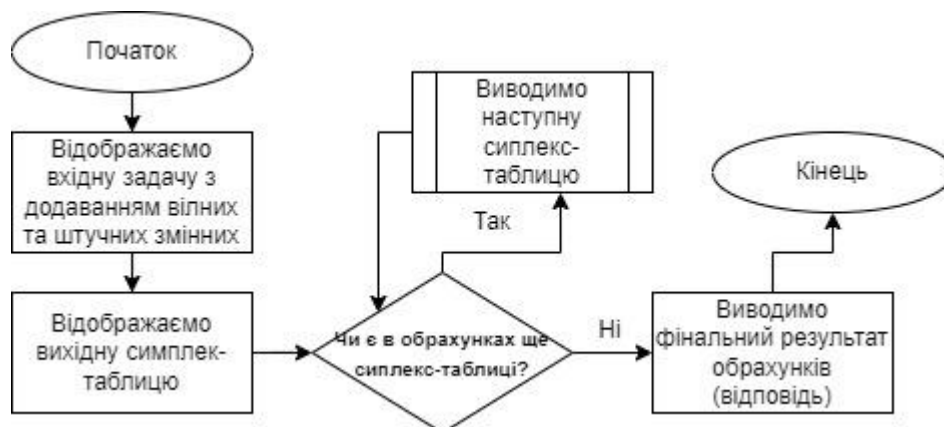


Рисунок 3.4 – схема алгоритму графічного відображення результатів обрахунку

### 3.2. Вибір інструментальних засобів для реалізації програми

Для реалізація програмного продукту необхідно обрати мову програмування, що буде використовуватися для створення додатку. Вибір проводився між наступними мовами та підсистемами: C#, C++, JavaScript, Python.

В таблиці 3.1 наведено порівняльну характеристику можливого інструментарію.

Таблиця 3.1

Критерій	C#	C++	JavaScript	Python
Тип мови	Статична	Статична	Динамічна	Динамічна
Використання	Загальнопризначена	Загальнопризначена	Веб-розробка, додатки	Загальнопризначена
Синтаксис	C-подібний	C-подібний	Java-подібний	Власний, особливий
Використання у веб-розробці	Так	Так	Так	Так
Використання в наукових обчисленнях	Так	Так	Так	Так
Підтримка функціонального програмування	Достатня	Достатня	Достатня	Достатня
Виконання математичних обчислень	Достатня	Достатня	Частково	Достатня
Наявність математичного функціоналу	Достатня	Достатня	Низька	Достатня
Підтримка засобів візуалізації даних	Достатня	Середня	Достатня	Достатня



Під час аналізу та визначення оптимального інструментарію вибір зупинився на мові програмування C# та Windows Forms як додаток, який відповідає за графічний інтерфейс користувача. Дані інструменти розробки надають можливість створити десктопний додаток зі зручним створенням інтуїтивного дизайном та гарним функціоналом.

Тож, перейдемо до безпосереднього створення додатку.

### 3.3. Написання програмного продукту

Основними елементами нашої форми є DataGridView (для відображення обраної симплекс таблиці) та TabControl (для перемиканням між симплекс-таблицями).

Основний клас програми – Simplex. Він відповідає безпосередньо за вирішення задачі.

На рисунку 3.4 зображено метод Calculate, що відповідає за розрахунок симплекс-таблиць та повертає симплекс-таблиці та результат обчислень.

```
public List<Fraction[,]> Calculate(Fraction[] result)
{
    int mainCol, mainRow;
    while (!IsEnd())
    {
        mainCol = FindMainColumn();
        mainRow = FindMainRow(mainCol);
        basis[mainRow] = mainCol;

        var newTable = new Fraction[m, n];

        for (int j = 0; j < n; j++)
            newTable[mainRow, j] = (simplexTable[mainRow, j] /
            simplexTable[mainRow, mainCol]).Abbreviation();

        for (int i = 0; i < m; i++)
        {
            if (i == mainRow)
                continue;

            for (int j = 0; j < n; j++)
                newTable[i, j] = (simplexTable[i, j] - simplexTable[i,
            mainCol] * newTable[mainRow, j]).Abbreviation();
        }
        tables.Add(newTable);
        simplexTable = newTable;
    }

    for (int i = 0; i < result.Length; i++)
    {
```

		Васьківський В.			ДУ «Житомирська політехніка».23.121.03.000 - ПЗ	Арк.
		Локтікова Т.М.				17
Змн.	Арк.	№ докум.	Підпис	Дата		

```

        int k = basis.IndexOf(i + 1);
        if (k != -1)
            result[i] = simplexTable[k, 0].Abbreviation();
        else
            result[i] = new Fraction(0);
    }
    return tables;
}

```

Рисунок- 3.4 Код програми (метод Calculate класу Simplex)

На рисунку 3.5 зображено метод IsEnd, що відповідає за перевірку чи є наступні ітерації.

```

private bool IsEnd()
{
    for (int j = 1; j < n; j++)
        if (simplexTable[m - 1, j] < new Fraction(0))
            return false;

    return true;
}

```

Рисунок- 3.5 Код програми (метод IsEnd класу Simplex)

На рисунку 3.6 зображено методи FindMainColumn та FindMainRow, що відповідає пошук головної колонки та головного рядка відповідно.

```

private int FindMainColumn()
{
    int mainColumn = 1;

    for (int j = 2; j < n; j++)
        if (simplexTable[m - 1, j] < simplexTable[m - 1, mainColumn])
            mainColumn = j;

    return mainColumn;
}

private int FindMainRow(int mainColumn)
{
    int mainRow = 0;

    for (int i = 0; i < m - 1; i++)
        if (simplexTable[i, mainColumn] > new Fraction(0))
        {
            mainRow = i;
            break;
        }

    for (int i = mainRow + 1; i < m - 1; i++)
        if (simplexTable[i, mainColumn] > new Fraction(0) && (simplexTable[i, 0] / simplexTable[i, mainColumn] < simplexTable[mainRow, 0] / simplexTable[mainRow, mainColumn]))
            mainRow = i;

    return mainRow;
}

```

		Васьківський В.			ДУ «Житомирська політехніка».23.121.03.000 - ПЗ	Арк.
		Локтікова Т.М.				18
Змн.	Арк.	№ докум.	Підпис	Дата		

Рисунок- 3.6 Код програми (методи FindMainColumn та FindMainRow класу Simplex)

### 3.4. Тестування програмного продукту

Можемо перейти до перевірки коректності програми на прикладі заданої задачі.

Для початку введення даних користувач повинен ввести кількість невідомих і умов-обмежень.

На рисунку 3.7 зображено форму формування задачі.

Рисунок 3.7- Форма формування задачі

Далі користувач повинен заповнити вихідну математичну модель у відповідному вікні.

На рисунку 3.8 зображено форму заповнення математичної моделі.

2000	15	20	12	14	18
30000	20	30	15	25	30
1000	4	5	6	3	2
1600	4	3	4	6	3
-	300	260	320	400	450

Рисунок 3.8- Форма заповнення математичної моделі

Після натиснення кнопки “Розрахувати” користувач отримає всі симплекс-таблиці і значення цільової функції у разі вдалого розв’язання задачі і повідомлення, якщо щось пішло не так.

На рисунку 3.9 зображено проміжну симплекс-таблицю №1.

Метод симплекс-таблиць

— □ ×

Головна [Допомога](#)

Розрахунки

Проміжна №1

Вихідна

1000/9	5/6	10/9	2/3	7/9	1	1/18	0	0	0		
80000/3	-5	-10/3	-5	5/3	0	-5/3	1	0	0		
7000/9	7/3	25/9	14/3	13/9	0	-1/9	0	1	0		
3800/3	3/2	-1/3	2	11/3	0	-1/6	0	0	1		
50000	75	240	-20	-50	0	25	0	0	0		

$$F_{\max} = 400000/7$$

$$x_1 = 0$$

$$x_2 = 0$$

$$x_3 = 0$$

$$x_4 = 1000/7$$

$$x_5 = 0$$

Рисунок 3.9 - Проміжна симплекс-таблицю №1

На рисунку 3.10 зображено результуючу симплекс-таблицю.

## Розрахунки

Проміжна №1	Вихідна										
1000/7	15/14	10/7	6/7	1	9/7	1/14	0	0	0		
185000/7	-95/14	-40/7	-45/7	0	-15/7	-25/14	1	0	0		
4000/7	11/14	5/7	24/7	0	-13/7	-3/14	0	1	0		
5200/7	-17/7	-39/7	-8/7	0	-33/7	-3/7	0	0	1		
400000/7	900/7	2180/7	160/7	0	450/7	200/7	0	0	0		

$$F_{\max} = 400000/7$$

$$x_1 = 0$$

$$x_2 = 0$$

$$x_3 = 0$$

$$x_4 = 1000/7$$

$$x_5 = 0$$

Рисунок 3.10 - Результуюча симплекс-таблицю

Після п'яти тестувань програма показала, результати, які зведені до таблиці 3.2.

**Коректність виконання програми**

Функції	Введення кількості змінних та умов-обмежень	Введення умови задачі	Отримання всіх проміжних симплекс-таблиць	Отримання відповіді
1 спроба	+	+	+	+
2 спроба	+	+	+	+
3 спроба	+	+	+	+
4 спроба	+	+	+	+
5 спроба	+	+	+	+

**Висновки до третього розділу**

В даному розділі були розроблені алгоритми програмного додатку та побудовано відповідні блок-схеми. Було проаналізовано можливий інструментарій для створення додатку необхідного функціоналу. В результаті чого було створено програмний продукт для розв'язання задач лінійного програмування методом штучного базису з використанням мови програмування C# і графічного відображення WinForms.

Було протестовано розроблений додаток на прикладі заданої задачі. Всі результати, отримані за допомогою програми, співпали з результатами графічного методу і методу штучного базису.

## ВИСНОВКИ

Отже, під час виконання даного курсового проекту було закріплено вивчений матеріал, який охоплював розв'язання задач лінійного програмування методом штучного базису та графічним методом. Виконання курсового проекту було розділено на три етапи.

У першому етапі була поставлена задача лінійного програмування з лінійною цільовою функцією та обмежувальними умовами. Згідно завдання була побудована математична модель задачі, і було надано достатньо інформації для розв'язання поставленої задачі.

У другому етапі була розв'язана задача лінійного програмування двома методом штучного базису.

У третьому етапі були розроблені алгоритми програмного додатку та побудовано відповідні блок-схеми. Було проаналізовано можливі інструменти для створення додатку з необхідним функціоналом. В результаті був створений програмний продукт для розв'язання задач лінійного програмування методом штучного базису з використанням мови програмування C# та графічного інтерфейсу WinForms. Розроблений додаток був протестований на прикладі задачі, і всі отримані результати збіглися з результатами графічного методу та методу штучного базису.

Отже, виконана робота показала, що з використанням вивченого матеріалу за курс можна успішно розв'язувати задачі лінійного програмування та створювати програмні продукти на основі вивчених алгоритмів.

Розроблений додаток повністю задовольняє вимоги, і його використання для користувача не повинно становити складностей.

		Васьківський В.			ДУ «Житомирська політехніка».23.121.03.000 - ПЗ	Арк.
		Локтікова Т.М.				23
Змн.	Арк.	№ докум.	Підпис	Дата		

## СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Документація з мови програмування C# [Електронний ресурс] – Режим доступу до ресурсу: <https://docs.microsoft.com/uk-ua/dotnet/csharp/>.
2. Математичні методи дослідження операцій [Електронний ресурс] / Освітній портал ДУ «Житомирська політехніка» – 2022. – Режим доступу до ресурсу: <https://learn.ztu.edu.ua/course/view.php?id=1348>
3. Симплекс-метод [Електронний ресурс] – Режим доступу до ресурсу: <https://uk.wikipedia.org/wiki/%D0%A1%D0%B8%D0%BC%D0%BF%D0%BB%D0%B5%D0%BA%D1%81-%D0%BC%D0%B5%D1%82%D0%BE%D0%B4>.
4. Метод штучного базису [Електронний ресурс] – Режим доступу до ресурсу: [https://wiki.cuspu.edu.ua/index.php/%D0%9C%D0%B5%D1%82%D0%BE%D0%B4\\_%D1%88%D1%82%D1%83%D1%87%D0%BD%D0%BE%D0%B3%D0%BE\\_%D0%B1%D0%B0%D0%B7%D0%B8%D1%81%D1%83](https://wiki.cuspu.edu.ua/index.php/%D0%9C%D0%B5%D1%82%D0%BE%D0%B4_%D1%88%D1%82%D1%83%D1%87%D0%BD%D0%BE%D0%B3%D0%BE_%D0%B1%D0%B0%D0%B7%D0%B8%D1%81%D1%83).
5. Графічний метод розв’язування задач лінійного програмування [Електронний ресурс] – Режим доступу до ресурсу: <https://fingal.com.ua/content/view/449/76/1/1/>.
6. Математичне програмування: Навчальний посібник / За загальною редакцією В.М. Михайленка. – 3-е видання, виправлене і доповнене. – К.: Вид-во Європ. ун-ту, 2007. – 497 с.
7. Windows Forms [Електронний ресурс] – Режим доступу до ресурсу: [https://uk.wikipedia.org/wiki/Windows\\_Forms](https://uk.wikipedia.org/wiki/Windows_Forms).

		Васьківський В.			ДУ «Житомирська політехніка».23.121.03.000 - ПЗ	Арк.
		Локтікова Т.М.				24
Змн.	Арк.	№ докум.	Підпис	Дата		



# ДОДАТКИ

		Васьківський В.			ДУ «Житомирська політехніка».23.121.03.000 - ПЗ	Арк.
		Локтікова Т.М.				25
Змн.	Арк.	№ докум.	Підпис	Дата		

## ЛІСТИНГ класу Simplex:

```

using System;
using System.Collections.Generic;

namespace WindowsFormsApp1
{
    internal class Simplex
    {
        Fraction[,] simplexTable;
        int m, n;
        List<Fraction[,]> tables = new List<Fraction[,]>();
        List<int> basis;

        public Simplex(Fraction[,] source)
        {
            m = source.GetLength(0);
            n = source.GetLength(1);
            simplexTable = new Fraction[m, n + m - 1];
            basis = new List<int>();

            for (int i = 0; i < m; i++)
            {
                for (int j = 0; j < simplexTable.GetLength(1); j++)
                {
                    if (j < n)
                        simplexTable[i, j] = source[i, j];
                    else
                        simplexTable[i, j] = new Fraction(0);
                }

                if ((n + i) < simplexTable.GetLength(1))
                {
                    simplexTable[i, n + i] = new Fraction(1);
                    basis.Add(n + i);
                }
            }

            n = simplexTable.GetLength(1);

            public List<Fraction[,]> Calculate(Fraction[] result)
            {
                int mainCol, mainRow;
                while (!IsEnd())
                {
                    mainCol = FindMainColumn();
                    mainRow = FindMainRow(mainCol);
                    basis[mainRow] = mainCol;

                    var newTable = new Fraction[m, n];

                    for (int j = 0; j < n; j++)
                        newTable[mainRow, j] = (simplexTable[mainRow, j] /
simplexTable[mainRow, mainCol]).Abbreviation();

                    for (int i = 0; i < m; i++)
                    {
                        if (i == mainRow)
                            continue;

                        for (int j = 0; j < n; j++)

```

		Васьківський В.			ДУ «Житомирська політехніка».23.121.03.000 - ПЗ	Арк.
		Локтікова Т.М.				26
Змн.	Арк.	№ докум.	Підпис	Дата		

```

        newTable[i, j] = (simplexTable[i, j] - simplexTable[i,
mainCol] * newTable[mainRow, j]).Abbreviation();
    }
    tables.Add(newTable);
    simplexTable = newTable;
}

for (int i = 0; i < result.Length; i++)
{
    int k = basis.IndexOf(i + 1);
    if (k != -1)
        result[i] = simplexTable[k, 0].Abbreviation();
    else
        result[i] = new Fraction(0);
}
return tables;
}

public Fraction GetF()
{
    var f = simplexTable[simplexTable.GetLength(0) - 1, 0].Abbreviation();
    if ((double)f != 0)
        return f;

    throw new Exception("Сталася помилка під час розрахунку. Перевірте, будь
ласка, введені дані");
}

private bool IsEnd()
{
    for (int j = 1; j < n; j++)
        if (simplexTable[m - 1, j] < new Fraction(0))
            return false;

    return true;
}

private int FindMainColumn()
{
    int mainColumn = 1;

    for (int j = 2; j < n; j++)
        if (simplexTable[m - 1, j] < simplexTable[m - 1, mainColumn])
            mainColumn = j;

    return mainColumn;
}

private int FindMainRow(int mainColumn)
{
    int mainRow = 0;

    for (int i = 0; i < m - 1; i++)
        if (simplexTable[i, mainColumn] > new Fraction(0))
        {
            mainRow = i;
            break;
        }

    for (int i = mainRow + 1; i < m - 1; i++)
        if (simplexTable[i, mainColumn] > new Fraction(0) && (simplexTable[i,
0] / simplexTable[i, mainColumn] < simplexTable[mainRow, 0] / simplexTable[mainRow,
mainColumn]))
            mainRow = i;
}

```

		Васьківський В.			ДУ «Житомирська політехніка».23.121.03.000 - ПЗ	Арк.
		Локтікова Т.М.				27
Змн.	Арк.	№ докум.	Підпис	Дата		

```

        return mainRow;
    }
}

```

## Лістинг класу форми Result:

```

using System;
using System.Windows.Forms;

namespace WindowsFormsApp1
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();

            private void Form1_Load(object sender, EventArgs e)
            {
                comboBoxType.SelectedIndex = 0;
                dataGridViewInputMatrix.RowCount = 4;
                dataGridViewInputMatrix.ColumnCount = 3;
                dataGridViewInputMatrix[0, 3].Value = "-";
                dataGridViewInputMatrix[0, 3].ReadOnly = true;
            }

            private void numericUpDownRows_ValueChanged(object sender, EventArgs e)
            {
                var countRows = Convert.ToInt32(numericUpDownRows.Value);
                dataGridViewInputMatrix.RowCount = countRows;
                if (countRows > 1)
                {
                    dataGridViewInputMatrix[0, dataGridViewInputMatrix.RowCount - 2].Value
= null;
                    dataGridViewInputMatrix[0, dataGridViewInputMatrix.RowCount -
2].ReadOnly = false;
                }
                dataGridViewInputMatrix[0, dataGridViewInputMatrix.RowCount - 1].Value =
"-";
                dataGridViewInputMatrix[0, dataGridViewInputMatrix.RowCount - 1].ReadOnly
= true;
            }

            private void numericUpDownColumns_ValueChanged(object sender, EventArgs e)
            {
                var countColumns = Convert.ToInt32(numericUpDownColumns.Value);
                dataGridViewInputMatrix.ColumnCount = countColumns;
            }

            private void buttonCalculate_Click(object sender, EventArgs e)
            {
                var countRows = Convert.ToInt32(numericUpDownRows.Value);
                var countColumns = Convert.ToInt32(numericUpDownColumns.Value);
                var source = new Fraction[countRows, countColumns];
                for (int i = 0; i < countRows; i++)
                {
                    for (int j = 0; j < countColumns; j++)
                    {
                        if (dataGridViewInputMatrix.Rows[i].Cells[j].Value.ToString() ==
"-")

```

		Васьківський В.			ДУ «Житомирська політехніка».23.121.03.000 - ПЗ	Арк.
		Локтікова Т.М.				28
Змн.	Арк.	№ докум.	Підпис	Дата		

```

        {
            source[i, j] = new Fraction(0);
            continue;
        }
        if (i == countRows - 1)
        {
            source[i, j] = new
Fraction(Convert.ToInt64(dataGridViewInputMatrix.Rows[i].Cells[j].Value) * -1);
            continue;
        }
        source[i, j] = new
Fraction(Convert.ToInt64(dataGridViewInputMatrix.Rows[i].Cells[j].Value));
    }
}

new Result(source, comboBoxType.Text).Show();
}

private void dataGridViewInputMatrix_EditingControlShowing(object sender,
DataGridViewEditingControlShowingEventArgs e)
{
    e.Control.KeyPress -= new KeyPressEventHandler(Column1_KeyPress);

    TextBox tb = e.Control as TextBox;
    if (tb != null)
    {
        tb.KeyPress += new KeyPressEventHandler(Column1_KeyPress);
    }
}

private void Column1_KeyPress(object sender, KeyPressEventArgs e)
{
    if (!char.IsControl(e.KeyChar) && !char.IsDigit(e.KeyChar) && e.KeyChar !=
'-')
    {
        e.Handled = true;
    }
}
}
}

```

Лістинг класу форми Result:

```

using System;
using System.Collections.Generic;
using System.Drawing;
using System.Windows.Forms;

namespace WindowsFormsApp1
{
    public partial class Result : Form
    {
        Fraction[,] source;
        int countRows;
        int countColumns;
        string action;

        public Result(Fraction[,] source, string action)
        {
            InitializeComponent();

```

		Васьківський В.			ДУ «Житомирська політехніка».23.121.03.000 - ПЗ	Арк.
		Локтікова Т.М.				29
Змн.	Арк.	№ докум.	Підпис	Дата		

```

        this.source = source;
        this.countRows = source.GetLength(0);
        this.countColumns = source.GetLength(1);
        this.action = action;
        Calculate();
    }

    private void Calculate()
    {
        try
        {
            var result = new Fraction[source.GetLength(1) - 1];
            List<Fraction[,]> tablesResult;

            var simplex = new Simplex(source);
            tablesResult = simplex.Calculate(result);

            for (int i = 0; i < result.Length; i++)
            {
                var labelResult = new Label();
                labelResult.Text = $"x{i + 1} = {result[i]}";
                labelResult.Font = new Font("Microsoft Sans Serif", 14);
                labelResult.Location = new Point(18, i * 30 + 330);
                labelResult.Width = 125;
                tabPageMain.Controls.Add(labelResult);
            }

            labelResultF.Text = $"F{action} = {simplex.GetF()}";
            var tabPage = new TabPage[tablesResult.Count];
            var gridViewTables = new DataGridView[tablesResult.Count];

            for (int i = 0; i < tablesResult.Count; i++)
            {
                string title;
                if (i == tablesResult.Count - 1)
                {
                    title = "Вихідна";
                }
                else
                {
                    title = $"Проміжна №{i + 1}";
                }

                tabPage[i] = new TabPage(title);
                tabPageMain.TabPages.Add(tabPage[i]);
                gridViewTables[i] = GetStyleDataGridView();

                for (int j = 0; j < countRows; j++)
                {
                    for (int k = 0; k < countColumns * 2; k++)
                    {
                        if (i == tablesResult.Count - 1 && k == 0)
                        {
                            foreach (var res in result)
                            {
                                if (tablesResult[i][j, k] == res)
                                {
                                    gridViewTables[i].Rows[j].Cells[k].Style.BackColor = Color.Blue;
                                    break;
                                }
                            }
                        }
                    }
                }
                try
                {
                    gridViewTables[i].Rows[j].Cells[k].Value =

```

		Васьківський В.			ДУ «Житомирська політехніка».23.121.03.000 - ПЗ	Арк.
		Локтікова Т.М.				30
Змн.	Арк.	№ докум.	Підпис	Дата		

```

tablesResult[i][j, k];
        }
        catch { }
    }
}
if (i == tablesResult.Count - 1)
    gridViewTables[i].Rows[countRows -

1].Cells[0].Style.BackColor = Color.Yellow;

        tabPage[i].Controls.Add(gridViewTables[i]);
        gridViewTables[i].ClearSelection();
    }
}
catch (NullReferenceException ex)
{
    MessageBox.Show("Введено не всі значення", "Сталася помилка",
    MessageBoxButtons.OK, MessageBoxIcon.Error);
}
catch (Exception ex)
{
    MessageBox.Show(ex.Message, "Сталася помилка", MessageBoxButtons.OK,
    MessageBoxIcon.Error);
}
}

private DataGridView GetStyleDataGridView()
{
    var dataGridView = new DataGridView();
    dataGridView.ReadOnly = true;
    dataGridView.ColumnCount = countColumns * 2;
    dataGridView.RowCount = countRows;
    dataGridView.Size = new Size(345, 170);
    dataGridView.AllowUserToResizeRows = true;
    dataGridView.AllowUserToResizeColumns = true;
    dataGridView.AutoSizeColumns();
    dataGridView.ColumnHeadersVisible = false;
    dataGridView.RowHeadersVisible = false;
    dataGridView.AutoSizeColumnsMode = DataGridViewAutoSizeColumnsMode.Fill;
    dataGridView.RowsDefaultCellStyle.Alignment =
    DataGridViewContentAlignment.MiddleCenter;
    dataGridView.MultiSelect = false;
    return dataGridView;
}
}
}

```