

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

**НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ  
«ХАРКІВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ»**

Інститут Комп'ютерних наук та інформаційних технологій

Кафедра Програмної інженерії та інтелектуальних технологій управління

Спеціальність 122 Комп'ютерні науки

Освітня програма Комп'ютерні науки та інтелектуальні системи

**ЛАБОРАТОРНА РОБОТА №5 за курсом**

**«ВІ-технології»**

Тема лабораторної роботи Аналітично-Ієрархічний процес прийняття й  
обґрунтування рішень

Виконав студент 5 курсу, групи КН-М422

Захар ПАРАХІН

(підпис, прізвище та ініціали)

Перевірів доцент, кандидат технічних наук

Олег ГУЦА

(підпис, прізвище та ініціали)

Харків 2022

## Мета роботи:

Практичне підтвердження ефективності аналітично-ієрархічного процесу прийняття й обґрунтування рішень та придбання практичних навичок та придбання навичок роботи з табличним процесором Microsoft Excel для рішення задач прийняття й обґрунтування рішень.

## Хід роботи

### Завдання 1.

Реалізувати в середовищі табличного процесора Microsoft Excel приведений нижче приклад аналітично-ієрархічного процесу прийняття й обґрунтування рішень. Нехай дівчині необхідно вибрати чоловіка серед одного з трьох молодих людей. Критерії, якими керується дівчина, наступні: вік, зовнішність, інтелект і багатство. Потрібно розглянути усі варіанти і вибрати найкращий. Використовується шкала відносної важливості у таб. 1.1

Таблиця 1.1 - Шкала відносної важливості

Найменування	Кількісне значення
набагато краще	9
значне краще	7
краще	5
ледве краще	3
рівні	1
ледве гірше	1/3
гірше	1/5
значно гірше	1/7
набагато гірше	1/9

Також за кожним критерієм необхідно обчислити власний вектор матриці порівнянь. Значення компонентів власного вектора обчислюються в такий спосіб: витягається корінь  $n$ -й ступеня ( $n$  - розмірність матриці порівнянь) з добутків елементів кожного рядка. А також вагу, тобто відносну частку значення вектора по відношенню суми власних векторів всіх альтернатив. Сформуємо відносну важливість критеріїв для ОНР рис.1.1

	Возраст	Внешность	Интеллект	Богатство	Вектор	Вес
Возраст	1	1/3	1/5	1/9	0,2933706	0,0421488
Внешность	3	1	1/3	1/9	0,5773503	0,0829485
Интеллект	5	3	1	1/7	1,2098967	0,1738271
Богатство	9	9	7	1	4,8797297	0,7010756
				Итого:	6,9603473	1

Рисунок 1.1 - Відносна важливість критеріїв для ОНР

Маємо критеріальний опис альтернатив для задачі (рис. 1.2)

	возраст	внешность	интеллект	богатство
Кирилл	молодой	красивый	умный	бедный
Влад	средний	обычная	средний	среднее
Артем	старый	некрасивый	глупый	богатый

Рисунок 1.2 - Критеріальний опис альтернатив

За кожним варіантом отримані результати порівняння ОНР кожної пари альтернатив і вектору і ваги щодо критерію “Вік” (рис. 1.3), за критерієм “Зовнішність” (рис. 1.4), за критерієм “Інтелект” (рис.1.5), за критерієм “Багатство”(рис. 1.6)

Возраст	Кирилл	Влад	Артем	Вектор	Вес
Кирилл	1	1/5	7	1,1187	0,2573
Влад	5	1	5	2,924	0,6724
Артем	1/7	1/5	1	0,3057	0,0703

Рисунок 1.1 - Результати порівняння за критерієм “Вік”

Внешность	Кирилл	Влад	Артем	Вектор	Вес
Кирилл	1	7	9	3,9791	0,772
Влад	1/7	1	5	0,8939	0,1734
Артем	1/9	1/5	1	0,2811	0,0545

Рисунок 1.2 - Результати порівняння за критерієм “Зовнішність”

<b>Интеллект</b>	Кирилл	Влад	Артем	<b>Вектор</b>	<b>Вес</b>
Кирилл	1	7	9	3,9791	0,772
Влад	1/7	1	5	0,8939	0,1734
Артем	1/9	1/5	1	0,2811	0,0545

Рисунок 1.3 - Результати порівняння за критерієм “Інтелект”

<b>Богатство</b>	Кирилл	Влад	Артем	<b>Вектор</b>	<b>Вес</b>
Кирилл	1	1/7	1/9	0,25132	0,04805
Влад	7	1	1/7	1	0,19119
Артем	9	7	1	3,97906	0,76076

Рисунок 1.4 - Результати порівняння за критерієм “Багатство”

Підсумок для кожного нареченого підводиться дуже просто: спочатку оцінка, отримана нареченим за визначеним критерієм, збільшується на вагу цього критерію, потім отримані добутки складаються (рис. 1.7). За цим підсумком обираємо найбільш цікавого для дівчини (за більшим вектором) людину.

	<b>возраст</b>	<b>внешность</b>	<b>интеллект</b>	<b>богатство</b>	<b>Вектор</b>
<b>Кирилл</b>	молодой	красивый	умный	бедный	0,242764675
<b>Влад</b>	средний	обычная	средний	среднее	0,206915504
<b>Артем</b>	старый	некрасивый	глупый	богатый	0,550319821

Рисунок 1.7 - Підсумок альтернатив

Як видно з результатів ключовим критерієм виступило багатство на комбінації інших критеріїв. Цей алгоритм прийняття і обґрунтування рішень добре підходить для пояснення за кожним критерієм і оцінкою його важливості і тим самим дозволяє краще описувати важливість того чи іншого критерію і дає загальну оцінку альтернативам, де обираємо з ліпшим власним вектором.

Загалом цей алгоритм дозволяє побудувати ієрархію за якою потім вибрати кращу альтернативу з існуючих і підходить більше для важливих завдань з економіки як підбір контрактів у системі яка може включати модель попиту-пропозиції як наприклад задачі з вибору для логістики за допомогою багато-агентної системи, де цей алгоритм може ефективно і реалістично працювати.

## 2 Завдання

Реалізувати в середовищі табличного процесора Microsoft Excel аналітично-ієрархічний процес прийняття й обґрунтування рішень у власній предметній області для п'яти критеріїв і чотирьох альтернатив.

Предметною областю була обрана галузь мов програмування, а завдання ставиться по вибору мови програмування для роботи в ІТ-сфері. Оскільки ця задача пов'язана на пряму з вибором мови для розробника головною метою є підбір найбільш вдалої мови з балансом між конкуренцією та потенційною користю (прибутком і вільним часом). Через це критерії базуються на ринкових умовах.

Критеріїв усього п'ять:

- популярність мови - скільки людей її вивчають та скільки фахівців по ній зараз на ринку (як наприклад з C# є велика кількість людей і роботу знайти складніше, бо велика пропозиція);
- рівень зарплат - одним з основних критеріїв включає оцінку діапазона зарплат і мінімальну ставку на ринку;
- фреймворки - оцінка популярності серед замовників фреймів мови та їх використання для розуміння довгостроковості обставин ринку;
- попит - кількість вакансій і попиту на рішення (чим більше тим краще на відміну від популярності);
- складність - відносна суб'єктивна оцінка ОПР наскільки тяжко вивчати технології мови та стек пов'язаний з нею і оцінка того наскільки довго займає вирішення типових завдань на цій мові. Краща якщо мова легше.

Відношення ОПР до критеріїв буде базуватися на 9-бальній шкалі рис. 2.1

наименование	количественное значение
<b>гораздо лучше</b>	<b>9</b>
<b>значительно лучше</b>	<b>7</b>
<b>лучше</b>	<b>5</b>
<b>чуть лучше</b>	<b>3</b>
<b>равны</b>	<b>1</b>
<b>чуть хуже</b>	<b>1/3</b>
<b>хуже</b>	<b>1/5</b>
<b>значительно хуже</b>	<b>1/7</b>
<b>гораздо хуже</b>	<b>1/9</b>

Рисунок 2.1 - шкала порівняння ОНР

На базі цього сформована відносна важливість критеріїв для ОНР, що зображено матрицею на рис. 2.2. Нормалізований вектор в стовпці “Вага”.

	Популярність	Рівень зарплат	Фреймворки	Попит	Складність	Вектор	Вес
Популярність	1	1/9	1/5	1/7	1/3	0,25	0,033306985
Рівень зарплат	9	1	3	5	7	3,94	0,516069331
Фреймворки	5	1/3	1	1/5	5	1,11	0,145208299
Попит	7	1/5	5	1	3	1,84	0,241026918
Складність	3	1/7	1/5	1/3	1	0,49	0,064388466
					Итого:	7,62743	1

Рисунок 2.2 - Відносна важливість критеріїв

Відносно порівнянь можна вказати, що вони сформовані на доступній інформації з різномовних сайтів і власного досвіду, тому не включають можливість того, що ОНР не знає наприклад польської чи англійської. Були обрані такі альтернативи як Ruby, C#, Java, JavaScript (JS), які все таки живуть на ринку і підібрано під Web-development (FullStack) через причини збільшення інтеграції вимог до Front-End і Back-End і як наслідок обрізається частина даних для оцінки ( без мобільної розробки чи гейм-деву).

Наступний крок це проведення порівнянь відповідно до сформованих критерії для альтернатив, що зображено на рис.2.3-2.7.

<b>Популярність</b>	Ruby	C#	JS	Java	<b>Вектор</b>	<b>Вес</b>
Ruby	1	1/5	1/9	1/7	0,24	0,036072
C#	5	1	1/7	1/3	0,70	0,106155
JS	9	7	1	5	4,21	0,64022
Java	7	3	1/5	1	1,43	0,217553
					6,5803374	1

Рисунок 2.3 - Матриця порівняння за критерієм “Популярність”

<b>Попит</b>	Ruby	C#	JS	Java	<b>Вектор</b>	<b>Вес</b>
Ruby	1	3	1/9	1/3	0,58	0,087747
C#	1/3	1	1/7	1/3	0,35	0,053946
JS	9	7	1	7	4,58	0,69647
Java	3	3	1/7	1	1,06	0,161837
					6,5797184	1

Рисунок 2.4 - Матриця порівняння за критерієм “Попит”

<b>Фреймворки</b>	Ruby	C#	JS	Java	<b>Вектор</b>	<b>Вес</b>
Ruby	1	5	1/3	5	1,70	0,302877
C#	1/5	1	1/5	1/5	0,30	0,053313
JS	3	5	1	5	2,94	0,524598
Java	1/5	5	1/5	1	0,67	0,119212
					5,6096853	1

Рисунок 2.5 - Матриця порівняння за критерієм “Фреймворки”

<b>Рівень зарплат</b>	Ruby	C#	JS	Java	<b>Вектор</b>	<b>Вес</b>
Ruby	1	7	9	5	4,21	0,636145
C#	1/7	1	7	1/3	0,76	0,114736
JS	1/9	1/7	1	1/7	0,22	0,032951
Java	1/5	3	7	1	1,43	0,216168
					6,6224886	1

Рисунок 2.6 - Матриця порівняння за критерієм “Рівень зарплат”

Складність	Ruby	C#	JS	Java	Вектор	Вес
Ruby	1	9	3	7	3,71	0,572811
C#	1/9	1	1/7	1/9	0,20	0,031659
JS	1/3	7	1	5	1,85	0,285518
Java	1/7	9	1/5	1	0,71	0,110012
					6,4729742	1

Рисунок 2.7 - Матриця порівняння за критерієм “Складність”

Наступним є вирахування для кожної альтернативи власного вектора базуючись сумі їхніх вагах по кожному критерію (рис. 2.3-2.7) помножених на вагу відповідного критерія з рис.2.2. Це робиться для визначення сумісної пари як відповідь на запитання яка мова програмування підійде ОПР. Отримали результат, який зображено на рис. 2.8

	Популярність	Рівень зарплат	Фреймворки	Попит	Складність	
	Популярність	Рівень зарплат	Фреймворки	Попит	Складність	Вектор
<b>Ruby</b>	ниже середньої	високий	багато і вживані	середній	низька	0,431508674
<b>C#</b>	середня-	середній-	вживані	середній	висока	0,08552964
<b>JS</b>	висока	низький	багато	середній	середня-	0,300756875
<b>Java</b>	середня+	середній+	вживані+	високий	висока	0,182204811
						1

Рисунок 2.8 - Результат обчислення власних векторів альтернатив

Проаналізуємо отримані результати таким чином, найкращою альтернативою виступає мова Ruby в цій предметній області для Веб-розробки і для обґрунтування можемо розглянути результати по критеріям, тобто за порівнянням критеріїв (рис. 2.2) найменш вагомим критерієм виступає “Популярність”, як означено ще з опису критеріїв, але не має нульового значення, бо для навчання потрібен хоча б якийсь рівень використання мови, потім найважливішим критерієм є “Рівень зарплат”, так як метою було знайти мову програмування, яка буде забезпечувати вигідний баланс між зарплатою і вільним часом, та критерії пов’язані з цим як “Попит”, який впливає на сьогодишню ситуацію і критерій “Фреймворки”. Тим часом складність краща чим нижча, є просто фактором комфорту використання мови. За порівнянням за самими



критеріями кожної альтернативи, то Ruby переважає значно за рівнем зарплат інші, також має достатню вагу за “Фреймворками” і одночасно з цим популярність нижче попиту, що вказує на нижчу конкуренцію.

### **Висновки:**

Було отримано практичні навички з роботи з табличним процесором Microsoft Excel для рішення задач прийняття і обґрунтування рішень. Також отримані знання з аналітично-ієрархічного процесу на вирішенні задачі та підтверджена ефективність методу. Цей метод спрощує обґрунтування так як показує важливість кожного критерію і відносно один одного та розбирає всі альтернативи за нормалізованими власними векторами, що надає можливість розуміти, які критерії більш значущі для ОПР.