

# ДЕРЖАВНИЙ ТОРГОВЕЛЬНО-ЕКОНОМІЧНИЙ УНІВЕРСИТЕТ

Кафедра комп'ютерних наук та інформаційних систем

## ВИПУСКНА КВАЛІФІКАЦІЙНА РОБОТА

на тему:

### "Web-сервіс оцінки професійної діяльності працівників торговельних підприємств"

Студентки 4 курсу, 10 групи,  
спеціальності

122 «Комп'ютерні науки»

*підпис студента*

Підковка Тетяна  
Олександрівна

Науковий керівник  
доктор фізико-  
математичних наук,  
професор

*підпис керівника*

Пурський Олег  
Іванович

Гарант освітньої програми  
кандидат технічних наук,  
доцент

*підпис керівника*

Демідов Павло  
Георгійович

Київ 2023

# Державний торговельно-економічний університет

Факультет інформаційних технологій

Кафедра комп'ютерних наук та інформаційних систем Спеціальність 122 «Комп'ютерні науки»

Зав. кафедри \_\_\_\_\_

Затверджую  
Пурський О.І.  
«12» грудня 2022р.

## Завдання на випускню кваліфікаційну роботу студентці

Підковка Тетяна Олександрівна

(прізвище, ім'я, по батькові)

1. Тема випускного кваліфікаційного проекту (проекту)  
«Web-сервіс оцінки професійної діяльності працівників торговельних підприємств»

Затверджена наказом ректора від «09» грудня 2022 р. № 3332

2. Строк здачі студентом закінченої роботи 30 травня 2023 року

3. Цільова установка та вихідні дані до роботи

Мета проекту: Розробити математичне, алгоритмічне та веб-орієнтовне програмне забезпечення оцінки професійної діяльності працівників торговельних підприємств

Об'єкт дослідження: засоби створення веб-орієнтованих інформаційних систем

Предмет дослідження: моделі, методи та інформаційні технології в системі оцінки діяльності працівників

4. Перелік графічного матеріалу

---



5. Консультанти по роботі із зазначенням розділів, за якими здійснюється консультування:

Розділ	Консультант (прізвище, ініціали)	Підпис, дата	
		Завдання видав	Завдання прийняв
1	Пурський О.І.	15.12.2022 р.	15.12.2022 р.
2	Пурський О.І.	15.12.2022 р.	15.12.2022 р.
3	Пурський О.І.	15.12.2022 р.	15.12.2022 р.

6. Зміст випускної кваліфікаційної роботи (перелік питань за кожним розділом)

### ВСТУП

#### РОЗДІЛ 1. Дослідження та аналіз підходів до професійної діяльності працівників торгових підприємств

##### 1.1. Проблеми оцінки професійної діяльності працівників торгових підприємств

##### 1.2 Теорія нечітких множин та нечіткої логіки

##### 1.3 Характеристика багатокритеріальних задач

#### РОЗДІЛ 2. Проектування програмного забезпечення вирішення задачі оцінки професійної діяльності працівників торгових підприємств

##### 2.1 Постановка задачі на розробку системи оцінки професійної діяльності працівників торгових підприємств

##### 2.2 Метод нечіткого логічного виведення в розв'язанні задачі професійної діяльності працівників торгових підприємств

##### 2.3 Метод визначення вагових коефіцієнтів правил виведення

#### РОЗДІЛ 3. Інформаційна технологія оцінки професійної діяльності працівників торгових підприємств

##### 3.1 Алгоритмічна реалізація методу нечіткого логічного виведення

##### 3.2 Алгоритмічна реалізація методу визначення вагових коефіцієнтів

##### 3.3 Характеристика програмного забезпечення системи оцінки професійної діяльності працівників торгових підприємств

### ВИСНОВКИ

## СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

### 7. Календарний план виконання роботи

№ По р .	Назва етапів випускної кваліфікаційної роботи	Строк виконання етапів проекту	
		За планом	фактично
1	2	3	4
1	<i>Вибір теми випускного кваліфікаційного проекту</i>	04.10.2022	04.10.2022
2	<i>Розробка та затвердження завдання на випускний кваліфікаційний проект</i>	15.12.2022	15.12.2022
3	<i>Вступ</i>	03.02.2023	03.02.2023
4	<i>РОЗДІЛ 1. Дослідження та аналіз підходів до оцінки професійної діяльності працівників торгових підприємств</i>	28.02.2023	28.02.2023
5	<i>РОЗДІЛ 2. Проектування програмного забезпечення вирішення задачі оцінки професійної діяльності працівників торгових підприємств</i>	06.04.2023	06.04.2023
6	<i>РОЗДІЛ 3. Інформаційна технологія оцінки професійної діяльності працівників торгових підприємств</i>	12.05.2023	12.05.2023
7	<i>Висновки</i>	15.05.2023	15.05.2023
8	<i>Задача випускного кваліфікаційного проекту на кафедрі науковому керівнику</i>	30.05.2023	30.05.2023
9	<i>Попередній захист випускного кваліфікаційного проекту</i>	31.05.2023 - 01.06.2023	31.05.2023 - 01.06.2023



11	<i>Виправлення зауважень, зовнішнє рецензування випускного кваліфікаційного проекту</i>	02.06.2023	02.06.2023
12	<i>Представлення готового зшитого випускного кваліфікаційного проекту на кафедру</i>	05.06.2023	05.06.2023
13	<i>Публічний захист випускного кваліфікаційного проекту</i>	<i>За розкладом роботи ЕК</i>	<i>За розкладом роботи ЕК</i>

8. Дата видачі завдання «15» грудня 2022 р.

9. Керівник випускної кваліфікаційної роботи

Пурський О.І. \_\_\_\_\_

*(прізвище, ініціали, підпис)*

10. Гарант освітньої програми

Демідов П.Г. \_\_\_\_\_

*(прізвище, ініціали, підпис)*

11. Завдання прийняв до виконання студент-дипломник

Підковка Т.О. \_\_\_\_\_

*(прізвище, ініціали, підпис)*

12. Відгук керівника випускної кваліфікаційної роботи

---



---



---



---



---



---



---



---



---



---



---



---



---



---



---



---



---



---



---



---



---



---

Керівник випускної кваліфікаційної роботи

Пурський О.І. 30.05.2023 р.

*(підпис, дата)*

### 13. Висновок про випускну кваліфікаційну роботу

Випускна кваліфікаційна робота студентки Підковки Тетяни  
Олександрівни може бути допущений до захисту в екзаменаційній комісії.

Гарант освітньої програми Демідов П.Г. *(підпис,*

*прізвище, ініціали)*

Завідувач кафедри Пурський О.І.

*(підпис, прізвище, ініціали)*

«    » 2023 р.



### **Анотація**

У випускній кваліфікаційній роботі здійснено комплексну розробку моделей та інформаційної технології оцінки професійної діяльності працівників торгових підприємств з метою підвищення ефективності діяльності підприємства. Теоретичного обґрунтовано основні положення формування і проведення оцінки професійної діяльності працівників торгових підприємств та запропоновано концепцію створення інформаційної системи оцінювання професійної діяльності працівників торгових підприємств. Розроблено метод автоматизованого розрахунку комплексної оцінки працівників. Створено програмне забезпечення оцінки професійної діяльності працівників торгових підприємств

**Ключові слова:** оцінка працівників, метод нечітких множин та нечіткої логіки, інформаційна технологія.

### **Anotation**

In the final qualification work, a comprehensive development of models and information technology for evaluating the professional activity of employees of trade enterprises was carried out in order to improve the efficiency of the enterprise. The basic principles of formation and evaluation of the professional activity of employees of trade enterprises are theoretically substantiated, and the concept of creating an information system for evaluating the professional activity of employees of trade enterprises is proposed. A method of automated calculation of comprehensive assessment of employees has been developed. The software for evaluating the professional activity of employees of trade enterprises has been created

**Keywords:** employee evaluation, method of fuzzy sets and fuzzy logic, information technology.

## Зміст

<b>ВСТУП.....</b>	<b>9</b>
<b>РОЗДІЛ 1. ДОСЛІДЖЕННЯ ТА АНАЛІЗ ПІДХОДІВ ДО ОЦІНКИ ПРОФЕСІЙНОЇ ДІЯЛЬНОСТІ ПРАЦІВНИКІВ ТОРГОВИХ ПІДПРИЄМСТВ</b>	
1.1. Проблеми оцінки професійної діяльності працівників торгових підприємств...	12
1.2. Теорія нечітких множин та нечіткої логіки.....	15
1.3. Характеристика багатокритеріальних задач.....	19
<b>РОЗДІЛ 2. ПРОЕКТУВАННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ВИРІШЕННЯ ЗАДАЧІ ОЦІНКИ ПРОФЕСІЙНОЇ ДІЯЛЬНОСТІ ПРАЦІВНИКІВ ТОРГОВИХ ПІДПРИЄМСТВ</b>	
2.1. Постановка задачі на розробку системи оцінки професійної діяльності працівників торгових підприємств.....	26
2.2. Метод нечіткого логічного виведення в розв’язанні задачі оцінки професійної діяльності працівників торгових підприємств.....	29
2.3. Метод визначення вагових коефіцієнтів правил виведення..	33
<b>РОЗДІЛ 3. ІНФОРМАЦІЙНА ТЕХНОЛОГІЯ ОЦІНКИ ПРОФЕСІЙНОЇ ДІЯЛЬНОСТІ ПРАЦІВНИКІВ ТОРГОВИХ ПІДПРИЄМСТВ</b>	
3.1. Алгоритмічна реалізація методу нечіткого логічного виведення.....	38
3.2. Алгоритмічна реалізація методу визначення вагових коефіцієнтів.....	42
3.3. Характеристика програмного забезпечення системи оцінки професійної діяльності працівників торгових підприємств.....	44
<b>ВИСНОВКИ.....</b>	<b>49</b>
<b>СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....</b>	<b>50</b>
<b>ДОДАТКИ.....</b>	<b>54</b>



## ВСТУП

У процесі здійснення господарської чи комерційно-виробничої діяльності постійно виникає потреба у залученні робочої сили на підприємство. Процес залучення працівників можемо поділити на відбір та підбір. Відбір кадрів - це діяльність, у процесі якої здійснюється вибір зі списку можливих кандидатів одного, який більш повно за своїми якостями відповідає визначеним роботодавцем критеріям для заповнення того чи іншого вакантного робочого місця. Зазвичай, відбір проводиться поетапно, що залежить від вакантної посади, службових обов'язків та особистих якостей претендента [11].

Процес оцінки професійної діяльності працівників торгових підприємств досить трудомісткий і якщо у випадку з невеличкою компанією її засновник чи менеджер з кадрів котрий був в команді з її зародження може досить ефективно опиратися на власні судження і знання справи якою займається, то при рекрутингу людей до великих корпорацій підбір персоналу стає складнішим. Обов'язки підбору персоналу у різні відділи чи навіть офіси вже неможливо здійснити декількома людьми, які точно знають якого працівника шукати і якими рисами він повинен володіти. Ці обов'язки делегуються іншим людям, а ті делегують іншим і т.д. Це є нормальним процесом росту організації. Але це також і впливає на якість підбору кадрів. Будь то відсутність контролю за рекрутерами, некомпетентність чи банальні людські помилки. А підбір некомпетентного персоналу може привести до зменшення ефективності роботи підприємства, зменшення його прибутків або навіть до збитків. Таким чином постає необхідність розробки методів і засобів автоматизації підбору персоналу, з метою підвищення ефективності роботи підприємства, що і зумовило **актуальність** обраної теми дослідження.

**Аналіз останніх досліджень і публікацій.** Дослідження проблем підбору кадрів, а також формування ефективних систем управління підприємствами привертає усе більше уваги як науковців, так й фахівців. Їх реалізація вимагає від усіх учасників ринку та пов'язаних із ними суб'єктів, постійного пошуку раціональних форм організації діяльності, що висуває на перший план підвищення ефективності управління проектами зі створення нових та забезпечення результативного функціонування існуючих підприємств.

Проблеми підвищення ефективності добору працівників, незважаючи на їх актуальність і практичну значимість, досліджується на недостатньому рівні. Активніше досліджено загальні проблеми управління людськими ресурсами.

Наявний цілий ряд робіт вчених В. А. Архипова, Л. О. Кості, Е. В. Маслова та інших, у яких за часів адміністративно-командної економіки досліджувалася сутність близького йому поняття „кадрове забезпечення підприємств”, однак і за його розгляду єдина точка зору серед них була відсутня, що не дозволяло провести чітку межу між вищезазначеними поняттями та розробити методичні підходи щодо вдосконалення кадрового забезпечення.

За останнє десятиліття з'явилося чимало наукових праць, присвячених різним аспектам організації роботи з кадрами, авторами яких є: В. Г. Бикова, Н. Д. Лук'яненко, Г. М. Чорний, В. Н. Гончарова, С. И. Радомський, Л. С. Лісогора та інші [30].

Закордонний досвід із проблем керування кадрами описаний у роботах У. Бреддика, М. Вудкока, Дж. (мол.) Грейсона, К. О'делла, Г. Десслера, Д. Мерсера, Я. Мейтланда, Р. С. Сміта, Д. Френсіса, Дж. Ханта, Р. Дж. Еренберга та інших[43]. Концептуальні положення, що розкривають



залежність зміни трудових процесів від застосування нових інформаційних технологій, розкриті в дослідженнях тільки закордонних авторів: Є. Дайсона, М. Кастельса, Дж. Милковича, М. Розенберга, Є. Тоффлера, П. Хейне, А. Шермана.[10]

**Мета і завдання дослідження.** Метою даного дослідження є розробка математичного, алгоритмічного та програмного забезпечення оцінки кандидатів та оцінки професійної діяльності працівників торгових підприємств. Для досягнення поставленої мети необхідно було вирішити наступні **завдання**:

- провести комплексне аналітичне дослідження проблематики підбору персоналу на підприємство;
- дослідити методи оцінки професійної діяльності працівників торгових підприємств;
- розробити математичну модель підбору персоналу;
- розробити метод автоматизованого визначення оцінки якостей персоналу;
- розробити алгоритм оцінки професійної діяльності працівників торгових підприємств;
- розробити автоматизовану систему оцінки професійної діяльності працівників торгових підприємств.

**Об'єкт дослідження:** процеси оцінки професійної діяльності працівників торгових підприємств.

**Предмет дослідження:** моделі, методи та інформаційні технології в системі оцінки професійної діяльності працівників торгових підприємств.

**Методи дослідження:** теоретичною основою дослідження є загальнонауковий аналітичний метод, теорія нечітких множин та нечіткої логіки, системний підхід і праці провідних вчених з розв'язання

багатокритеріальних задач, з проблем дослідження підбору персоналу. Для практичного вирішення поставлених задач використовувались такі методи:

- загальнонауковий аналітичний метод;
- метод нечіткого логічного виведення в розв'язанні задачі оцінки професійної діяльності працівників торгових підприємств;
- метод визначення вагових коефіцієнтів правил виведення;
- метод алгоритмічного програмування, для створення автоматизованої системи оцінки професійної діяльності працівників торгових підприємств.

**Практичне значення.** Така інформаційна технологія процесу підбору фахівця, може забезпечити підвищення продуктивності й ефективності, спростити виконання менеджером (керівником) своїх задач і скоротити кількість кроків у процесі добору і розподілу персоналу, а також може дозволити знизити обсяг "паперової" роботи, у результаті чого значно скоротиться кількість помилок, підвищиться якість даних – їх цілісність і несуперечність.

Також при використанні такої системи скоротяться витрати – збільшуючи продуктивність і оптимізуючи розподіл завантаження робітників підприємства, можна значно скоротити операційні й адміністративні витрати.



# РОЗДІЛ 1.

## ДОСЛІДЖЕННЯ ТА АНАЛІЗ ПІДХОДІВ ДО ОЦІНКИ ПРОФЕСІЙНОЇ ДІЯЛЬНОСТІ ПРАЦІВНИКІВ ТОРГОВИХ ПІДПРИЄМСТВ

### 1.1 Проблеми оцінки професійної діяльності працівників торгових підприємств

Однією з найбільш важливих складових частин найму працівників є професійний відбір, який значною мірою визначає результативність управління персоналом. Тому в сучасних організаціях, як правило, оцінки персоналу надається належна увага. Об'єктивне рішення про вибір, в залежності від обставин, може ґрунтуватися на створенні враження про кандидата, рівня його професійних навичок, досвіду попередньої роботи, особистих якостях[34, с.61-65].

Але цей процес досить трудомісткий і вимагає від підприємства витрат часу, коштів та людського капіталу. І на підприємствах, які не приділяють цьому належної уваги часто виникають проблеми, безпосередньо пов'язані з роботою працівників. Наприклад, зниження ефективності роботи підприємства, зменшення його прибутків, плинність кадрів, конфлікти між керівниками та підлеглими, нездорова атмосфера в колективі.

Ці проблеми спричинені не лише браком часу чи коштів, існує ряд інших причин, які перелічені в праці з управління персоналу Одегова Ю.Г. та Ніконової Т.В.[36]:

- Стихійний відбір, в наслідок якого відбувається часта зміна працівників на одному робочому місці.
- Відсутність у компанії єдиних критеріїв оцінки професійної діяльності працівників торгових підприємств. Тобто в залежності від посади

до кандидата застосовуються різні критерії оцінювання, що не дає змогу визначити відповідність фахівця загальним цілям і культурі підприємства, що і може спричинити внутрішні конфлікти.

- Відбір кандидата виключно за побажаннями керівника, що може суперечити внутрішній культурі організації.
- Невідповідальне ставлення до підбору фахівців, яке може проявлятися у формі занижених (готовність взяти будь-якого кандидата) або навпаки завищених вимог; у виборі кандидата з домінуванням “непрофільних” вимог.
- Невідпрацьована схема у прийнятті рішень щодо кандидата. Наприклад, рішення щодо одного кандидата може виносити один керівник, а щодо іншого - декілька керівників, причому у декілька етапів. Іншим прикладом незбалансованого процесу прийняття рішення є його надмірна тривалість (два і більше тижні).
- Недопрацьована система оцінювання, що проявляється у надмірних або навпаки у недостатніх вимогах до кандидата. Це може стосуватись як окремого підрозділу, так і підприємства в цілому.

Ті компанії, які все ж приділяють належну увагу підбору фахівців користуються методами оцінки, які можна розділити до двох груп: традиційні та нетрадиційні. До традиційних методів відносять тестування на профпридатність, біографічні тести та виявлення біографії, особистісні тести, інтерв'ю[16]. Нетрадиційними методами є використання поліграфу (детектор брехні)[12], методики психоаналізу [4] та тести, які ґрунтуються на аналізах сечі і крові, експертиза почерку, метод аналізу ранніх дитячих спогадів.

Але і ці методи оцінювання мають ряд недоліків. Що стосується традиційних методів, а саме тестувань, то фахівці виділяють такі недоліки:



- Труднощі у виконанні тестів через незвичну форму подання питань, що не зустрічається у повсякденному житті.
- Можливість підготуватись до тесту. Тести, які найчастіше використовуються підприємствами є в широкому доступі в Інтернеті та зустрічаються в спеціальній літературі.
- Використання неадаптованих в Україні закордонних тестів.
- Відсутність перевіреної інформації про зв'язок результатів тестів з результатами подальшої діяльності відібраних кандидатів[14].

Ще один популярний метод оцінювання - інтерв'ю є процесом міжособистісної взаємодії, а тому виникає ряд ризиків, які пов'язані саме з міжособистісним сприйняттям:

- помилка центральної тенденції виникає у випадку оцінювання більшості кандидатів середнім балом;
- ефект ореолу, який виникає якщо інтерв'юер орієнтується в оцінці кандидата на якусь певну рису (хорошу чи погану), яка є на його погляд головною;
- помилка контрасту, яка виникає, якщо середній кандидат отримує високу оцінку через те, що проходить інтерв'ю після декількох слабких претендентів і навпаки - низьку, якщо йде після сильних.

Нетрадиційні методи оцінки претендентів не можуть використовуватись як основні або єдині засоби оцінки тому що оцінюють переважно особистісні, а не професійні якості кандидата та є досить суб'єктивними.

Данюк В.М., Петюх В.М., Цимбалюк С.О. розрізняють методи в залежності від форми вираження кінцевого результату на три групи: описові, кількісні та комбіновані методи. При описових методах оцінка подається в текстовій формі, при кількісних і комбінованих

використовуються шкали оцінок, які дозволяють кількісно виміряти різні рівні виконання робіт в організації[15].

Фрагмент після переписування:

Згідно з Федуловою Л.І., методи оцінки персоналу можна розділити на індивідуальні та групові методи [40]. Враховуючи недоліки, які були згадані вище, можна сформулювати загальні критерії для технологій оцінки претендентів на вакантні посади:

1. Об'єктивність - незалежність від будь-яких особистих думок.
2. Надійність - захищеність від зовнішніх впливів, таких як настрої, погода, минулі успіхи або невдачі.
3. Достовірність - оцінка реальних вмінь, знань та навичок претендента.
4. Комплексність - відповідність кандидата не лише вимогам посади, але й цілям та завданням підприємства.
5. Доступність - критерії та методи, які використовуються при оцінці, повинні бути зрозумілими та доступними не тільки фахівцям, але й тим, кого оцінюють [27].

Автоматизовані системи оцінки кандидатів використовуються саме для уникнення помилок, які можуть виникнути через людський фактор при прийнятті рішення про прийняття чи відхилення кандидата на посаду. Математичне моделювання замінює складні тестування обчисленнями, що дозволяє проводити порівняння варіантів швидше та точніше.

## **1.2 Теорія нечітких множин та нечіткої логіки**

Переписаний фрагмент:



Для автоматизації процесу оцінювання кандидатів потрібний математичний інструментарій, який може взаємодіяти з людиною та перетворювати конкретні критерії на чітку та формальну мову математичних формул. Один з таких інструментів - теорія нечітких множин, яка була розроблена відомим американським математиком Лотфі Заде наприкінці 60-х років [3]. Заде створив цей інструмент для опису процесів інтелектуальної діяльності, включаючи нечіткість та невизначеність виразів (наприклад, відповідний, більш відповідний, менш відповідний кандидат). У відмінність від класичного поняття множини, де функція може мати лише значення 0 та 1, нечіткі множини за Заде можуть приймати будь-які значення в інтервалі  $[0,1]$ .

Методи нечітких множин особливо корисні, коли відсутня точна математична модель функціонування системи. Теорія нечітких множин дозволяє застосовувати неточні та суб'єктивні експертні знання про предметну область без їх формалізації у традиційні математичні моделі. З використанням теорії нечітких множин вирішуються питання узгодження суперечливих критеріїв прийняття рішень та створення логічних регуляторів систем. Нечіткі множини дозволяють застосовувати лінгвістичний опис складних процесів, встановлювати нечіткі відношення між поняттями, прогнозувати поведінку системи, формувати множину альтернативних дій та робити формальний опис нечітких правил прийняття рішень [25].

Основними поняттями теорії нечітких множин є нечітка множина, лінгвістична змінна, терм, терм-множина.

Розглянемо поняття нечіткої множини. Нехай  $X$  – універсальна множина,  $x$  – елемент  $X$ , а  $G$  – деяка властивість. Нечіткою підмножиною  $A$  універсальної множини  $E$  називається множина впорядкованих пар:  $A = \{\mu_A(x) / x\}$ , де  $\mu_A(x)$  - функція належності, яка приймає значення в деякій заданій множині  $M$  (наприклад  $M = [0,1]$ ). Функція належності  $\mu_A(x)$  вказує ступінь істинності, з яким елемент  $x$  з  $X$  належить підмножині  $A$ . Нечітка підмножина  $A$  може бути також названою нечіткою множиною на універсальній множині  $X$ . Якщо множина  $M$  складається тільки з двох чисел 0 та 1, то нечітка підмножина  $A$  стає звичайною підмножиною універсальної множини  $X$ [24]. Функція належності може задаватись:

- самостійно дослідником на основі власного досвіду; - в просторовому випадку;
- із залученням експертів, обробкою їх оцінок - для більш відповідальних випадків[28].

Існує більше десятка типових форм кривих для задання функції належності. Найбільшою популярністю користуються трикутна, трапецієвидна та гаусова функції належності.



Найменування функції	Аналітичний вираз	Інтерпретація параметрів
Трикутна	$f_6(x; a; b; c) = \begin{cases} 0, & x \leq a \\ \frac{x-a}{b-a}, & a \leq x \leq b \\ \frac{c-x}{c-b}, & b \leq x \leq c \\ 0, & c \leq x \end{cases}$	a, c – носій нечіткої множини; b – координата максимуму
Трапецієвидна	$f_7(x; a; b; c; d) = \begin{cases} 0, & x \leq a \\ \frac{x-a}{b-a}, & a \leq x \leq b \\ 1, & b \leq x \leq c \\ \frac{d-x}{d-c}, & c \leq x \leq d \\ 0, & d \leq x \end{cases}$	a, d – носій нечіткої множини; b, c – ядро нечіткої множини
Гаусова	$f_8(x; \sigma; c) = e^{-\frac{(x-c)^2}{2\sigma^2}}$	c – координата максимуму; $\sigma$ – коефіцієнт концентрації
Сигмоїдальна	$f_{sig}(x; a; b) = \frac{1}{1 + e^{-a(x-b)}}$	a – коефіцієнт крутизни; b – координата переходу через 0,5
Сінгтон	$f_{sing}(x; a) = \begin{cases} 1, & x = a \\ 0, & x \neq a \end{cases}$	a – нечітке число, що представлене нечіткою множиною

Рис.1 Основні функції належності

За допомогою нечітких множин формалізують терми лінгвістичних змінних нечітких моделей. Лінгвістична змінна характеризується набором (L, T (L), U, G, M), в якому L – назва змінної; T (L) – терм-множина змінної L, U

- універсальна множина базових значень (область, в якій визначені значення лінгвістичної змінної); G – синтаксичне правило; M – семантичне правило [6]. Терм-множина T (L) являє собою сукупність термів – назв лінгвістичних значень змінної L. Кожному терму відповідає нечітка підмножина безлічі U, визначальне лінгвістичне значення терма. Іншими словами, зміст терма характеризується функцією належності  $\mu: U \rightarrow [0, 1]$ , яка кожному елементу, де u належить U ставить у відповідність значення приналежності цього елементу терму. Синтаксичне правило G має зазвичай форму граматики, що

породжує терми. Терм, що складається з одного або більше атомарних термів, називається складовим термом.

Семантичне правило М ставить у відповідність кожному атомарному терму його зміст у вигляді нечіткої множини. Крім того, семантичне правило М пов'язує приналежності атомарних термів в складені лінгвістичні значення з приналежністю складеного значення[26].

Важливість використання логіко-лінгвістичних змінних обумовлена наступними причинами:

- не всі критерії оптимізації можуть задаватись кількісно;
- між рядом параметрів, які впливають на функціонування системи, не вдається встановити точних кількісних залежностей;
- процес автоматизації системи - багатокроковий, а зміст кожного кроку не завжди можна визначити однозначно;
- існуючі описи системи та її структури дуже громіздкі, їх важко, а часто і неможлива використовувати[24, с.78].

Для нечітких множин можна визначити такі ж логічні операції, які і для однозначної логіки, наприклад, NOT, OR, AND[6]. Також для нечітких множин характерні такі властивості, як: об'єднання відношень, операції включення, комутативність, асоціативність, ідемпотентність, дистрибутивність, рефлексивність, транзитивність, симетричність[17].

Основними характеристиками нечітких множин є наступні характеристики:

- Носій нечіткої множини А - чітка множина А1 таких точок в U, для яких величина  $\mu_A(x)$  додатна, тобто  $A1 = \{x \mid \mu_A(x) > 0\}$ .
- Висота нечіткої множини А - верхня межа її функції належності  $\mu_A(x)$ .



- Нормальна нечітка множина  $A$ , якщо  $\mu_A(x) = 1$ . В іншому випадку воно називається субнормальною нечіткою множиною [44].
- Ядро нечіткої множини - чітка підмножина універсальної множини  $U$ , елементи якої мають ступені належності рівні 1. Ядро субнормальної нечіткої множини пуста.
- Точка переходу нечіткої множини  $A$  - це такий елемент  $x \in U$ , для якого  $\mu_A(x) = 0,5$ .

Головною операцією нечіткої логіки є процедура нечіткого висновку, за допомогою якої з нечітких умов отримують наближенні рішення. Процедура нечіткого висновку заснована на операції логічного слідування. Імплікація дозволяє формалізувати знання експерта за формою “якщо  $A$ , то  $B$ ”, де  $A$  - передумова, а  $B$  - висновок. Стосовно нечіткого управління  $X$  - базова множина значень  $x$  регулярної змінної. Тоді  $X$  базова множина  $x$ ,  $A$  - множина значень  $x$ ,  $U$  - базова множина значень управління  $u$ ,  $B$  - множина значень  $u$ .

В залежності від способу отримання логічних висновків з нечітких правил можуть бути різні регулятори. Наприклад, алгоритми Мамдані, Ларсена, Цукамото, Сугено, системи нечіткого логічного виведення із зваженою істинністю.

### 1.3 Характеристика багатокритеріальних задач

Вибір кандидата на посаду в підприємстві можна віднести до задачі багатокритеріального вибору, в якій у ролі альтернатив виступає саме скінченна кількість претендентів на вакантну посаду, і ці альтернативи якісно оцінюються. Особливістю таких задач, які виступають у ролі певної моделі є те, що вибір здійснюється з множини альтернатив, які не покращуються[37].

В економічні діяльності при моделюванні процесів прийняття рішень, які вимагають розгляду багатьох альтернатив, тобто є багатокритеріальними виникають дві ситуації:

- Повна формалізація процесу пошуку найвигідніших для підприємства рішень, можлива лише для вивчених та кількісно сформульованих задач.
- Для розв'язання якісно виражених задач повністю сформованих алгоритмів не існує, окрім примітивного та ненадійного алгоритму перебору (метод проб та помилок).

Також проблеми, що виникають в процесі управління соціально-економічними системами можна поділити на три типи.

До першого типу належать проблеми, які чітко виражені, тобто числами або символами і тому при їх вирішенні не виникає труднощів, оскільки вони проблеми такого типу добре формалізовані. Прикладами таких задач на підприємстві є задачі бухгалтерського обліку, складського обліку.

До другого типу проблем належать погано структуровані проблеми, для яких описані лише певні характеристики, ознаки та ресурси, але не встановлені зв'язки між ними, відсутні кількісні залежності. Розв'язання таких задач можливе лише у разі використання неформалізованих процедур, які базуються на неструктурованій та невизначеній інформації. До таких проблем на підприємстві відносяться проблеми, що пов'язані з плануванням, організаційним перетворенням, прийняттям стратегічних рішень. Більшість неструктурованих проблем розв'язується за допомогою евристичних методів, у яких не передбачена жодна упорядкована логічна процедура пошуку їх розв'язку, а сам метод цілком залежить від особистих характеристик людини (інформованості, кваліфікації, досвіду, інтуїції



тощо). Особливістю евристичних методів є здатність здійснювати пошук рішення навіть тоді, коли не сформульовано задачу і невідомі способи її розв'язування[42, с.159].

Переписаний фрагмент:

Для автоматизації процесу оцінювання кандидатів потрібний математичний інструментарій, який може взаємодіяти з людиною та перетворювати конкретні критерії на чітку та формальну мову математичних формул. Один з таких інструментів - теорія нечітких множин, яка була розроблена відомим американським математиком Лотфі Заде наприкінці 60-х років [3]. Заде створив цей інструмент для опису процесів інтелектуальної діяльності, включаючи нечіткість та невизначеність виразів (наприклад, відповідний, більш відповідний, менш відповідний кандидат). У відмінність від класичного поняття множини, де функція може мати лише значення 0 та 1, нечіткі множини за Заде можуть приймати будь-які значення в інтервалі  $[0,1]$ .

Методи нечітких множин особливо корисні, коли відсутня точна математична модель функціонування системи. Теорія нечітких множин дозволяє застосовувати неточні та суб'єктивні експертні знання про предметну область без їх формалізації у традиційні математичні моделі. З використанням теорії нечітких множин вирішуються питання узгодження суперечливих критеріїв прийняття рішень та створення логічних регуляторів систем. Нечіткі множини дозволяють застосовувати лінгвістичний опис складних процесів, встановлювати нечіткі відношення між поняттями, прогнозувати поведінку системи, формувати множину альтернативних дій та робити формальний опис нечітких правил прийняття рішень [25].

У загальному випадку прийняття рішення відбувається на основі дискретної множини альтернатив. І в результаті цього рішення буде отриману підмножину початкової множини. Звуження цієї множини можливо шляхом порівняння критеріїв між собою, та вибором найбільш привабливих. При цьому вважається, що множина, на якій потрібно цей вибір зробити є попередньо сформованою. В цій множині визначено цілі, заради яких здійснюється вибір та існують критерії оцінки та порівняння альтернатив.

Для вирішення багатокритеріальних задач математичним способом можемо використати наступний набір інформації:  $\langle T, X, R, A, F, G \rangle$ , де  $T$  – постановка задачі вибору рішення;  $X$  – множина допустимих альтернатив;  $R$  – множина критеріїв вибору;  $A$  – множина шкал вимірювань за критеріями (порядкові, інтервальні, відношень, найменувань);  $F$  – відображення множини допустимих альтернатив на множину критеріальних оцінок (наслідки рішень);  $G$  – система переваг особи, що приймає рішення.

Математичні постановки багатокритеріальних задач прийняття рішень можуть суттєво відрізнятись одна від одної. Класифікація цих задач здійснюється за наступними ознаками[2;5]:

- за множиною альтернатив  $X$  задачі поділяються на злічені, скінченні та континуальні;
- за потужністю множини  $R$  (оцінка може здійснюватися за одним або декількома критеріями, які в свою чергу, можуть мати як кількісні, так і якісні характеристики) – задачі прийняття рішень бувають однокритеріальні (з скалярним критерієм) та багатокритеріальні (з векторним критерієм);
- за видом відображення множини  $F$  (детерміновані, ймовірнісні та невизначені) - розрізняють задачі прийняття рішень в умовах визначеності, коли наслідки прийнятого рішення точно відомі; в умовах



ризик, якщо відомі ймовірності можливих наслідків прийнятих рішень; та в умовах невизначеності, коли жодних здогадів не існує;

- за типом систем переваг  $G$  (переваги однієї особи або переваги колективу) - задачі індивідуального та групового вибору;
- за режимом вибору рішення - одноразові задачі, або такі, що повторюються допускаючи навчання в процесі прийняття рішень;
- за відповідальністю прийняття рішень - односторонні чи багатосторонні. При багатосторонньому прийнятті рішення ступінь його узгодженості може коливатись від повної узгодженості до її протилежності, тобто прийняттю рішення в конфліктних умовах.

В теорії для вирішення багатокритеріальних задач найкращою буде альтернатива, що матиме найменше значення за усіма  $n$  критеріями з множини  $X$ , де  $f(x)$ ,  $i = 1, 2, \dots, n$ . Рішення, яке відповідає такій альтернативі є найефективнішим. Проте на практиці розв'язання багатокритеріальних задач можливо лише в тому разі, якщо пошук здійснюється на множині узгоджених рішень. В іншому випадку, така задача не має єдиного розв'язку.

Найбільш поширеними методами розв'язання багатокритеріальних задач є [13, с.8]:

- метод скаляризації, суть якого полягає у зведенні багатокритеріальної задачі до однокритеріальної шляхом введення певного суперкритерію у вигляді скалярної функції векторного аргументу:  $F(x) = G(f_1(x), f_2(x), \dots, f_n(x))$ ;
- метод умовної мінімізації, суть якого полягає у використанні припущення про те, що окремі критерії, як правило, є нерівнозначними між собою (деякі менш важливі, ніж інші). Виділяють головний критерій, а решту розглядають як додаткові, неосновні. Тоді задача прийняття рішення

полягатиме в знаходженні екстремуму основного критерію, за умови, що значення решти критеріїв буде знаходитись у певних інтервалах;

- метод послідовних поступок. Суть методу полягає у впорядкуванні частинних критеріїв у порядку убуття їх важливості, виборі найважливішого критерію і визначенні найкращої за цим критерієм альтернативи. Далі визначають величину, на яку можливо зменшити отримане значення найважливішого критерію, щоб за його рахунок збільшити, наскільки можливо, значення наступного за важливістю критерію і т. д.;

- пошук множини Парето. За цим методом відмовляються від пошуку найкращої альтернативи і погоджуються з тим, що перевагу одній альтернативі перед іншою можна надавати лише у випадку, коли вона краща за усіма критеріями. У випадку, коли перевага за одним критерієм не збігається з перевагою за іншим, такі альтернативи вважають непорівнянними. У результаті попарного порівняння альтернатив, усі гірші за усіма критеріями альтернативи відхиляються, а ті, що залишились (не домінуючі), приймаються. Якщо усі максимально допустимі значення частинних критеріїв не відповідають єдиній альтернативі, то прийняті альтернативи утворюють множину Парето і на цьому процедура прийняття рішення закінчується. При необхідності вибору лише однієї альтернативи вводяться додаткові критерії, обмеження, залучають експертів;

- вибір на основі багатокритеріальної функції корисності використовується, якщо задані  $n$  альтернатив,  $m$  наборів зовнішніх умов та  $k$  критеріїв. Кожна альтернатива оцінюється за усіма критеріями для усіх наборів зовнішніх умов. Далі формують  $k$  матриць  $n \times m$ . Кожен елемент матриці  $x$  являє собою числову оцінку  $i$ -ї альтернативи по критерію  $k$  при реалізації  $j$ -го набору зовнішніх умов. Для прийняття рішення вводиться



багатокритеріальна функція корисності, що має вигляд числової функції, визначеної на множині можливих альтернатив і для якої виконується умова: якщо альтернатива  $x_1$  більш приваблива, ніж альтернатива  $x_2$ , то  $u(x_1) \geq u(x_2)$ . Основна проблема методу полягає у складності побудови єдиної для усіх критеріїв функції корисності. Але усі наведені вище методи мають певні обмеження при використанні у вирішенні реальних економічних проблем. Це пов'язано з тим, що значна кількість економічних параметрів є невизначеними, неточними, нечітко заданими, тому застосування при побудові формальних моделей прийняття рішень детермінованих методів призводить до значних помилок. У ситуаціях прийняття рішень, коли хоча б один із елементів задачі (альтернативи, критерії, обмеження, залежності) описується якісно (нечітко), має місце задача прийняття рішення при нечіткій вхідній інформації.

Перспективним напрямком розробки методів прийняття рішень у нечіткому середовищі є лінгвістичний підхід на базі теорії нечітких множин[1;39;8].

Задача прийняття рішення у нечіткому середовищі характеризується наступними поняттями:

- нечіткими критеріями;
- нечітким оператором максимізації(нечіткі інструкції типу "бажано, щоб значення критерію  $x$  було якомога більшим");
- нечітким критеріальним обмеженням(нечіткі інструкції типу "бажано, щоб значення критерію  $x$  було більше, менше або дорівнювало значенню  $\alpha$ ");
- інформацією про приблизну важливість критеріїв(вагові коефіцієнти, низка пріоритетів)

$X$ , яка включає в себе вибрані альтернативи. Нечітке рішення може бути отримане на основі нечіткої цілі  $G$  і нечітких обмежень  $S$  шляхом

вибору альтернатив, які найкраще задовольняють цілі та відповідають обмеженням. Для цього використовується функція належності, яка визначає ступінь відповідності кожної альтернативи цілі та обмеженням. У процесі прийняття рішення в нечітких умовах важливу роль відіграють операції агрегування та дефазифікації. Агрегування дозволяє поєднати інформацію з різних джерел та врахувати багатоаспектність проблеми. Дефазифікація визначає точку прийняття рішення шляхом перетворення нечіткої множини рішень у чітку кількісну форму. У результаті застосування логічної схеми та нечітких множин при прийнятті рішень в нечітких умовах можливо отримати більш гнучке та адаптивне рішення, яке враховує нечіткість, невизначеність та багатоаспектність проблеми. Такий підхід дозволяє ефективно вирішувати проблеми, що мають якісні та кількісні аспекти і вимагають комплексного аналізу та оцінки просторі альтернатив, що отримана в результаті перетину заданих цілей та обмежень[21].

## **РОЗДІЛ 2.**

### **ПРОЕКТУВАННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ВИРІШЕННЯ ЗАДАЧІ ОЦІНКИ ПРОФЕСІЙНОЇ ДІЯЛЬНОСТІ ПРАЦІВНИКІВ ТОРГОВИХ ПІДПРИЄМСТВ**

#### **2.1 Постановка задачі на розробку системи оцінки професійної діяльності працівників торгових підприємств**

Задача оцінки професійної діяльності працівників торгових підприємств є багатокритеріальною задачею, адже містить в собі певну кількість альтернатив (претендентів на посаду), кожна з яких має певний пріоритет, який залежить від навиків та якостей якими володіють кандидати.

Але характеристики професійної діяльності працівників торгових



підприємств часто задаються не чітко, тобто не мають кількісного вираження, а лише якісне (наприклад, пунктуальність, відповідальність, стресостійкість і т. д. ). В такому випадку має місце задача прийняття рішення при нечіткій вхідній інформації. Тому рішення таких задач здійснюється за допомогою теорії нечітких множин та нечіткої логіки.

Змістовна постановка задачі вибору кандидата на вакантну посаду. Елементи, що належать різним множинам назовемо призначенням, а сукупність  $n$  призначень, що охоплює усі суб'єкти та об'єкти - рішенням задачі. Пред'являючи вимоги до якості призначень, тобто до ступеня в відповідності характеристик елементів множини, допустимої при утворенні пар, формується область допустимих значень, яка також включатиме обов'язкові призначення або виключатиме неприпустимі. Формуючи призначення в області допустимих значень, посадова особа, яка приймає рішення прагне до одного з можливих рішень, за якого не можна поліпшити якість призначення для якої-небудь пари елементів, не погіршивши при цьому якість призначення для інших пар. Назвемо такі рішення ефективними. Серед цих рішень посадова особа прагне відшукати таке, яке дає змогу отримати максимальну кількість якнайкращих можливих призначень. Варіант, який повністю задовольняє співвідношення між можливостями та потребами, назовемо оптимальним.

Враховуючи описані вище особливості, сформулюємо змістовну постановку задачі наступним чином.

Початковим етапом вирішення цієї задачі буде формування лінгвістичних термів типу "якщо, то" (наприклад, якщо кандидат має відповідну кваліфікацію, освіту та досвід бухгалтерського обліку, то він задовольняє вимогам). Наступним кроком буде визначення можливих значень лінгвістичних змінних  $X$  та  $Y$ , які будуть використовуватись для

оцінки кандидатів, наприклад “ якщо  $X_1 =$  відповідний,  $X_2 =$  вища,  $X_3 =$  достатній, то  $Y =$  задовільний ”. Далі потрібно визначити функцію належності та нечіткі множини оцінки кандидатів.

Таб.1.1

Нечіткі значення лінгвістичної шкали

Значення лінгвістичної змінної	Оцінка
Дуже низька	$\mu_{ДН}=\{0,0,0.1,0.3\}$
Низька	$\mu_{Н}=\{0.1,0.3,0.3,0.5\}$
Допустима	$\mu_{Д}=\{0.3,0.5,0.5,0.7\}$
Висока	$\mu_{В}=\{0.5,0.7,0.7,0.9\}$
Дуже висока	$\mu_{ДВ}=\{0.7,0.9,1,1\}$

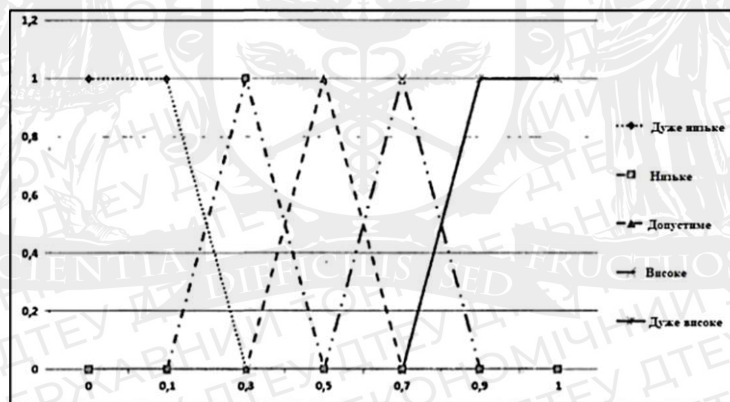


Рис.1.2. Функція належності п'ятибальної лінгвістичної нечіткої шкали

Як видно з малюнка, нечіткі числа «зачеплені» один за одного. Це відображає той факт, що немає різкого поділу між сусідніми оцінками, і перехід від однієї бальної оцінки до іншої відбувається поступово. В результаті оцінки якісної характеристики за допомогою бальної шкали фахівці можуть отримати його нечітку кількісну інтерпретацію.



Для визначення задовільності кожної альтернативи необхідно застосувати правило композиційного виведення в нечіткому середовищі. Потім порівняти нечіткі підмножини в одиничному інтервалі для порівняння найкращого рішення на основі точкових оцінок кожної альтернативи.

Точковими оцінками параметрів розподілу генеральної сукупності будуть такі оцінки, які визначаються одним числом. Наприклад, вибіркова середня  $\bar{x}$ , вибіркова дисперсія  $D_b$  – точкові оцінки відповідних числових характеристик генеральної сукупності. Точкові оцінки параметрів розподілу є випадковими величинами, їх можна вважати первинними результатами обробки вибірки оскільки невідомо, з якою точністю кожна з них оцінює відповідну числову характеристику генеральної сукупності. Якщо об'єм вибірки досить великий, то точкові оцінки задовольняють практичні потреби точності[29]. При розв'язку нашої задачі, в якості найкращої вибираємо альтернативу з найбільшою точковою оцінкою.

Формалізація знань за допомогою правил дозволяє враховувати різну важливість критеріїв і самих правил. В задачі оцінки та вибору кандидата можливі різні підходи: м'який, жорсткий та раціональний. М'який підхід зазвичай має місце в умовах дефіциту часу і кваліфікованих кадрів, основну директиву цього методу можна сформулювати так “аби щось вмів робити”. Жорсткий підхід до вибору кандидата на вакантну посаду можливий у випадку великої кількості кваліфікованих кадрів та достатньої кількості часу, відведеного на вибір кандидата. Ціллю жорсткого оцінки професійної діяльності працівників торгових підприємств пошук кандидата, який найбільш відповідає образу так званого “ідеального” кандидата. Підхід з використанням правил, які мають однакову вагу можна вважати раціональним[2].

Отож, виходячи з вище сказаного для вирішення даної задачі необхідна:

- алгоритмічна реалізація методу нечіткого логічного виведення, який визначатиме найкращу альтернативу, тобто найбільш відповідного кандидата;
- алгоритмічна реалізація методу визначення вагових коефіцієнтів, який буде базуватись на вхідних даних у вигляді лінгвістичних змінних.

### **Метод нечіткого логічного виведення в розв'язанні задачі оцінки професійної діяльності працівників торгових підприємств**

Існує достатньо методів розв'язання задач з нечіткими вхідними даними. Розглянемо їх більш детально.

Система нечіткого логічного виведення для одержання узагальнених оцінок об'єктів має  $n$  входів та один вихід – результат нечіткого логічного виведення. На вхід системи нечіткого логічного виведення подаються вхідні величини (впливові фактори)  $x_i$ ,  $i = 1, 2, \dots, k$ . На виході система видає узагальнену чітку оцінку, що є результатом системи нечіткого логічного виведення для заданих вхідних значень. Нечітке логічне виведення визначає відображення вектора вхідних даних у скалярне вихідне значення за допомогою нечітких правил. Система нечіткого логічного виведення складається з трьох основних компонентів: фазифікатора, механізму логічного виведення та дефазифікатора. Фазифікатор визначає ступінь належності вхідних значень,  $x_i$ ,  $i = 1, 2, \dots, k$  до нечітких множин входу  $K_{x_i}$ ,  $\{K_{x_i}^1, K_{x_i}^2, \dots, K_{x_i}^{m_{x_i}}\}$ , де  $m_{x_i}$  – кількість лінгвістичних змінних з відповідної лінгвістичної шкали для  $i$ -го входу.

Необхідність у введенні процедури фазифікації зумовлена використанням у системі нечіткого логічного виведення лінгвістичних



правил і здійснюється вона для визначення ступеня істинності кожної передумови кожного правила. Основою механізму логічного виведення є нечітка база правил, яка містить лінгвістичні правила. Вони можуть бути задані

$x_i$   $y$

$y$

експертним шляхом чи отримані із деяких інших міркувань. Зазначимо, що це один із найбільш трудомістких та ресурсовитратних етапів нечіткого логічного виведення. Механізм логічного виведення відображає входні нечіткі множини  $K^j$ ,  $i = 1, 2, \dots, k$  кожного правила ц вихідну множину  $K^j$  з набору вихідних лінгвістичних змінних  $K_y = \{K^1, K^2, \dots, K^{m_y}\}$ . Загалом у нечіткій базі  $j$ -те правило ( $j=1, 2, \dots, r$ ) формулюється так:

$$\text{якщо } x_1 \in K^1 \text{ і } x_2 \in K^2 \text{ і } \dots \text{ і } x_k \in K^k \text{ то } y \in K^j \quad (2.1)$$

$x_1$

$x_2$

$x_1$

$x_k$

$y$

Вихідні нечіткі множини  $y_j$ ,  $j = 1, 2, \dots, r$  кожного правила об'єднуються в одну нечітку множину виведення  $y$ . Потім дефазифікатор відображає нечітку множину виведення  $\tilde{y}$  у чітке число  $\check{Y}$ , яке і буде результатом системи нечіткого логічного виведення для заданих входних значень  $x_i$ ,  $i=1, 2, \dots, k$ . Часто вживаними методами дефазифікації є центроїдний метод, методи максимуму, метод центру максимумів, висотна дефазифікація[18].

Нечіткий алгоритм Мамдані [18]. У цьому алгоритмі процес нечіткого логічного виведення будується так. База знань складається з нечітких правил у вигляді. На етапі фазифікації визначаються ступені

належності вхідних значень  $x_i$ ,  $i = 1, 2, \dots, k$  до нечітких множин входу, тобто визначаються ступені істинності  $\mu_i^j(x_i)$  для кожної передумови кожного  $j$ -го правила. Потім для кожного правила на основі ступенів істинності передумов  $\mu_i^j(x_i)$  розраховується ступінь його виконання  $a_j$ . Для цього застосовують композицію на основі оператора мінімуму:

$$a_j = \min \{ \mu_1^j(x_1), \mu_2^j(x_2), \dots, \mu_k^j(x_k) \}, j = 1, 2, \dots, r. \quad (2.2)$$

Для кожного правила на основі ступеня виконання  $a_j$ ,  $j = 1, 2, \dots, r$  розраховується результат його виконання – вихідна нечітка множина з усіченою функцією належності  $\mu^j(y)$ . Визначають усічену функцію належності також за допомогою оператора мінімуму:

$$\mu^j(y) = \min \{ a_j, \mu^j(y) \}, j = 1, 2, \dots, r. \quad (2.3)$$

Механізм логічного виведення завершується тим, що вихідні нечіткі множини виконаних правил за допомогою оператора максимуму агрегуються в нечітку множину виведення  $\bar{y}$ , функція належності якої має вигляд:

$$\mu_{\bar{y}} = \max \{ \mu^1(y), \mu^2(y), \dots, \mu^r(y) \} \quad (2.4)$$

Нечіткий алгоритм Ларсена [18]. За цим алгоритмом фазифікація проводиться так само, як і за алгоритмом Мамдані. На відміну від використання оператора мінімуму, функцію належності  $\mu^j(y)$  результату виконання кожного правила розраховують на основі оператора добутку:

$$\mu^j(y) = a_j \times \mu^j(y), j = 1, 2, \dots, r. \quad (2.5)$$

Механізм логічного виведення завершується так само, як і за алгоритмом Мамдані. Для приведення до чіткості проводять процедуру дефазифікації нечіткої множини. Метод дефазифікації вибирають залежно від конкретної задачі. Придатним для багатьох класів задач та найпоширенішим є центроїдний метод, що полягає у знаходженні центру ваги (центроїду) нечіткої множини, який і обирається за кінцевий результат.



Нечіткий алгоритм Цукамото [18]. Цей алгоритм використовують у тому випадку, коли функції належності вхідних та вихідних величин є монотонними. Фазифікація входів проводиться так само, як і за алгоритмом Мамдані. Для кожного  $j$ -го правила на основі ступенів істинності передумов  $\mu_j^i(x_i)$  розраховується ступінь його виконання  $a_j$ . Для цього застосовують композицію на основі оператора мінімуму. Механізм логічного виведення завершується тим, що розв'язуються рівняння виду:

$$a_j = \mu^j(y), j = 1, 2, \dots, r \quad (2.6)$$

визначаються чіткі значення для кожного правила. Чітке значення вихідної величини знаходиться як зважене середнє.

Нечіткий алгоритм Сугено [18]. У цьому алгоритмі нечіткого логічного виведення база правил складається з правил, у яких виходом є чітке значення, що визначається лінійною функцією:

Правило  $j$  : “Якщо  $x_1 \in K_{x_1}^j$  і  $x_2 \in K_{x_2}^j$  і... і  $x_k \in K_{x_k}^j$  то  $\bar{y}^j$  дорівнює сумі  $a_j^i x_i$ ,  
де  $a_j^i$  – константи.

Остаточний результат логічного виведення  $\bar{Y}$  розраховується як зважене середнє чітких результатів  $y^j$  виконаних правил, в якому ваговими коефіцієнтами є ступені виконання правил  $a_j$ .

Система нечіткого логічного виведення із зваженою істинністю [23]. База нечітких продукційних правил у системі нечіткого логічного виведення створюється або групою експертів, або її можна побудувати із деяких інших міркувань. Важливість всіх впливових факторів при складанні правил вважають однаковою. При застосуванні системи нечіткого логічного виведення коригують процес логічного виведення з урахуванням значень вагових коефіцієнтів усіх впливових факторів. Зазвичай вагові коефіцієнти є нормованими, і їх можна використовувати як коригувальні множники.

Тобто, в системі нечіткого логічного виведення за таким методом ступінь істинності кожної передумови  $\mu^j$ , що відповідає фактору  $x$   $j$ -го правила, множиться на відповідний ваговий коефіцієнт  $p_i$ . Тоді ступінь виконання  $j$ -го правила  $a_j$  при застосуванні системи нечіткого логічного виведення визначатиметься так:

$$a_j = \min \{p_1 \mu^j_1(x_1), p_2 \mu^j_2(x_2), \dots, p_k \mu^j_k(x_k)\}. \quad (2.6)$$

Але у такому випадку значення оцінки за менш важливим фактором, якому відповідає менше значення вагового коефіцієнта, значно впливатиме на значення ступеня виконання  $a_j$  всього правила, тому що у процедурах композиції та імплікації відповідно до логічного “і” використовується оператор мінімуму. Отже, може виникати ситуація штучного заниження значення ступеня виконання певного правила  $a_j$ . Тому такий спосіб зважування ступенів істинності передумов правил для проблеми визначення рівня компетентності може бути некоректним. Для врахування значень вагових коефіцієнтів впливових факторів та забезпечення адекватної участі кожного фактора у формуванні значення ступенів виконання правил у результаті логічного виведення пропонується для визначення ступеня виконання  $a_j$  правила в процедурі композиції ступенів істинності передумов або замість оператора мінімуму використовувати спеціальний апарат зваженої

агрегації значень ступенів істинності  $\mu^j_i(x_i)$  передумов правила. При цьому правило виконується, якщо значення ступенів істинності  $\mu^j(x)$  кожної передумови правила більше за нуль. В окремих випадках доцільно застосовувати в процедурі композиції зваженої суми значень оцінок, тобто лінійної комбінації значень оцінок за всіма факторами та їх вагових



коефіцієнтів. За такого підходу ступінь виконання  $j$ -го правила  $a_j$  при застосуванні системи нечіткого логічного виведення визначатиметься так:

$$a_j \text{ дорівнює сумі } p_i \times \mu^j(x_i), \mu^j(x_i) > 0.$$

Описана система нечіткого логічного виведення із зваженою істинністю об'єднує позитивні моменти ідей різних нечітких алгоритмів і відрізняється від зваженої системи нечіткого логічного виведення, в якій кожному правилу надається своє значення ваги.

## 2.2 Метод визначення вагових коефіцієнтів правил виведення

Відома значна кількість методів оцінювання вагових коефіцієнтів. В різних літературних джерелах вони можуть мати назву коефіцієнтів ваги, вагомості, впливу, значимості тощо. В найпростішому випадку ваговий коефіцієнт визначається як математичне сподівання частки вихідного ефекту, що вноситься окремим показником. Однак, коректніше було б шукати для вагових коефіцієнтів миттєві інтегральні значення, оскільки тоді максимально збільшувалася б вірогідність кінцевої оцінки, отриманої з їх використанням[9]. Існує два основні підходи

визначення вагових коефіцієнтів :  
експертний та розрахунковий. Розглянемо детальніше кожний з них.

Перевагою експертних методів є простота застосування, недолік – можлива недостовірність даних, певна суб'єктивність, трудомісткість збору та обробки результатів. На сьогодні існує багато методів обробки результатів експертиз (методів експертного оцінювання та прогнозування): попарних зіставлень (порівнянь); надання переваг; рангів; нечітких експертних оцінок; аналізу ієрархій та їх удосконалені чи узагальнені варіанти. Крім того, існують методи організації проведення експертиз (інтерв'ю, анкетування, метод Делфі,

метод комісій та ін.) і методи визначення компетентності експертів та вагомості їх оцінок.

Для підтвердження узгодженості думок експертів необхідно застосувати коефіцієнт конкордації, який у подальшому був оцінений за критерієм Пірсона [20;38;41;22]. При оцінці суттєвості зазначених коефіцієнтів виходили з того, що чим ближче до одиниці наближається коефіцієнт конкордації, тим більш узгодженими є думки експертів. Одним із методів підтвердження істотності коефіцієнта конкордації є його оцінка за критерієм Пірсона  $\chi^2$ . Якщо  $\chi^2_{\text{розп.}} > \chi^2_{\text{табл.}}$ , то коефіцієнт конкордації є істотним при ступенях свободи  $f = n-1$  та заданому рівні значимості  $\alpha = 0,01$ .

При визначенні вагових коефіцієнтів на основі попарних порівнянь [35] докладна числова інформація про досліджувані об'єкти не є обов'язковою, а порівняння можна здійснювати за якісними критеріями: “більше-менше”, “краще-гірше”, без уточнення, у скільки разів або на скільки більше чи краще. Метод попарних порівнянь було розроблено відомим американським математиком Т. Сааті.

Кожен показник, підгрупа або група показників співставляються з іншими (показниками, підгрупами та групами відповідно) на основі шкали відносної значущості. Експерти, виконуючи порівняння показників, заповнюють таблицю попарних порівнянь.

Оскільки значення показників невідомі заздалегідь, порівняння виконується на основі суб'єктивних суджень, які чисельно оцінюються за шкалою. Необхідно відмітити, що кожна клітинка таблиці містить не одне число (безпосередню оцінку експерта), а сукупність чисел, з урахуванням того, що порівняння показників проводиться опосередковано через порівняння їх з певним допоміжним фактором. Врахування цих додаткових порівнянь дозволяє підвищити надійність отриманих результатів, або



зменшити кількість необхідних експертів. Результатом порівняння є певне числове значення: показник, який має перевагу, оцінюється цілим числом ( $a_{ij}$ ), протилежний – оберненим до нього ( $1/a_{ij}$ ). Наприклад, якщо показник 1 за рішенням експерта має “помірну перевагу” над показником 2, то перший оцінюється числом  $a_{12} = 2$ , а другий –  $a_{21}=1/2$ . Отримані значення записуються у вигляді матриці попарних порівнянь (табл. 4). Таблиця заповнюється по рядках (при порівнянні показників 1 з 2, оцінка 2 ставиться на перетині рядка 1 і стовпця 2, а оцінка  $1/2$  – на перетині рядка 2 і стовпця 1). На діагоналі заповненої таблиці завжди будуть стояти одиниці – порівняння показника самого з собою буде давати рівноцінність.

Методом ранжування[32] передбачає упорядкування елементів за мірою збільшення або зменшення пріоритетності за деякою ознакою. На основі отриманих кількісних оцінок об'єктів обирають найбільш пріоритетний. Упорядкування проводять у часі чи просторі або за конкретною ознакою для випадків, коли значення ознаки неможливо виміряти з причин практичного або теоретичного характеру.

Визначення вагових коефіцієнтів методом адаптивної оцінки[33] використовується у випадках, коли значущість показників змінюється з часом і відповідно виникає необхідність зменшення чи збільшення їх вагових коефіцієнтів:

- протягом тривалого часу показник стабільний і його значення практично не змінюється;
- величина вагового коефіцієнта може бути пропорційною темпу його зміни;
- при аналізі індикаторів економічної безпеки ряд показників може сягнути порогового значення;

- доцільно зменшити значення його вагового коефіцієнта, збільшивши при цьому ваги показників, які не досягли необхідного порогу; – показник може втратити свій економічний сенс з різних економічних, політичних і соціальних причин.

Визначення вагових коефіцієнтів методом парної кореляції[33] є розрахунковим методом, який проводиться з метою виявлення основних тенденцій зв'язку між показниками економіки. Якщо  $x_i(t)$  – значення  $i$ -го

показника в момент часу  $t$ , то коефіцієнт кореляції між значеннями показників (додатного чи від'ємного) знаходиться за формулою:

$$r_{ij}(t) = \frac{\langle (x_i(t) - \langle x_i \rangle)(x_j(t) - \langle x_j \rangle) \rangle}{\sqrt{\langle (x_i(t) - \langle x_i \rangle)^2 \rangle \langle (x_j(t) - \langle x_j \rangle)^2 \rangle}} \quad (3.1)$$

Коефіцієнт парної кореляції між показниками, близький до одиниці, свідчить про тісний зв'язок між ними. У цьому випадку один з показників можна виключити з розгляду, істотно зменшити його вагу або об'єднати декілька показників в один, додавши їх ваги.

Метод вагових коефіцієнтів, який найчастіше використовують в умовах нечіткої логіки передбачає, що дослідник володіє певними апріорними даними, наприклад, про вагомість окремих показників властивостей. Але вагові коефіцієнти, що ймовірно можуть бути приписані показникам, характеризуються лише нечисловою, неточною і можливо неповною інформацією (ппп-інформацією) на зразок того, що показник



властивостей  $Q_i$  є важливішим для узагальненого показника якості, ніж показник  $Q_j$ , або вказані межі варіювання значень вагових коефіцієнтів  $0 \leq a \leq w \leq b \leq 1$ . Застосовуючи для прикладу метод, який можна було б назвати рендомізацією невизначеності і який базується на застосуванні концепції Т. Баєса, сукупність  $i = 1, m$  вагових коефіцієнтів інтерпретується як вектор в  $m$  - вимірному евклідовому просторі.

Суть методу полягає в моделюванні невизначеності вибору конкретного вектору вагових коефіцієнтів  $w = (w_1, w_2, \dots, w_m)$  з множини  $W(I)$  всіх допустимих вагових коефіцієнтів за допомогою рендомізованого вектору вагових коефіцієнтів, заданого за допомогою деякого розподілу на множині  $W(I)$ . Підставлення рендомізованого вектору вагових коефіцієнтів замість детермінованого дає рендомізований узагальнений комплексний показник, який можна інтерпретувати як "числове зображення nnp-інформації" і який поєднує інформацію про окремі значення показників властивостей разом з нечисловою, неточною і нечіткою інформацію про значимість цих окремих

показників властивостей. Тим самим задача порівняння багатокритеріальних оцінок якості двох складних об'єктів зведена до виявлення якого-небудь стохастичного домінування між двома випадковими величинами. Метод потребує хоча і нечіткої, та все ж апіорної інформації, яку можна отримати знову ж таки експертним методом. В поєднанні з методом регресивних залежностей обидва підходи можна трактувати як шлях об'єктивізації процесу ранжування вагових коефіцієнтів[7].

## РОЗДІЛ 3.

# ІНФОРМАЦІЙНА ТЕХНОЛОГІЯ ОЦІНКИ ПРОФЕСІЙНОЇ ДІЯЛЬНОСТІ ПРАЦІВНИКІВ ТОРГОВИХ ПІДПРИЄМСТВ

### 3.1 Алгоритмічна реалізація методу нечіткого логічного виведення

Для вирішення задачі оцінки професійної діяльності працівників торгових підприємств будуть використані певні критерії їх оцінки. Розділимо ці критерії на декілька груп : функціональні, когнітивні та надпредметні (метапредметні).

Когнітивні критерії включають в себе освіту та рівень знань в широкому розумінні, здобуті в процесі навчання або особистісного пізнавального досвіду. Функціональні критерії включають в себе практичні навички, здобутий досвід. До надпредметних критеріїв відносяться знання та навички, які прямо не стосуються даної професії, в нашому випадку посади бухгалтера. Наприклад, юридична грамотність, знання іноземних мов, знання мов програмування[31].

Для нашої задачі оцінки професійної діяльності працівників торгових підприємств визначимо п'ять критеріїв : кваліфікація, освіта, досвід роботи на подібній посаді, вміння працювати з сучасним програмним забезпеченням та юридична грамотність. Даним критеріям присвоїмо значення  $X_1, X_2, X_3, X_4, X_5$ , а  $Y$  буде означати ступінь відповідності. Для формування правил необхідно визначити можливі значення лінгвістичних змінних  $X_i$  та  $Y$ , які будуть використовуватись для оцінки кандидатів:

$d_1$ : "Якщо  $X_1 =$  належна і  $X_2 =$  вища і  $X_3 =$  достатнє, то  $Y =$  задовільний";  $d_2$ : "Якщо  $X_1 =$  належна і  $X_2 =$  вища і  $X_3 =$  достатнє і  $X_4 =$  здатний, то  $Y =$  більш ніж задовільний";



$d_3$ : "Якщо  $X_1 =$  належна і  $X_2 =$  вища і  $X_3 =$  достатнє і  $X_4 =$  здатний і  $X_5 =$  володіє, то  $Y =$  бездоганний";

$d_4$ : "Якщо  $X_1 =$  належна і  $X_2 =$  вища і  $X_3 =$  достатнє і  $X_4 =$  володіє, то  $Y =$  дуже задовільний";

$d_5$ : "Якщо  $X_1 =$  належна і  $X_2 =$  не вища і  $X_3 =$  достатнє і  $X_5 =$  володіє, то  $Y =$  задовільний";

$d_6$ : "Якщо  $X_1 =$  не має і  $X_3 =$  недостатнє, то  $Y =$  незадовільний".

Змінна  $Y$  задана на множині  $J = \{0; 0,1; 0,2; \dots; 1\}$ . Значення змінної  $Y$  задаються відповідно до наступних функцій належності:

$S =$  задовільний, що визначено як  $\mu_S(x) = x, x \in J$ ;

$MS =$  більш ніж задовільний -  $\mu_{MS}(x) = \sqrt{x}, x \in J$ ;

$J$ ;

$P =$  бездоганний - як  $\mu_P(x) = \{1, \text{якщо } x = 1; 0, \text{якщо } x < 1\}, x \in J$ ;

$VS =$  дуже задовільний - як  $\mu_{VS}(x) = x^2, x \in J$ ;

$US =$  незадовільний - як  $\mu_{US}(x) = 1 - x, x \in J$ .

Вибір здійснюється з-поміж п'яти кандидатів на множині  $U = \{u_1, u_2, u_3, u_4, u_5\}$ . В даній задачі оцінки кандидатів задані наступними нечіткими множинами:

належна (кваліфікація)  $A = \{0,8/u_1, 0,61u_2, 0,5/u_3, 0,1/u_4,$

$0,3/u_5\}$ ; вища освіта  $B = \{0,5/u_1, 1/u_2, 0/u_3, 0,5/u_4, 1/u_5\}$ ;

достатній досвід  $C = \{0,6/u_1, 0,9/u_2, 1/u_3, 0,7/u_4, 1/u_5\}$ ;

здатний (працювати з програмним забезпеченням)  $D = \{1/u_1, 0,3/u_2,$   
 $1/u_3, 0/u_4, 0/u_5\}$ ;

володіє (юридичними знаннями)  $E = \{0/u_1, 0,5/u_2, 1/u_3, 0,8/u_4, 1/u_5\}$ .

З врахуванням уведених означень правила  $d_1 \dots d_6$  приймають вигляд:

$d_1$ : "Якщо  $X = A$  і  $B$ , і  $C$ , то  $Y = S$ ";

$d_2$ : "Якщо  $X = A \text{ і } B, \text{ і } C, \text{ і } D$ , то  $Y = MS$ ";  
 $d_3$ : "Якщо  $X = A \text{ і } B, \text{ і } C, \text{ і } D, \text{ і } E$ , то  $Y = P$ ";  
 $d_4$ : "Якщо  $X = A \text{ і } B, \text{ і } C, \text{ і } E$ , то  $Y = VS$ ";  
 $d_5$ : "Якщо  $X = A, \text{ і не } B, \text{ і } C, \text{ и } E$ , то  $Y = S$ ";  
 $d_6$ : "Якщо  $X = \text{не } A \text{ і не } C$ , то  $Y = US$ ".

Визначимо функції належності  $\mu_m$  для лівих частин приведених правил:

$$\text{для } d_1: \mu_{M_1}(u) = \min(\mu_A(u), \mu_B(u), \mu_C(u));$$

$$M_1 = \{0,5/u_1, 0,6/u_2, 0/u_3, 0,1/u_4, 0,3/u_5\};$$

$$\text{для } d_2: \mu_{M_2}(u) = \min(\mu_A(u), \mu_B(u), \mu_C(u), \mu_D(u));$$

$$M_2 = \{0,5/u_1, 0,3/u_2, 0/u_3, 0/u_4, 0/u_5\};$$

$$\text{для } d_3: \mu_{M_3}(u) = \min(\mu_A(u), \mu_B(u), \mu_C(u), \mu_D(u)\mu_E(u));$$

$$M_3 = \{0/u_1, 0,3/u_2, 0/u_3, 0/u_4, 0/u_5\};$$

$$\text{для } d_4: \mu_{M_4}(u) = \min(\mu_A(u), \mu_B(u), \mu_C(u), \mu_D(u)\mu_E(u));$$

$$M_4 = \{0/u_1, 0,5/u_2, 0,5/u_3, 0,1/u_4, 0,3/u_5\};$$

$$\text{для } d_5: \mu_{M_5}(u) = \min(\mu_A(u), 1 - \mu_B(u), \mu_C(u), \mu_E(u));$$

$$M_5 = \{0/u_1, 0/u_2, 0,5/u_3, 0,1/u_4, 0/u_5\};$$

$$\text{для } d_6: \mu_{M_6}(u) = \min(1 - \mu_A(u), 1 - \mu_C(u));$$

$$M_6 = \{0,2/u_1, 0,1/u_2, 0/u_3, 0,3/u_4, 0/u_5\};$$

Тоді правила можна записати у вигляді:

$$d_1: \text{Якщо } X = M_1, \text{ то } Y = S;$$

$$d_2: \text{Якщо } X = M_2, \text{ то } Y =$$

$$MS; d_3: \text{Якщо } X = M_3, \text{ то } Y$$

$$= P; d_4: \text{Якщо } X = M_4, \text{ то } Y$$

$$= VS; d_5: \text{Якщо } X = M_5, \text{ то}$$

$$Y = S; d_6: \text{Якщо } X = M_6, \text{ то}$$

$$Y = US.$$



Використовуючи, для перетворення правило виду “Якщо  $X = M$ , то  $Y = Q$ ” імплікацію Лукасевича[19]  $\mu_d(u, j) = \min(1, 1 - \mu_M(u) + \mu_Y(j))$ , для кожної пари  $(u, j) \in U \times J$  отримуємо наступні нечіткі відношення на  $U \times J$ :

$u_1$	0,5	0,6	0,7	0,8	0,9	1	1	1	1	1	1
$u_2$	0,4	0,5	0,6	0,7	0,8	0,9	1	1	1	1	1
$D_1 = u_3$	1	1	1	1	1	1	1	1	1	1	1
	0	0,1	0,2	0,3	0,4	0,5	0,6	0,7	0,8	0,9	1

0,9	1	1	1	1	1	1	1	1	1	1
0,7	0,8	0,9	1	1	1	1	1	1	1	1

	0	0,1	0,2	0,3	0,4	0,5	0,6	0,7	0,8	0,9	1
$u_1$	0	0,82	0,95	1	1	1	1	1	1	1	1
$u_2$	0	1	1	1	1	1	1	1	1	1	1
$u_4$	1										
$u_5$	1										

$D_2 = u_3$	1	1	1	1	1	1	1	1	1	1		
		0	0,1	0,2	0,3	0,4	0,5	0,6	0,7	0,8	0,9	1
$u_1$	1	1	1	1	1	1	1	1	1	1	1	1
$u_2$	0,7	0,7	0,7	0,7	0,7	0,7	0,7	0,7	0,7	0,7	0,7	1





$u_2$

0 0,1 0,2 0,3 0,4 0,5 0,6 0,7 0,8 0,9 1

$u_4$

0,9 0,8 0,7

$u_5$

1

$$D_6 = u_3 \parallel 1 \quad 1 \quad 1 \quad 1 \quad 1 \quad 1 \quad 1 \quad 1 \quad 1 \quad 1 \quad 0,9 \parallel$$

В результаті перетину відношень  $D_1, \dots, D_6$  отримуємо загальне функціональне рішення:

	0	0,1	0,2	0,3	0,4	0,5	0,6	0,7	0,8	0,9	1
$u_1$	0,5	0,6	0,7	0,8	0,9	1	1	1	1	0,9	0,8
$u_2$	0,4	0,5	0,5	0,6	0,6	0,7	0,7	0,7	0,7	0,7	0,7
$u_9$	0,9	0,9	0,9	0,9	0,9	1	1	1	1	0,8	0,7
$u_4$	0,91										
$u_7$	0,7	0,7	0,7	0,86	0,95	1	1	1	1	1	1
$u_5$	0,71										
$u_3$	0,51	0,6	0,7	0,8	0,9	1	1	1	1	1	1

$$D = u_3 \parallel 0,51 \quad 0,6 \quad 0,7 \quad 0,8 \quad 0,9 \quad 1 \quad 1 \quad 1 \quad 1 \quad 1 \quad 1 \parallel$$

Для розрахування задовільності кожної із альтернатив застосуємо правило композиційного виводу в нечіткому середовищі:

$E_k = G_k \circ D$ , де  $E_k$  — степінь задовільності альтернативи  $k$ ;  $G_k$  - відображення альтернативи  $k$  у вигляді нечіткої підмножини на  $U$ ,  $D$  - загальне функціональне рішення. Тоді:

$$\mu_{E_k}(u) = \max(\min(\mu_{G_k}(u), \mu_D(u))) \quad (1.1)$$

Крім цього, у випадку  $\mu_G(u) = 0; u \neq u_k, \mu_G(u) = 1; u = u_k$ . Звідси  $\mu_{E_k}(u) = \mu_k(u_k, i)$ . Іншими словами,  $E_k$  є  $k$ -тий рядок в матриці  $D$ .

### 3.2 Алгоритмічна реалізація методу визначення вагових коефіцієнтів

Для визначення вагових коефіцієнтів використаємо метод точкових оцінок. Застосуємо вище описану процедуру для порівняння нечітких підмножин в одиничному інтервалі для отримання найкращого рішення на основі точкових оцінок.

Для першої альтернативи:

$$E_1 = \{0,5/0; 0,6/0,1; 0,7/0,2; 0,8/0,3; 0,9/0,4; 1/0,5; 1/0,6; 1/0,7; 1/0,8; 0,9/0,9; 0,8/1\}.$$

Вирахуємо множини  $E_{j_a}$  і потужність такої множини  $M(E_a)$  за формулою:

$$\sum_{i=1}^n x_i \quad (2.1)$$

для  $0 < a < 0,5; da = 0,5$



$i=1 n$

$$E_{1a} = \{0; 0,1; 0,2; 0,3; 0,4; 0,5; 0,6; 0,7; 0,8; 0,9; 1\} M(E_{1a}) = 0,5$$

для  $0,5 < a < 0,6; da = 0,1$

$$E_{1a} = \{0,1; 0,2; 0,3; 0,4; 0,5; 0,6; 0,7; 0,8; 0,9; 1\} M(E_{1a}) = 0,55$$

для  $0,6 < a < 0,7; da = 0,1$

$$E_{1a} = \{0,2; 0,3; 0,4; 0,5; 0,6; 0,7; 0,8; 0,9; 1\} M(E_{1a}) = 0,6$$

для  $0,7 < a < 0,8; da = 0,1$

$$E_{1a} = \{0,3; 0,4; 0,5; 0,6; 0,7; 0,8; 0,9; 1\} M(E_{1a}) = 0,65$$

для  $0,8 < a < 0,9; da = 0,1$

$$E_{1a} = \{0,4; 0,5; 0,6; 0,7; 0,8; 0,9\} M(E_{1a}) = 0,65$$

для  $0,9 < a < 1; da = 0,1$

$$E_{1a} = \{0,5; 0,6; 0,7; 0,8\} M(E_{1a}) = 0,65$$

Розрахуємо точкову оцінку для  $E_1$ :

$$F(E_1) = \int_0^{\alpha_{max}} M(E_{1a}) da = 0,5 * 0,5 + 0,55 * 0,1 + 0,6 * 0,1 + 0,65 * 0,1 + 0,65 * 0,1 + 0,65 * 0,1 = 0,560$$

$max$

$$\int_0^{\alpha_{max}} M(E_{1a}) da = 1 \int_0^{\alpha_{max}} M(E_{1a}) da = 0,5 * 0,5 + 0,55 * 0,1 + 0,6 * 0,1 + 0,65 * 0,1 + 0,65 * 0,1 + 0,65 * 0,1 = 0,560$$

Аналогічним чином розрахуємо точкові оцінки для інших альтернатив: для другої альтернативи  $F(E_2) = 0,656$ ; для третьої -  $F(E_3) = 0,575$ ; для четвертої

•  $F(E_4) = 0,483$ ; для п'ятої -  $F(E_5) = 0,562$ . В якості найкращої обираємо альтернативу, яка має найбільшу точкову оцінку.

В даному випадку це альтернатива  $u_2$ , відповідно вона і буде найкращою. Друге місце займе альтернатива  $u_3$ ; третє -  $u_5$ ; четверте -  $u_1$ , а найгіршої з усіх альтернатив є  $u_4$ .

Описаний вище алгоритм оцінки професійної діяльності працівників торгових підприємств, який буде використовуватись в програмному забезпеченні зобразимо у вигляді блок-схеми:

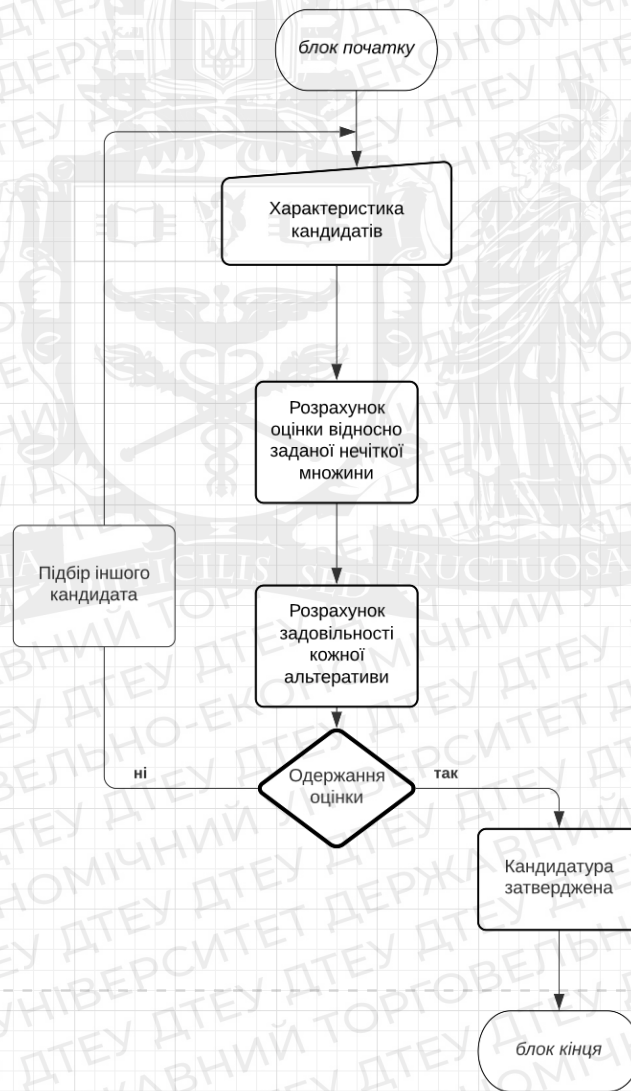


Рис.2.1 Схеми оцінки професійної діяльності працівників торгових підприємств



### **3.3 Характеристика програмного забезпечення системи оцінки професійної діяльності працівників торгових підприємств**

Кожна розробка програмного продукту починається з архітектури а саме: архітектура бази даних та архітектура програмного засобу. Архітектура бази даних включає в себе вибір бази даних та саму структуру даних. Для реалізації програмного забезпечення системи оцінки кандидатів було обрано систему управління базами даних, яка входить до складу пакету офісних програм Microsoft Office – Microsoft Access. Основними компонентами даної програми є конструктори таблиць, екранних форм, запитів SQL (мова структурованих запитів) та конструктор звітів, що виводиться на друк.

Програма Access сумісна з іншими джерелами даних, що дозволяє будувати повноцінні клієнт-серверні додатки. Отже, в середовищі програми Access були створені наступні таблиці, які виконують роль бази даних для даного програмного продукту:

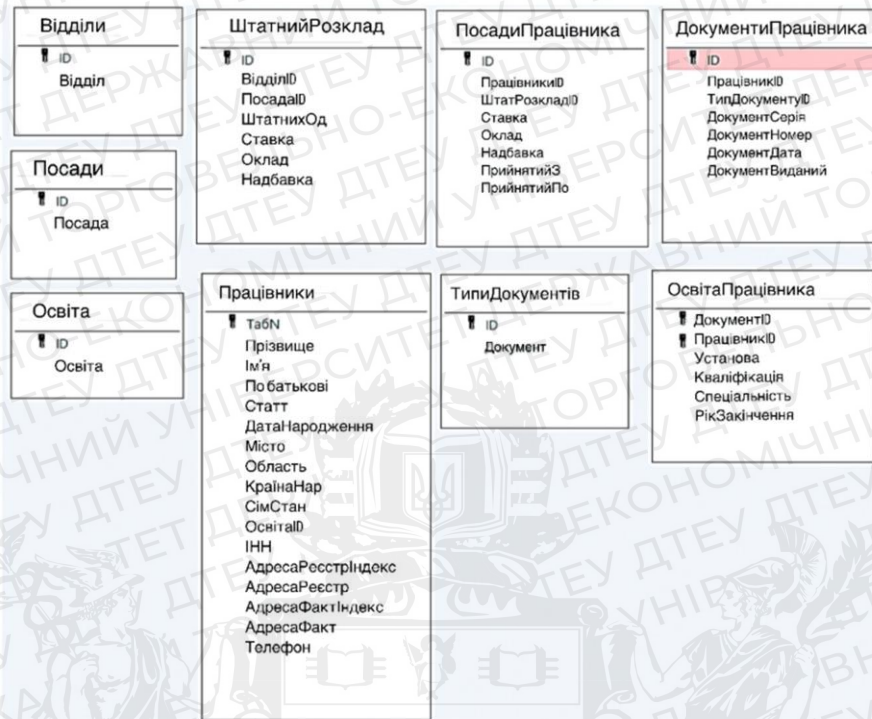


Рис.3.1 Таблиці бази даних

Таблиця “Відділи” містить в собі інформацію про відділи, які існують на підприємств. В таблиці “Посада Працівника” записані дані про посаду кожного працівника підприємства. В таблиці “Працівники” розміщені особисті дані кожного працівника. В таблиці “Освіта Працівника” знаходиться інформація про рівень освіти та кваліфікацію.

При прийнятті рішень в нечітких умовах, важливо враховувати неоднорідність та несхожість даних, а також нечіткість та невизначеність, які часто притаманні реальним ситуаціям. Для цього використовуються різні методи та підходи, такі як нечіткі множини, нечіткі відношення, нечіткі логічні системи та інші. Ці методи дозволяють враховувати та обробляти нечітку інформацію, що дозволяє зробити більш обґрунтовані та зрозумілі рішення.



Нечіткі множини дозволяють виразити нечіткість та невизначеність в описі альтернатив та критеріїв, а також установлювати нечіткі відношення між ними. Нечіткі відношення допомагають встановити ступінь відповідності альтернатив певним критеріям та обмеженням, враховуючи їх нечіткість. Нечіткі логічні системи використовуються для моделювання нечіткого мислення та прийняття рішень на основі нечіткої інформації.

Ці методи та підходи дозволяють ефективно працювати з нечіткими умовами та робити рішення, які враховують нестабільність, невизначеність та нечіткість даних. Вони знаходять широке застосування в різних сферах, таких як економіка, управління, штучний інтелект, системи підтримки прийняття рішень та багато інших.

Щодо технічних вимог до користувачі програми, то на клієнтській машині повинна бути встановлена операційна система Windows. Інших обмежень не має.

На початку роботи користувач повинен пройти процес авторизації, який полягає у введенні паролю та логіну(рис.3.2).

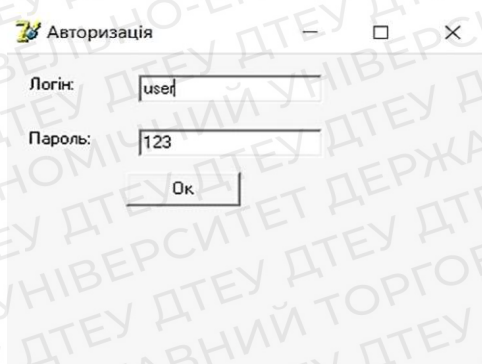


Рис.3.2 Вікно авторизації

У разі не правильно введених даних на екран користувача буде виведено повідомлення про помилку.

Для розрахунку оцінки кандидатів на вакантну посаду в головному вікні необхідно обрати вкладку “Оцінка кандидатів” після чого з’явиться наступне вікно(рис.3.3)

Рис.3.3 Вікно вводу параметрів кандидата на посаду

Наступним кроком для оцінки кандидата є введення його особистих даних у відповідні рядки вікна. Далі буде проведено розрахунок персональної оцінки кандидата. Для порівняння кандидатів та вибору найбільш відповідного необхідно здійснити аналіз шляхом натиснення відповідної клавіші у даному вікні.(рис.3.4)

Кандидат	Оцінка	Освіта	Кваліфікація
Петренко Валентина Ігорівна	0,7956	Вища	0,7
Івашенко Ангеліна Юріївна	0,8841	Технічна	0,4
Васеленко Станіслав Сергійович	0,9567	Середня	0,7

**Аналіз**

Висновок:  
 Кандидат Петренко Валентина Ігорівна отримав оцінку 0,7956;  
 Кандидат Івашенко Ангеліна Юріївна отримав оцінку 0,8841;  
 Кандидат Васеленко Станіслав Сергійович отримав оцінку 0,9567;  
 Найкращим кандидатом на посаду можна вважати Васеленко Станіслав Сергійович, який отримав оцінку 0,9567, яка є максимальною

Експорт до Word



Рис.3.4. Вікно вводу параметрів кандидата на посаду

Отриманий результат користувач має змогу експортувати до Word для надання звіту. Після підписання договору з обраним кандидатом його дані будуть внесені до особистої картки співробітника та збережені у базі даних(рис.3.5).

Картка співробітника

Основні дані | Додаткові дані | Посада | Документи | Освіта

Таб№: \_\_\_\_\_ Пол:  Ж  Ч

Прізвище: \_\_\_\_\_ Ім'я: \_\_\_\_\_ По батькові: \_\_\_\_\_

Дата народження: \_\_\_\_\_ Місце народження

Країна: \_\_\_\_\_

Сімейний стан: \_\_\_\_\_ Область: \_\_\_\_\_

Освіта: \_\_\_\_\_ Місто: \_\_\_\_\_

OK

Рис.3.5 Картка співробітника

Таким чином, розроблена і реалізована за допомогою сучасних програмних засобів автоматизована система оцінки професійної діяльності працівників торгових підприємств

проста у використанні, має інтуїтивно зрозумілий інтерфейс, забезпечує інтерактивний діалог користувача системи в процесі її використання, не потребує спеціалізованої підготовки користувачів та значних витрат на її впровадження в систему.

## ВИСНОВКИ

У випускній кваліфікаційній роботі представлені результати теоретичних та прикладних досліджень, що полягають у розробці інформаційної технології оцінки професійної діяльності працівників

торгових підприємств. Результати прикладних досліджень стали основою для створення автоматизованої системи оцінювання критеріїв претендентів на вакантну посаду. В результаті проведеного дослідження було зроблено такі висновки:

- від забезпечення підприємства кваліфікованими кадрами, раціонального їх використання залежать організаційно-технічний рівень виробництва, імідж підприємства, його фінансова стабільність, усі показники діяльності підприємства і важливим кроком у процесі забезпечення чи фірми організаційної системи відповідними її профілю людськими ресурсами є відбір кандидатів, що являє собою вибір із усієї підбраної групи індивідуальних працівників для наступного найму.
- при оцінці професійної діяльності працівників торгових підприємств доцільно використовувати сучасні інформаційні технології, що дозволить приймати адекватні та найбільш вигідні для підприємства рішення.
- досліджено метод визначення вагових коефіцієнтів та методи нелогічного виводу, що стали основою інформаційної технології оцінки професійної діяльності працівників торгових підприємств.
- здійснено програмну реалізацію методів вагових коефіцієнтів та нелогічного виводу у вигляді автоматизованої системи оцінки професійної діяльності працівників торгових підприємств.



## СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Алтунин А. Е., Семухин М. В. Модели и алгоритмы принятия решений в нечетких условиях: Монография. – Тюмень: Издательство Тюменского государственного университета, 2000. 352 с.
2. Андрейчиков А. В., Андрейчикова О. В. Анализ, синтез, планирование решений в экономике, Москва “Финансы и Статистика”-2000, с. 100, с.368.
3. Беллман Р., Заде Л. (1976). Принятие решений в расплывчатых условиях - в книге Вопросы анализа и принятия решений (ru). Москва: Мир. с. 172–215.
4. Березін Ф. Б. Методика багатостороннього дослідження особистості.
5. / Ф. Б. Березін, М. П. Мірошников, Е. Д. Соколова -- М.: Фолиум, 2004. с. 79.
6. Блюмин С. Л., Шуйкова И. А. Модели и методы принятия решений в условиях неопределенности. – Липецк: ЛЭГИ, 2001. 138 с.
7. Блюмин С. Л., Шуйкова И. А., Сараев П. В, Черпаков И. В. Нечеткая логика: алгебраические основы и приложения: Монография – Липецк: ЛЭГИ, 2002. 113 с.
8. Бойко Т. Г. Огляд методів визначення вагових коефіцієнтів показників властивостей продукції, ISSN 1993-9981 Методи та прилади контролю якості, № 24, 2010, с.87.
9. Бочарников В. П. Fuzzy-технология: Математические основы. Практика моделирования в экономике. – СПб: “Наука”, РАН, 2001. 328 с.
10. Варжапетян А. Г. Имитационное моделирование на GPSS/H: монография / А. Г. Варжапетян. – М.: Вузовская книга, 2004. – 340 с.
11. Василенко О. В. Проблема підбору управлінських кадрів та використання інноваційних технологій управління колективом у ресторанному господарстві при підготовці фахівців ресторанної справи

- [Електронний ресурс]: ЦДПУ, Київ, 2014. – Режим доступу:  
<https://www.cuspu.edu.ua/ю>. - Загл. з екрану (10.05.2020).
12. Войтко С. В., Мельниченко А. А. Управління трудовими ресурсами/  
Войтко С. В - ТОВ НВП 'Інтерсервіс' Київ, 2016 – 35 с.
  13. Гайнуллова Т. Використання поліграфа (детектора брехні) при роботі з  
персоналом / Т. Гайнуллова // Управління персоналом, 2001.. -№3. с.6.
  14. Дзюба Наталія Методи оцінки професійної діяльності працівників  
торгових підприємств [Електронний ресурс].Режим доступу:  
<https://www.trn.ua/articles/11242/> **HYPERLINK**  
"<https://www.trn.ua/articles/11242/>" - Загл. з екрану (10.05.2020).
  15. Дмитренко та ін. – К. : МАУП, 2004. – 368 с. 3. Данюк В. М.  
Менеджмент персоналу : навч. посіб. / В.М. Данюк, В. М. Петюх, С. О.  
Цимбалюк та ін. ; за заг. ред. В. М. Данюка, В. М. Петюха. – К. : КНЕУ,  
2004. 398 с.
  16. Дугіна О. Метод центру оцінки (Assessment-Center). Місце оцінки  
персоналу в кадровій роботі / О. Дугіна // Кадровий вісник, 2004. -№ 2  
(14) . - с. 24.
  17. Заде Л. Понятие лингвистической переменной и его применение к  
принятию приближенных решений: Пер. с англ. — М.: Мир, 1976. —  
165 с.
  18. Зайченко Ю. П. Нечёткие модели и методы в интеллектуальных  
системах: учеб. пособие / Ю. П. Зайченко. – К.: Изд. дом “Слово”, 2008.  
– 354с.
  19. Ивин А.А. Модальные теории Я. Лукасевича, 2001, с.19.
  20. Kendall M.G. Rank Correlation Methods / M. G. Kendall. – Н.У.: Hafner  
Publ. Co.,1995. – 196 p. (DOI: 10.1111/j.2044-8317.1956.tb00172.x).



21. Ковальчук В. М. Особливості розв'язання багатокритеріальних задач прийняття рішень у нечіткому середовищі. - Наукові записки. Серія "Економіка". Випуск 14, 2010. 449 - 455 с.
22. Ковтун Н. В. Теорія статистики: курс лекцій, практикум / Н. В. Ковтун. – К.: ІМЕКС. – 2007. – 276 с.
23. Кравець П. Системи прийняття рішень з нечіткою логікою / П. Кравець, Р. Киркало // Національний університет "Львівська політехніка". – 2009. с. 115–123.
24. Круглов В. В. Нечёткая логика и искусственные нейронные сети. Учеб. пособие / В. В. Круглов, М. И.
25. Кузьмін О. Є., Громовик Б. П. Менеджмент у фармації. - НОВА КНИГА, 2005. 335 с.
26. Ладанюк А. П. Основи системного аналізу, Вінниця - 2004, с.76-78.
27. Ліхоузова Т. А. Теорія імовірностей та математична статистика. Курс лекцій.: КПП ім. Ігоря Сікорського, 2018. – 182с.
28. Лук'янченко Н. Д. Керування трудом на промислових підприємствах. / Н. Д. Лук'янченко – Донецьк: Донбас, 2000. – 196 с.
29. Ляшенко О. І. Компетентність як об'єкт оцінювання навчальних досягнень учнів / О. І. Ляшенко // Збірник наукових праць Кам'янець-Подільського національного університету ім. Івана Огієнка. Серія: Педагогічна. – 2014. – Вип. 20. – С. 36–39. 12.
30. Медиковский М. О., Шуневич О. Б. Дослідження ефективності методів визначення вагових коефіцієнтів. Вісник Хмельницького національного університету №5, 2011. — С. 120-123.
31. Новосад В. П., Селіверстов Р. Г. – К. Методологія експертного оцінювання: конспект лекцій – К.: Вид-во НАДУ, 2008. – 48 с.

- 32.Одегов Ю. Г., Ніконова Т. В. Управління персоналом. Практикум: конкретні ситуації. – М.: Видавництво «Іспит», 2003. – 192 с.
- 33.Семенова Н. В., Колечкіна Л. М., Нагірна А. М. Багатокритеріальні задачі лексикографічної оптимізації на нечіткій множині альтернатив: УДК 519.8, 2010. 42 с.
- 34.Сеньо П. С. Теорія ймовірностей та математична статистика: підруч. / П. С. Сеньо. – [2-ге вид., перероб. і доп.]. – К.: Знання, 2007. – 556 с.
- 35.Сявавко М. С., Рибицька О. М. Математичне моделювання за умов невизначеності. – Львів: Українські технології, 2000. 320 с.
- 36.Федулова І.Л., Сокирник І.В., Стадник В.В., Йохна М.А., Новикова О.С., Рясних Є.Г. Менеджмент організацій : підручник – К. : Либідь, 2004. 448 с.
- 37.Черняк О. І. Теорія ймовірностей та математична статистика.
- 38.Збірник задач: навч. посіб. / О. І. Черняк, О. М. Обушна, А. В. Ставицький. – [2-ге вид.]. – К.: Знання, КІЩ, 2002. – 199 с.
- 39.Шарапов О. Д., Дербенцев В. Д., Семьонов Д. Є. Економічна кібернетика: Навч. посібник. – К.: КНЕУ, 2004. 231 с.
- 40.Шахмалов Ф. И. Американский менеджмент. Теория и практика / Ф. И. Шахмалов. – М. : Наука, 2002. – 272 с.
- 41.Под ред. Р.Р. Ягера. М. Нечеткие множества и теория возможностей. Последние достижения.: Радио и связь, 1986. 408 с.

## ДОДАТКИ



Програмний код реалізації оцінки професійної діяльності працівників  
торгових підприємств.

unit

Unit1;

interface

uses

Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls,  
Forms, Dialogs, StdCtrls, Unit2, Unit3;

type

TForm1 =

class(TForm) Edit1:

TEdit;

Edit2: TEdit;

Button1:

TButton; Label1:

TLabel; Label2:

TLabel;

procedure Button1Click(Sender:

TObject); private

{ Private declarations

} public

{ Public declarations

} end;

var

Form1:

TForm1;

implementation

```
{SR *.dfm}
```

```
procedure TForm1.Button1Click(Sender:  
TObject); var Form2:Tform2;  
Form3:TForm  
3; begin  
if (Edit1.Text='user') and (Edit2.Text='123') then  
begin  
Form2:=TForm2.Create(Self  
); Form2.ShowModal;  
Close  
e;  
end  
else  
begin  
n  
Form3:=TForm3.Create(Self  
); Form3.ShowModal;  
en  
d;  
en  
d;  
en  
d.  
nit Unit2;  
interface  
uses
```



Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,  
Dialogs, Menus, ExtCtrls, DB, ADODB, ComCtrls, Buttons, DBCtrls,  
Grids, unit4, DBGrids, unit5, unit6, unit7, StdCtrls, Unit8, Unit12, Unit10, Unit11,  
Unit13, Unit15;  
type

TForm2 = class(TForm)

MainMenu1:

TMainMenu; N1:

TMenuItem;

N2:

TMenuItem;

N3:

TMenuItem;

N5:

TMenuItem;

N6:

TMenuItem;

N7:

TMenuItem;

N9:

TMenuItem;

N10: TMenuItem;

N11:

TMenuItem;

N12:

TMenuItem; N4:

TMenuItem;

TreeView1: TTreeView;

ADOTable1: TADOTable;

Panel2: TPanel;

SpeedButton1:

TSpeedButton;

SpeedButton2:

TSpeedButton;

SpeedButton3:

TSpeedButton;

SpeedButton4:

TSpeedButton;

SpeedButton5:

TSpeedButton;

SpeedButton6:

TSpeedButton;

SpeedButton7:

TSpeedButton; DataSource1:

TDataSource; DBGrid1:

TDBGrid; ADOQuery1:

TADOQuery;

DBNavigator2:

TDBNavigator;

SpeedButton8: TSpeedButton;

N8: TMenuItem;

ADOTable2: TADOTable;



```
procedure N5Click(Sender: TObject);
procedure FormCreate(Sender:
TObject);
procedure TreeView1Change(Sender: TObject; Node:
TTreeNode); procedure N6Click(Sender: TObject);
procedure N7Click(Sender: TObject);
procedure N9Click(Sender: TObject);
procedure N10Click(Sender:
TObject);
procedure SpeedButton6Click(Sender: TObject);
procedure N11Click(Sender:
TObject); procedure
N12Click(Sender: TObject);

procedure N4Click(Sender: TObject);
procedure N8Click(Sender: TObject);
procedure N13Click(Sender: TObject);
private
{ Private declarations
} public
{ Public declarations
} end;
var
Form2:
TForm2;
i:TTreeNode;
```

```

t:string;
q:integer;
implementation
n uses unit1;
{$R *.dfm}
procedure TForm2.N5Click(Sender:
TObject); begin
Form1.Clos
e;
Form2.Clos
e; end;
procedure TForm2.TreeView1Change(Sender: TObject; Node:
TTreeNode); begin
q:=node.AbsoluteInde
x; i:= Node;
t:=TreeView1.Items.Item[q].Tex
t; ADOQuery1.SQL.Clear;
ADOQuery1.SQL.Add('select          Працівники.ТабН,
Працівники.Прізвище, Працівники.Ім'я,
Працівники.Побатькові,
Посади.Посада,Освіта.Освіта,Працівники.Телефон');
ADOQuery1.SQL.Add('from          Відділи,
Працівники, Посади,Освіта,
ПосадиПрацівника, ШтатнийРозклад'); ADOQuery1.SQL.Add('where
Працівники.ОсвітаID =Освіта.ID and');
ADOQuery1.SQL.Add('Працівники.ТабН = Посади Працівника. ПрацівникID
and');

```



```

ADOQuery1.SQL.Add(' ПосадиПрацівника.ШтатРозкладID
                    = ШтатнийРозклад.ID and');
ADOQuery1.SQL.Add(' ШтатнийРозклад.ПосадаID = Посади.ID
and' );
ADOQuery1.SQL.Add(' ШтатнийРозклад'.ВідділID = Відділи.ID and');
ADOQuery1.SQL.Add(' Відділи.Відділ = '"+t+"'");
ADOQuery1.Active:=True;
end;
procedure TForm2.FormCreate(Sender:
TObject); begin
While not ADOTable1.Eof
do begin
TreeView1.Items.Add(i,ADOTable1.FieldByName('Відділ').AsString
); ADOTable1.Next;
en
d;
en
d;
procedure TForm2.N6Click(Sender:
TObject); var Form4:Tform4;
begin
Form4:=TForm4.Create(Self
); Form4.ShowModal;
end;
procedure TForm2.N7Click(Sender:
TObject); var Form5:Tform5;

```

begin

```
Form5:=TForm5.Create(Self);  
Form5.ShowModal;
```

end;

```
procedure TForm2.N9Click(Sender:
```

```
TObject); var Form7:Tform7;
```

begin

```
Form7:=TForm7.Create(Self);  
Form7.ShowModal;
```

end;

```
procedure TForm2.N10Click(Sender:
```

```
TObject); var Form6:Tform6;
```

begin

```
Form6:=TForm6.Create(Self);  
Form6.ShowModal;
```

end;

```
procedure TForm2.SpeedButton6Click(Sender:
```

```
TObject); var Form8:Tform8;
```

begin

```
Form8:=TForm8.Create(Self);  
Form8.ShowModal;
```

end;

```
procedure TForm2.N11Click(Sender:
```

```
TObject); var Form11:Tform11;
```

begin

```
Form11:=TForm11.Create(Self);  
Form11.ShowModal;
```



```
end;  
procedure TForm2.N12Click(Sender:  
TObject); var Form10:Tform10;
```

```
begin  
Form10:=TForm10.Create(Self  
); Form10.ShowModal;
```

```
end;  
procedure TForm2.N4Click(Sender:  
TObject); var Form12:Tform12;
```

```
begin  
Form12:=TForm12.Create(Self  
); Form12.ShowModal;
```

```
end;  
procedure TForm2.N8Click(Sender:  
TObject); var Form13:Tform13;
```

```
begin  
Form13:=TForm13.Create(Self  
); Form13.ShowModal;
```

```
end;  
procedure TForm2.N13Click(Sender:  
TObject); var Form15:Tform15;
```

```
begin  
Form15:=TForm15.Create(Self  
); Form15.ShowModal;
```

```
en  
d;
```

```
en
d.
unit
Unit3;
interface
uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls,
  Forms, Dialogs, StdCtrls;
type
```

```
TForm3 = class(TForm)
```

```
  Label1:
```

```
  TLabel; private
```

```
  { Private declarations
```

```
  } public
```

```
  { Public declarations
```

```
  } end;
```

```
var
```

```
  Form3:
```

```
  TForm3;
```

```
implementation
```

```
  {$R *.dfm}
```

```
end.
```

```
unit
```

```
Unit4;
```

```
interface
```

```
uses
```



Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls,  
Forms, Dialogs, ExtCtrls, DBCtrls, Grids, DBGrids, DB, ADODB;  
type

```
TForm4 = class(TForm)  
  ADOTable1: TADOTable;  
  DataSource1: TDataSource;  
  DBGrid1: TDBGrid;  
  DBNavigator1: TDBNavigator;
```

```
private  
  { Private declarations  
} public  
  { Public declarations  
} end;
```

```
var  
  Form4:
```

```
TForm4;
```

```
implementation
```

```
{ $R *.dfm }
```

```
end.
```

```
unit
```

```
Unit5;
```

```
interface
```

```
uses
```

Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls,  
Forms, Dialogs, DB, ADODB, Grids, DBGrids, ExtCtrls, DBCtrls;  
type

```
TForm5 = class(TForm)
DataSource1: TDataSource;
ADOTable1: TADOTable;
DBGrid2: TDBGrid;
DBNavigator2: TDBNavigator;
ADOTable2: TADOTable;
DataSource2: TDataSource;
private
  { Private declarations }
public
  { Public declarations }
end;
var
  Form5:
TForm5;
implementation
  {SR *.dfm}
end.
unit
Unit6;
interface
uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
  Dialogs, DB, ADOdb, ExtCtrls, DBCtrls, Grids, DBGrids, StdCtrls, ComCtrls,
  ComObj;
type
```



```
TForm6 = class(TForm)
  DBGrid1: TDBGrid;
  DBNavigator1:
  TDBNavigator; ADOTable1:
  TADOTable; DataSource1:
  TDataSource; Button1:
  TButton;
  Button2: TButton;
  ADOTable2: TADOTable;
  TreeView1: TTreeView;
  ADOQuery1:
  TADOQuery;
  procedure FormCreate(Sender: TObject);
  procedure TreeView1Change(Sender: TObject; Node:
  TTreeNode); procedure Button1Click(Sender: TObject);
  procedure Button2Click(Sender:
  TObject); private
  { Private declarations
  } public
  { Public declarations
  } end;
var
  Form6:
  TForm6;
  i:TTreeNode;
  t:string;
  q:integer;
```

```

ExcelApplication:Variant; implementation
uses unit2;

{$R *.dfm}
procedure TForm6.FormCreate(Sender:
TObject); begin
While not ADOTable1.Eof
do begin
TreeView1.Items.Add(i,ADOTable1.FieldByName('Відділ').AsString
); ADOTable1.Next;
en
d;
en
d;
procedure TForm6.TreeView1Change(Sender: TObject; Node:
TTreeNode); begin
q:=node.AbsoluteIndex; i:= Node;
t:=TreeView1.Items.Item[q].Text; ADOQuery1.SQL.Clear;
ADOQuery1.SQL.Add('select ШтатнийРозклад.ID,Посади.Посада,
ШтатнийРозклад.ШтатнихОд, ШтатнийРозклад.Ставка,
ШтатнийРозклад.Оклад, ШтатнийРозклад.Надбавка, Відділи.Відділ');
ADOQuery1.SQL.Add('from ШтатнийРозклад, Відділи, Посади ');
ADOQuery1.SQL.Add(' Where ШтатноеРасписание.ПосадаID
= Посади.ID and' );

```



```
ADOQuery1.SQL.Add(' ШтатнийРозклад.ВідділID = Відділи.ID and');
ADOQuery1.SQL.Add(' Відділи.Відділ = '"+t+"");
ADOQuery1.Active:=True;
end;
procedure TForm6.Button1Click(Sender:
TObject); begin
close;
end;
procedure TForm6.Button2Click(Sender:
TObject); var Path, FileName : String;
Excel:
Variant;
i:integer;
begin
Path := ExtractFilePath(ParamStr(0));
Excel:=CreateOleObject('Excel.Application');
Excel.Application.WorkBooks.Add( ""+Path +
'Книга1.xlsx'); Excel.Visible:=True;
Excel.DisplayAlerts:=False;
Excel.WorkSheets.Item['Лист1'].Cells[2, 1]:='ID';
Excel.ActiveWorkbook.SaveAs('Книга1.xlsx');
Excel.WorkSheets.Item['Лист1'].Cells[2, 2]:='Посада';
Excel.ActiveWorkbook.SaveAs('Книга1.xlsx');
Excel.WorkSheets.Item['Лист1'].Cells[2, 3]:='Штатних
Од'; Excel.ActiveWorkbook.SaveAs('Книга1.xlsx');
Excel.WorkSheets.Item['Лист1'].Cells[2, 4]:='Оклад';
Excel.ActiveWorkbook.SaveAs('Книга1.xlsx');
```

```
Excel.WorkSheets.Item['Лист1'].Cells[2, 5]='Надбавка';
Excel.ActiveWorkbook.SaveAs('Книга1.xlsx');
Excel.WorkSheets.Item['Лист1'].Cells[2, 6]='Відділ';
Excel.ActiveWorkbook.SaveAs('Книга1.xlsx');
ADOQuery1.First;
i:=3;
  while not ADOQuery1.Eof
    do begin
Excel.WorkSheets.Item['Лист1'].Cells[i,
1]:=ADOQuery1.FieldByName('ID').AsInteger
;
Excel.ActiveWorkbook.SaveAs('Книга1.xlsx')
;

Excel.WorkSheets.Item['Лист1'].Cells[i,
2]:=ADOQuery1.FieldByName('Посада').AsString;
Excel.ActiveWorkbook.SaveAs('Книга1.xlsx');
Excel.WorkSheets.Item['Лист1'].Cells[i,
3]:=ADOQuery1.FieldByName('ШтатнихОд').AsStrin
g; Excel.ActiveWorkbook.SaveAs('Книга1.xlsx');
Excel.WorkSheets.Item['Лист1'].Cells[i,
4]:=ADOQuery1.FieldByName('Оклад').AsString;
Excel.ActiveWorkbook.SaveAs('Книга1.xlsx');
Excel.WorkSheets.Item['Лист1'].Cells[i,
5]:=ADOQuery1.FieldByName('Надбавка').AsString;
Excel.ActiveWorkbook.SaveAs('Книга1.xlsx');
Excel.WorkSheets.Item['Лист1'].Cells[i,
```



```

6]:=ADOQuery1.FieldByName('Відділ').AsString;
Excel.ActiveWorkbook.SaveAs('Книга1.xlsx');
i:=i+1;
    ADOQuery1.Next;
end;
end;
end;
end;
unit Unit7;
interface
uses
    Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls,
    Forms, Dialogs, DB, ADODB, ExtCtrls, DBCtrls, Grids, DBGrids;
type
    TForm7 = class(TForm)
        DBGrid1: TDBGrid;
        DBNavigator1:
            TDBNavigator;
        DataSource1: TDataSource;

        ADOTable1:
            TADOTable; private
        { Private declarations }
        } public
        { Public declarations }
    } end;
var

```

Form7:

TForm7;

implementation

{SR \*.dfm}

end.

unit Unit8;

interface

uses

Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,

Dialogs, ADOdb, StdCtrls, DB, ExtCtrls, DBCtrls, Grids,

DBGrids type

TForm8 = class(TForm)

  ComboBox1:

  TComboBox; Label1:

  TLabel; ADOQuery1:

  TADOQuery; DBGrid1:

  TDBGrid;

  DBNavigator1:

  TDBNavigator; DataSource1:

  TDataSource; Button1:

  TButton; ADOTable1:

  TADOTable;

  procedure FormCreate(Sender: TObject);

  procedure Button1Click(Sender: TObject);

private

  { Private declarations }

public



```

    { Public declarations
  } end;
var
  Form8:
  TForm8;
implementation
  {SR *.dfm}

  procedure TForm8.FormCreate(Sender:
  TObject); begin
    ADOTable1.First;

    While not ADOTable1.Eof
    do begin
      Combobox1.Items.Add(ADOTable1.FieldByName('Посада').AsString);
      ADOTable1.Next;
    end;
  end;

  procedure TForm8.Button1Click(Sender:
  TObject); var t:string;
  begin
    t:=Combobox1.Text;
    ADOQuery1.SQL.Clear
  ;
    ADOQuery1.SQL.Add('select

```

Працівники. ТабN,Працівники.Прізви

ище, Працівники.Ім'я,

Працівники.По

батькові,

Посади.Посада,Освіта.Освіта, Працівники.Телефон');

ADOQuery1.SQL.Add('from Відділи,

Працівники,

Посади,Освіта,ПосадиПрацівника, ШтатнийРозклад');

ADOQuery1.SQL.Add('where Працівники.ОсвітаID =Освіта.ID and');

ADOQuery1.SQL.Add('Работники.ТабN = Посади Працівника. ПрацівникID  
and');

ADOQuery1.SQL.Add(' Посади Працівника.ШтатРозкладID  
= ШтатнийРозклад.ID and');

ADOQuery1.SQL.Add(' ШтатнийРозклад.ПосадаID = Посади.ID  
and');

ADOQuery1.SQL.Add(' ШтатнийРозклад.ВідділID = Відділи.ID and');

ADOQuery1.SQL.Add(' Посади.Посада = '"+t+"'');

ADOQuery1.Active:=True;

en

d;

en

d.

unit

Unit9;

interface

uses

Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,

Dialogs, StdCtrls, ExtCtrls, DBCtrls, DB, ADODB, Grids,

DBGrids, ComCtrls,ComObj;



type

TForm9 = class(TForm)

DBGrid1: TDBGrid;

DataSource1: TDataSource;

DBNavigator1: TDBNavigator;

ADOTable1: TADOTable;

private

{ Private declarations

} public

{ Public declarations

} end;

var

Form9: TForm9;

ExcelApplication: Varian

t;

implementation

{SR \*.dfm

unit

Unit10;

interface

uses

Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls,

Forms, Dialogs, DB, ADODB, StdCtrls, ComCtrls, Grids, Buttons,

DBGrids, ExtCtrls, DBCtrls;

type

TForm10 = class(TForm)

PageControl1:

TPageControl; TabSheet1:

TTabSheet; TabSheet2:

TTabSheet; TabSheet3:

TTabSheet; TabSheet4:

TTabSheet; TabSheet5:

TTabSheet; Edit1: TEdit;

Label1: TLabel;

RadioButton1:

TRadioButton;

RadioButton2:

TRadioButton; Label2:

TLabel;

Label3: TLabel;

Label4: TLabel;

Label5: TLabel;

Edit2: TEdit;

Edit3: TEdit;

Edit4: TEdit;

Label6: TLabel;

ComboBox1: TComboBox;

Label7: TLabel;

ComboBox2:

TComboBox;

GroupBox1:

TGroupBox; Label8:

TLabel;



Label9: TLabel;

Label11:

TLabel; Edit5:

TEdit; Edit6:

TEdit; Edit7:

TEdit;

ADOTable1:

TADOTable; Label10:

TLabel;

Edit8: TEdit;

GroupBox2:

TGroupBox; Label12:

TLabel; Label14:

TLabel;

Edit9: TEdit;

Edit10:

TEdit;

GroupBox3:

TGroupBox; Label13:

TLabel; Label15:

TLabel;

Edit11: TEdit;

Edit12: TEdit;

Label16: TLabel;

Edit13: TEdit;

Label17: TLabel;

ComboBox3:

TComboBox;

ADOTable2: TADOTable;

Label18: TLabel;

Edit14: TEdit;

Label19:

TLabel; Edit15:

TEdit;

Label20: TLabel;

Edit16: TEdit;

Label21:

TLabel; Edit17:

TEdit; Label22:

TLabel; Edit18:

TEdit; Label23:

TLabel; Edit19:

TEdit; Label24:

TLabel;

Label25:

TLabel;

Label26:

TLabel; Edit20:

TEdit; Label27:

TLabel;

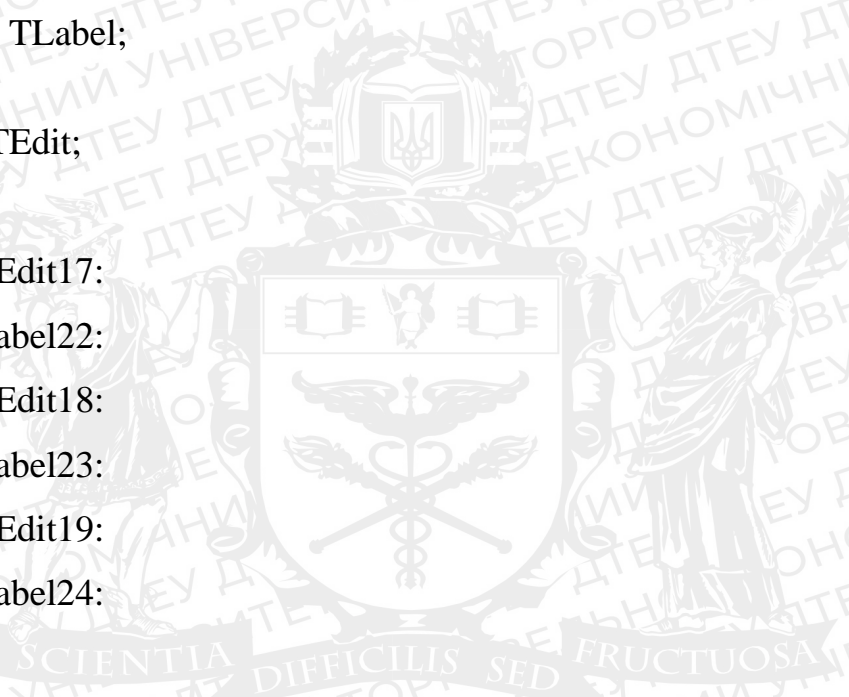
Label28:

TLabel; Edit21:

TEdit; Label29:

TLabel; Edit22:

TEdit; Label30:





TLabel; Edit23:

TEdit;

ADOTable3:

TADOTable; Label31:

TLabel; ComboBox4:

TComboBox; Label32:

TLabel;

Edit24: TEdit;

Label33:

TLabel; Edit25:

TEdit; Label34:

TLabel; Edit26:

TEdit; Label35:

TLabel; Edit27:

TEdit;

Button1: TButton;

BitBtn1: TBitBtn;

BitBtn2: TBitBtn;

ADOTable4:

TADOTable;

DataSource1:

TDataSource; DBGrid1:

TDBGrid; ADOTable5:

TADOTable; Label36:

TLabel;

Edit28: TEdit;

ADOTable6:

```
TADOTable; Label37:
TLabel;
Edit29: TEdit;
ADOTable7:
TADOTable;
ADOTable8:
TADOTable;
DBNavigator1:
TDBNavigator; ADOTable9:
TADOTable;
procedure FormCreate(Sender: TObject);
procedure BitBtn1Click(Sender: TObject);
procedure Button1Click(Sender: TObject);
procedure BitBtn2Click(Sender: TObject);
private
  { Private declarations
} public
  { Public declarations
} end;
var
  Form10:
TForm10;
implementation
  {$R *.dfm}
  procedure TForm10.FormCreate(Sender:
TObject); begin
```



```
ADOTable4.First;
While not ADOTable4.Eof
do begin
ADOTable4.Delet
e; end;
ADOTable9.First;
While not ADOTable9.Eof
do begin
Combobox1.Items.Add(ADOTable9.FieldByName('СімейнийСтан').AsStr ing);
ADOTable9.Nex
t; end;
ADOTable1.First
;
While not ADOTable1.Eof
do begin
Combobox2.Items.Add(ADOTable1.FieldByName('Освіта').AsString);
ADOTable1.Next;
end;
ADOTable3.Firs
t;
While not ADOTable3.Eof
do begin
Combobox4.Items.Add(ADOTable3.FieldByName('Документ').AsString
); ADOTable3.Next;
end;
ADOTable2.Firs
t;
```

```
While not ADOTable2.Eof
do begin
  Combobox3.Items.Add(inttostr(ADOTable2.FieldByName('ID').AsInteger)
); ADOTable2.Next;
end;

end;

procedure TForm10.BitBtn1Click(Sender:
TObject); begin
  ADOTable2.First;
  While not ADOTable2.Eof
  do begin
    if(Combobox3.Text=(inttostr(ADOTable2.FieldByName('ID').AsInteger)))then
    begin
      Edit16.Text:=floattostr(ADOTable2.FieldByName('Ставка').AsFloat);
      Edit17.Text:=inttostr(ADOTable2.FieldByName('Оклад').AsInteger);
      Edit18.Text:=inttostr(ADOTable2.FieldByName('Надбавка').AsInteger);
      Edit19.Text:=floattostr(ADOTable2.FieldByName('Ставка').AsFloat*ADOTable
      2
      .FieldByName('Оклад').AsInteger+ADOTable2.FieldByName('Надбавка').AsInt
      e ger);
    end;
  ADOTable2.Nex
  t; end;
end;

procedure TForm10.Button1Click(Sender:
TObject); begin
```



```

ADOTable4.Insert ;
ADOTable4.FieldName('Документ').AsString:=ComboBox4.Text
; ADOTable4.FieldName('Серія').AsString:=Edit24.Text;
ADOTable4.FieldName('Номер').AsInteger:=
strtoint(Edit25.Text);
ADOTable4.FieldName('Дата').AsString:=Edit26.Text;
ADOTable4.FieldName('Ким виданий').AsString:= Edit27.Text;
ADOTable4.Post ;
end;
procedure TForm10.BitBtn2Click(Sender:
TObject); var q,q1,q2:integer;

a:boolean;
n; begin
ADOTable9.First;
While not ADOTable9.Eof
do begin
if ADOTable9.FieldName('СімейнийСтан').AsString
=ComboBox1.Text then
q2:=
ADOTable9.FieldName('ID').Asinteger;
ADOTable9.Next;
end;
ADOTable1.Firs
t;
While not ADOTable1.Eof
do begin

```

```
if ADOTable1.FieldName(Освіта).AsString =ComboBox2.Text then q:=
ADOTable1.FieldName('ID').Asinteger;
ADOTable1.Nex
t; end;
if RadioButton1.Checked=True then
a:=False;
if RadioButton2.Checked=True then
a:=True;
ADOTable5.Insert;
ADOTable5.FieldName('ТабN').AsInteger:=strtoint(Edit1.Text) ;
ADOTable5.FieldName('Прізвище').AsString:=Edit2.Text;
ADOTable5.FieldName('Ім'я').AsString:= Edit3.Text;
ADOTable5.FieldName('По батькові').AsString:=Edit4.Text;
ADOTable5.FieldName('Стать').AsBoolean:= a;
ADOTable5.FieldName('ДатаНародження').AsString:=Edit28.T
ext ; ADOTable5.FieldName(МістоНар).AsString:=Edit7.Text;
ADOTable5.FieldName('Область').AsString:= Edit6.Text;
ADOTable5.FieldName('КраїнаНар').AsString:=Edit5.Text;
ADOTable5.FieldName('СімСтан').AsInteger:=q2;
ADOTable5.FieldName('ОсвітаID').AsInteger:=q;
ADOTable5.FieldName('ІНН').AsString:=Edit8.Text;
ADOTable5.FieldName('АдресаРеєстрІндекс').AsString:=Edit9.Text;
ADOTable5.FieldName('АдресаРеєстр').AsString:=Edit10.Text;
ADOTable5.FieldName('АдресаФактІндекс').AsString:=Edit11.Text;
ADOTable5.FieldName('АдресаФакт').AsString:=Edit12.Text;
ADOTable5.FieldName('Телефон').AsString:=Edit13.Text;
ADOTable5.Post ;
```



ADOTable6.Insert;

ADOTable6.FieldByName('ПрацівникID').AsInteger:=strtoint(Edit1.Text);  
ADOTable6.FieldByName('ДокументID').AsString:=Edit29.Text;

ADOTable6.FieldByName('Установа').AsString:= Edit21.Text;

ADOTable6.FieldByName('Кваліфікація').AsString:=Edit22.Text;

ADOTable6.FieldByName('Спеціальність').AsString:= Edit23.Text;

ADOTable6.FieldByName('РікЗакінчення').AsString:=Edit20.Text ;

ADOTable6.Post ;

ADOTable7.Insert ;

ADOTable7.FieldByName('ПрацівникID').AsInteger:=strtoint(Edit1.Text);

ADOTable7.FieldByName('ШтатРозкладID').AsInteger:=strtoint(ComboBox3.Text);

ADOTable7.FieldByName('Ставка').AsFloat:=strtofloat(Edit16.Text)

; ADOTable7.FieldByName('Оклад').AsFloat:= strtoint(Edit17.Text);

ADOTable7.FieldByName('Надбавка').AsFloat:=strtoint(Edit18.Text

); ADOTable7.FieldByName('ПрийнятийС').AsString:=Edit14.Text;

ADOTable7.FieldByName('ПрийнятийПо').AsString:= Edit15.Text;

ADOTable7.Post ;

ADOTable4.First;

While not ADOTable4.Eof do

begin

ADOTable3.Firs

t;

While not ADOTable3.Eof

do begin

if

ADOTable3.FieldName('Документ').AsString=ADOTable4.FieldName('Документ').AsString then

begin

q1:=

ADOTable3.FieldName('ID').AsInteger;

ADOTable8.Insert ;

ADOTable8.FieldName('ПрацівникID').AsInteger:=strtoint(Edit1.Text);

ADOTable8.FieldName('ТипДокументуID').AsInteger:=q1;

ADOTable8.FieldName('ДокументСерія').AsString:=ADOTable4.FieldName('Серія').AsString;

ADOTable8.FieldName('ДокументНомер').AsString:=ADOTable4.FieldName('Номер').AsString;

ADOTable8.FieldName('ДокументДата').AsString:=ADOTable4.FieldName('Дата').AsString;;

ADOTable8.FieldName('ДокументВиданий').AsString:=ADOTable4.FieldName('Ким виданий').AsString;

ADOTable8.Post ;

end;

ADOTable3.Next;

end;

ADOTable4.Next;

end;

ADOTable8.Insert ;

ADOTable8.FieldName('ПрацівникID').AsInteger:=strtoint(Edit1.Text);

ADOTable8.FieldName('ТипДокументаID').AsInteger:=strtoint(ComboBox3.Text);



```
ADOTable8.FieldName('ДокументСерія').AsInteger:=strtoint(Edit16.Text)
; ADOTable8.FieldName('ДокументНомер').AsFloat:=
strtoint(Edit17.Text);
ADOTable8.FieldName('ДокументДата').AsFloat:=strtoint(Edit18.Text);
ADOTable8.FieldName('ДокументВиданий').AsString:=Edit14.Text;
ADOTable8.Post ;
Clos
e;
end
unit
Unit11;
interface
uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
  Dialogs, ExtCtrls, DBCtrls, DB, ADODB, Grids, DBGrids, StdCtrls, Mask,
  Unit16;
type
  TForm11 = class(TForm)
    DBGrid1: TDBGrid;
    ADOTable3: TADOTable;
    DataSource1: TDataSource;
    DBNavigator1:
    TDBNavigator; Label1:
    TLabel;
    Label2: TLabel;
    DBEdit1:
```

TBEdit;  
DBEdit2:  
TBEdit;  
DBEdit3:  
TBEdit;  
DBEdit4:  
TBEdit;  
DBEdit5:  
TBEdit;  
DBEdit6:  
TBEdit;  
DBEdit7:  
TBEdit; Label3:  
TLabel;  
Label4: TLabel;  
Label5: TLabel;  
Label6: TLabel;  
Label7: TLabel;  
Label8: TLabel;  
Label9: TLabel;  
Button2:  
TButton;  
procedure Button2Click(Sender:  
TObject); private  
  { Private declarations  
  } public





```

    { Public declarations
    } end;
var
    Form11:
TForm11;
implementation
    {$R *.dfm}
    procedure TForm11.Button2Click(Sender:
TObject); var Form16:Tform16;
    begin
        Form16:=TForm16.Create(Self
    ); Form16.ShowModal;
    en
    d;
    en
    d.
    unit
    Unit12;
    interface
    uses
        Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls,
        Forms, Dialogs, ExtCtrls, StdCtrls, Grids, Spin, DB, ADODB, DBGrids,
        Mask,
        DBCtrls, Menus, ComObj, Math, jpeg; // стандартний набір модулів
    type
        TForm12 =
        class(TForm) Memo1:

```

TMemo; Button3:  
TButton; Panell1:  
TPanel; Label14:  
TLabel; Label15:  
TLabel; Label17:  
TLabel; Label11:  
TLabel; Label9:  
TLabel; Label18:  
TLabel;  
ComboBox2:  
TComboBox;  
ComboBox4:  
TComboBox; Button1:  
TButton;  
Button2: TButton;  
DBGrid1: TDBGrid;  
DataSource1:  
TDataSource; Edit1:  
TEdit;  
Edit2: TEdit;  
Edit3: TEdit;  
Edit4: TEdit;  
Label2:  
TLabel;  
Label5:  
TLabel;  
ADOTable1: TADOTable;  
Image1: TImage





```
procedure N1Click(Sender: TObject);
procedure Button1Click(Sender:
TObject); procedure FormCreate(Sender:
TObject); procedure
Button2Click(Sender: TObject);
procedure Button3Click(Sender:
TObject);
private
  { Private declarations
} public
  { Public declarations
} end;
Var
  Form12:
TForm12;
implementation
uses Unit11, Unit1, Unit2;
{$R *.dfm}
procedure TForm12.N1Click(Sender:
TObject); begin
  Form1.Close;
  Form2.Close;
  Form12.Clos
e; end;
procedure TForm12.Button1Click(Sender:
TObject); Var   min,res,sum,w:real;
```

```
q:array[1..5] of real ;
m:array[1..6] of real ;
e:array[0..10] of real ;
d:array[1..6,0..10] of real
; i,j,k:integer;
st:strin
g;
begin
Image1.Visible:=False
;
DBGrid1.Visible:=tru
e;
Button2.Visible:=true;
if ComboBox4.Text='Вища'
then q[1]:=0.9;
if ComboBox4.Text='Технічна' then
q[1]:=0.5;
if ComboBox4.Text='Середня' then
q[1]:=0.25;
if ComboBox4.Text='Немає'
then q[1]:=0.01;
if ComboBox2.Text='від 1 року до 5'
then q[2]:=0.25;
if ComboBox2.Text='від 5 до 10' then
q[2]:=0.5;
if ComboBox2.Text='більше 10'
then q[2]:=0.9;
```



```

if ComboBox2.Text='Hemac'
then q[2]:=0.01;
q[3]:=strtofloat(Edit1.Text
);
q[4]:=strtofloat(Edit2.Text
);
q[5]:=strtofloat(Edit3.Text
); min:=q[1];
for i:=2 to 3 do
if q[i]<min then
min:=q[i]; m[1]:=min;
if q[5]<min then m[4]:=q[5] else
m[4]:=min; if q[4]<min then m[2]:=q[4]
else m[2]:=min; for i:=4 to 5 do
if q[i]<min then
min:=q[i]; m[3]:=min;
if q[3]<(1-q[1]) then min:=q[3] else min:=1-
q[1]; if q[2]<min then min:=q[2];
if q[5]<min then
min:=q[5]; m[5]:=min;
if (1-q[3])<(1-q[2]) then m[6]:=1-q[3] else min:=1-q[2];

for i:=0 to 10 do
if (1-m[1]+(i/10))<1 then d[1,i]:=(1-m[1]+(i/10)) else d[1,i]:=1;
for i:=0 to 10 do
if (1-m[2]+sqrt(i/10))<1 then d[2,i]:=(1-m[2]+sqrt(i/10)) else d[2,i]:=1;
for i:=0 to 9 do
if (1-m[3])<1 then d[3,i]:=(1-m[3]) else d[3,i]:=1;

```

```

if (1-m[3]+1)<1 then d[3,10]:=(1-m[3]+1) else d[3,10]:=1;
for i:=0 to 10 do
if (1-m[4]+sqr(i/10))<1 then d[4,i]:=(1-m[4]+sqr(i/10)) else
d[4,i]:=1; for i:=0 to 10 do
if (1-m[5]+(i/10))<1 then d[5,i]:=(1-m[5]+(i/10)) else d[5,i]:=1;
for i:=0 to 10 do
if (1-m[6]+(1-(i/10)))<1 then d[6,i]:=(1-m[6]+(1-(i/10))) else d[6,i]:=1;
for j:=0 to 10
do begin
min:=d[1,j];
for i:=1 to 6 do
if d[i,j]<min then min:=d[i,j]
; e[j]:=min;
end;
w:=
0;
res:=0;
While w<>1
do begin
min:=1;
for j:=0 to 10 do
if (e[j]<min) and (e[j]>w) then
min:=e[j]; k:=1;
sum:=0;
for j:=0 to 10 do

```



```

if e[j]>=min then
begin
sum:=sum+e[j];
k:=k+1;
end;

res:=res+(sum/(k-1)*(min-
w)); w:=min;
end;
ADOTable1.Insert
t;
ADOTable1.FieldName('Кандидат').AsString:=Edit4.Text ;
ADOTable1.FieldName('Оцінка').AsString:=floattostr(RoundTo(res,-4));
ADOTable1.FieldName('Освіта').AsString:= ComboBox4.Text;
ADOTable1.FieldName('Кваліфікація').AsString:=Edit1.Text;
ADOTable1.FieldName('Вміння працювати з
ПЗ').AsString:=Edit2.Text;
ADOTable1.FieldName('Досвід').AsString:=ComboBox2.Text ;
ADOTable1.FieldName('Юридична грамотність').AsString:=Edit3.Text
; ADOTable1.Post ;
end;
procedure TForm12.FormCreate(Sender:
TObject); begin
ADOTable1.First;
while not ADOTable1.EOF
do begin
ADOTable1.Delet
e;
ADOTable1.Next;

```

```
end;
ADOTable1.First;
while not ADOTable1.EOF
do begin
ADOTable1.Delete;

ADOTable1.Next;
end;
end;
procedure TForm12.Button2Click(Sender:
TObject); var pib:string ;
max:real;
begin
Button3.Visible:=true;
Memo1.Visible:=true;
Memo1.Lines.Add('Висновок:');
); ADOTable1.First;
While not ADOTable1.Eof
do begin
Memo1.Lines.Add('Кандидат'+ADOTable1.FieldName('Кандидат').AsString
+ ' отримав оцінку '+ADOTable1.FieldName('Оцінка').AsString+'););
ADOTable1.Next;
end;
max:=
0;
ADOTable1.First;
While not ADOTable1.Eof
do begin
```



```

if (ADOTable1.FieldByName('Оцінка').AsFloat>max)
then begin
max:= ADOTable1.FieldByName('Оцінка').AsFloat;
pib:=ADOTable1.FieldByName('Кандидат').AsString
end;
ADOTable1.Next;
end;
Memo1.Lines.Add('Найкращім кандидатом на посаду можна вважати '+pib+ ',
який отримав оцінку '+floattostr(max)+' , яка є максимальною');
end;
procedure TForm12.Button3Click(Sender:
TObject); var WordApplication:variant;
begin
WordApplication:=CreateOLEObject('Word.Application'
); WordApplication.Visible:=True;
WordApplication.DisplayAlerts:=True;
Memo1.SelectAll;
Memo1.CopyToClipboard;
WordApplication.Documents.Add;
WordApplication.Documents.Item(1).Range.Paste
; end;
end.
unit
Unit13;
interface
uses

```

Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls,  
Forms, Dialogs, StdCtrls, ADODB, Grids, DBGrids, DB, DBCtrls, Unit14;  
type

```
TForm13 = class(TForm)
```

```
  ADOTable1:
```

```
  TADOTable; Label1:
```

```
  TLabel;
```

```
  Label2:
```

```
  TLabel;
```

```
  Label3:
```

```
  TLabel;
```

```
  Label4:
```

```
  TLabel;
```

```
  Label5:
```

```
  TLabel;
```

```
  DBComboBox1:
```

```
  TDBComboBox;
```

```
  DBComboBox2:
```

```
  TDBComboBox;
```

```
  DBComboBox3:
```

```
  TDBComboBox;
```

```
  DBComboBox4: TDBComboBox;
```

```
  DataSource1:
```

```
  TDataSource;
```

```
  DataSource2:
```

```
  TDataSource; DBGrid1:
```

```
  TDBGrid; ADOQuery1:
```



```

TADOQuery; Button1:
TButton; ADOTable2:
TADOTable; ADOTable3:
TADOTable; ADOTable4:
TADOTable;
procedure Button1Click(Sender:
TObject); private
{ Private declarations
} public
{ Public declarations
} end;
var
Form13:
TForm13;
implementation
{$R *.dfm}
procedure TForm13.Button1Click(Sender:
TObject); var Form14:TForm14;
begin
ADOQuery1.SQL.Clear
;
ADOQuery1.SQL.Add('select *');
ADOQuery1.SQL.Add('from Працівники
');
ADOQuery1.SQL.Add(' Where Прізвище = "'+DBCComboBox2.Text+'" and'
); ADOQuery1.SQL.Add(' Ім'я= "'+DBCComboBox3.Text+'" and');
ADOQuery1.SQL.Add(' По батькові = "'+DBCComboBox4.Text+'" and');

```

```
ADOQuery1.SQL.Add(' Табл = '+DBComboBox1.Text+");
ADOQuery1.Active:=True;
Form14:=TForm14.Create(Self);
Form14.ShowModal;
end;
end.
unit
Unit14;
interface
uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls,
  Forms, Dialogs, ExtCtrls, DBCtrls, Grids, DBGrids, StdCtrls, Buttons,
  ComCtrls, ADODB, DB;
type
  TForm14 = class(TForm)
    PageControl1:
      TPageControl; TabSheet1:
        TTabSheet; Label1: TLabel;
    Label2: TLabel;
    Label3: TLabel;
    Label4: TLabel;
    Label5: TLabel;
    Label6: TLabel;
    Label7: TLabel;
    Label36: TLabel;
    Edit1: TEdit;
```



RadioButton1:  
TRadioButton;  
RadioButton2:  
TRadioButton; Edit2: TEdit;  
Edit3:  
TEdit;  
Edit4:  
TEdit;  
ComboBox1: TComboBox;  
ComboBox2:  
TComboBox; GroupBox1:  
TGroupBox;  
Label8: TLabel;  
Label9: TLabel;  
Label11:  
TLabel; Edit5:  
TEdit; Edit6:  
TEdit; Edit7:  
TEdit; Edit28:  
TEdit;  
TabSheet2:  
TTabSheet; Label10:  
TLabel; Label16:  
TLabel; Edit8: TEdit;  
GroupBox2:  
TGroupBox; Label12:

TLabel; Label14:

TLabel;

Edit9: TEdit;

Edit10:

TEdit;

GroupBox3:

TGroupBox; Label13:

TLabel; Label15:

TLabel;

Edit11:

TEdit;

Edit12:

TEdit;

Edit13:

TEdit;

TabSheet3:

TTabSheet; Label17:

TLabel; Label18:

TLabel; Label19:

TLabel; Label20:

TLabel; Label21:

TLabel; Label22:

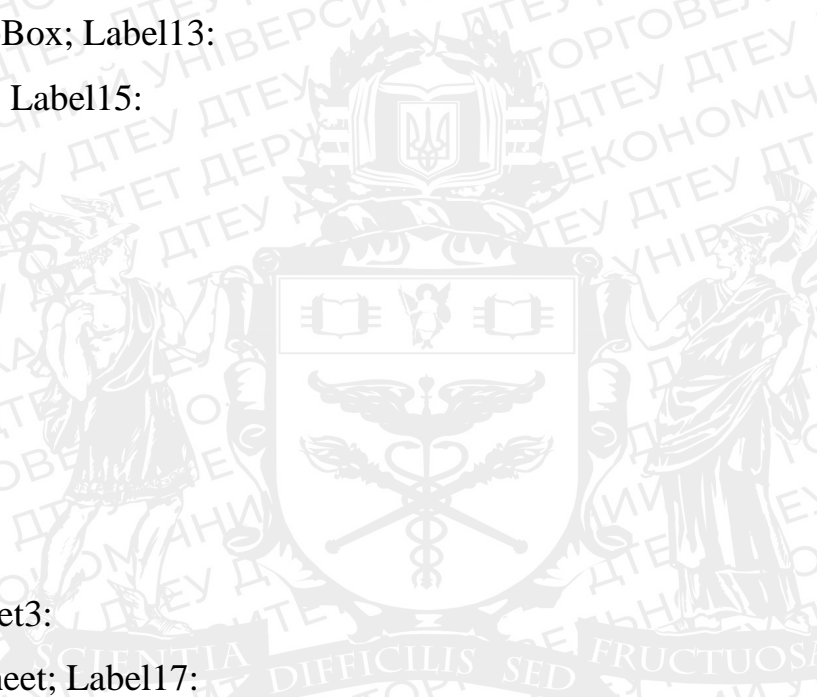
TLabel;

ComboBox3: TComboBox;

Edit14:

TEdit;

Edit15:





```
TEdit;  
Edit16:  
TEdit;  
Edit17:  
TEdit;  
Edit18:  
TEdit;  
ADOTable3: TADOTable;  
ADOTable4: TADOTable;  
ADOTable2: TADOTable;  
ADOQuery1:  
TADOQuery;  
ADOTable1: TADOTable;  
procedure FormCreate(Sender:  
TObject); private  
  { Private declarations  
  } public  
  { Public declarations  
  } end;  
var  
  Form14:  
  TForm14;  
implementation  
  {$R *.dfm}  
  procedure TForm14.FormCreate(Sender:  
  TObject); begin  
    ADOTable2.First;
```

```
While not ADOTable2.EOF
do begin
if
ADOTable2.FieldName('ID').AsInteger
=ADOQuery1.FieldName('ОбразованиеID').AsInteger then
ComboBox1.Text:=ADOTable2.FieldName('Образование').AsStrin
g; ADOTable2.Next;
end;

ADOQuery1.First;
if ADOQuery1.FieldName('Пол').AsBoolean=False then
RadioButton1.Checked:=True;
if ADOQuery1.FieldName('Пол').AsBoolean=True then
RadioButton2.Checked:=False;
Edit1.Text:=inttostr(ADOQuery1.FieldName('ТаблN').AsInteger);
Edit2.Text:=ADOQuery1.FieldName('Прізвище').AsString;
Edit3.Text:=ADOQuery1.FieldName('Ім'я').AsString;
Edit4.Text:=ADOQuery1.FieldName('По батькові').AsString;
Edit28.Text:=ADOQuery1.FieldName('ДатаНародження').AsString;
Edit7.Text:=ADOQuery1.FieldName('МістоНар').AsString;
Edit6.Text:=ADOQuery1.FieldName('Область').AsString;
Edit5.Text:=ADOQuery1.FieldName('КраїнаНар').AsString;
ComboBox1.Text:=ADOQuery1.FieldName('СімСтан').AsString;
Edit8.Text:=ADOQuery1.FieldName('ІНН').AsString;
Edit9.Text:=ADOQuery1.FieldName('АдресаРеєстрІндекс').AsString;
Edit10.Text:=ADOQuery1.FieldName('АдресаРеєстр').AsString;
Edit11.Text:=ADOQuery1.FieldName('АдресаФактІндекс').AsString;
Edit12.Text:=ADOQuery1.FieldName('АдресаФакт').AsString;
```





Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,  
Dialogs, StdCtrls, ComObj, DB, ADODB, ExtCtrls, DBCtrls, Grids, DBGrids;  
type

```
TForm15 = class(TForm)
```

```
DBGrid1: TDBGrid;
```

```
DBNavigator1: TDBNavigator;
```

```
DataSource1: TDataSource;
```

```
private
```

```
{ Private declarations
```

```
} public
```

```
{ Public declarations
```

```
} end;
```

```
var
```

```
Form15:
```

```
TForm15;
```

```
implementation
```

```
uses Unit12;
```

```
{ $R *.dfm
```

```
nit
```

```
Unit16;
```

```
interface
```

```
uses
```

Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls,

Forms, Dialogs, StdCtrls, DB, ADODB, DBCtrls, Menus, ExtCtrls;

```
type
```



TForm16 =

class(TForm) Panel1:

TPanel; Label14:

TLabel; Label15:

TLabel; Label16:

TLabel; Label17:

TLabel; Label1:

TLabel;

ComboBox1:

TComboBox;

ComboBox2:

TComboBox;

ComboBox3:

TComboBox;

ComboBox4:

TComboBox;

ComboBox5:

TComboBox;

MainMenu1:

TMainMenu; N1:

TMenuItem;

Panel2: TPanel;

Edit1: TEdit;

Edit2: TEdit;

Edit3: TEdit;

Edit4: TEdit;

Label2: TLabel;

Label3: TLabel;



Label4: TLabel;

Label5: TLabel;

Label6: TLabel;

Label7: TLabel;

Label8: TLabel;

ADOTable1: TADOTable;

ADOTable2: TADOTable;

DataSource2:

TDataSource; Edit8:

TEdit;

DataSource1:

TDataSource;

ComboBox6:

TComboBox; Label9:

TLabel;

Label10: TLabel;

ComboBox7: TComboBox;

Label11: TLabel;

ComboBox8: TComboBox;

Label12: TLabel;

ComboBox9: TComboBox;

Label13: TLabel;

ComboBox10:

TComboBox; Label18:

TLabel;

Label19: TLabel;

Button1:



```
TButton; Edit5:
TEdit; Button2:
TButton;
ADOTable3: TADOTable;
ComboBox11: TComboBox;
ComboBox12: TComboBox;
procedure N1Click(Sender:
TObject);
procedure Button1Click(Sender:
TObject); procedure Button2Click(Sender:
TObject); procedure FormCreate(Sender:
TObject); private
  { Private declarations }
public
  { Public declarations }
end;
var
  Form16:
  TForm16;
implementation
uses Unit1, Unit2;
{$R *.dfm}
procedure TForm16.N1Click(Sender:
TObject); begin
  Form1.Close;
  Form2.Close;
```

Form16.Clos

e; end;

procedure TForm16.Button1Click(Sender:

TObject); Var q:integer;

begin

edit5.Visible:=tru

e; q:=0;

if ComboBox4.Text='Вища освіта'

then q:=3;

if ComboBox4.Text='Технічне'

then q:=2;

if ComboBox4.Text='Середня загальна освіта'

then q:=1;

if ComboBox2.Text='від 1 року до 5'

then q:=q+1;

if ComboBox2.Text= 'від 5 до 10'

then q:=q+2;

if ComboBox2.Text='більше10' then

q:=q+3;

if ComboBox5.Text='Так'

then q:=q+3;

if ComboBox5.Text='Ні' then

q:=q+1;

if ComboBox1.Text='Так' then

q:=q+3;

if ComboBox1.Text='Ні' then

q:=q+1;



if ComboBox3.Text='Є' then  
q:=q+3;  
if ComboBox3.Text='Ні' then  
q:=q+1;  
if ComboBox6.Text='від 55 до 70' then  
q:=q+1;  
if ComboBox6.Text='від 71 до 85' then  
q:=q+2;  
if ComboBox6.Text='від 86' then  
q:=q+3;  
if ComboBox7.Text='Добре розуміє свої обов'язки' then  
q:=q+3;  
if ComboBox7.Text='Багато питань необхідно пояснювати' then  
q:=q+2;  
if ComboBox7.Text='Погано розуміє свою роботу' then  
q:=q+1;  
if ComboBox8.Text='Зберігає витримку' then  
q:=q+3;  
if ComboBox8.Text='Намагається стримувати себе у складних ситуаціях'  
then q:=q+2;  
if ComboBox8.Text='Нервова поведінка' then  
q:=q+1;  
if ComboBox9.Text='Уважно слухає, ставить уточнюючі питання' then  
q:=q+3;  
if ComboBox9.Text='Уважно слухає' then  
q:=q+2;

```
if ComboBox9.Text='Не слухає, перебиває' then
q:=q+1;
if ComboBox10.Text='Добре викладає свої думки'
then q:=q+3;
if ComboBox10.Text='Виникають складності в логічному обґрунтуванні своїх
думок' then
q:=q+2;
if ComboBox10.Text='Насилу викладає свої думки'
then q:=q+1;
edit5.Text:=floattostr(q/10
); end;
procedure TForm16.Button2Click(Sender:
TObject); begin
ADOTable3.Insert;
ADOTable3.FieldName('ID').AsInteger:=strtoint(Edit1.Text);
ADOTable3.FieldName('Відділ').AsString:=ComboBox11.Text;
ADOTable3.FieldName('Посада').AsString:= ComboBox12.Text;
ADOTable3.FieldName('Кількість').AsInteger:=strtoint(Edit2.Text);
ADOTable3.FieldName('Ставка').AsInteger:=strtoint(Edit3.Text);
ADOTable3.FieldName('Оклад').AsInteger:=strtoint(Edit4.Text);
ADOTable3.FieldName('Надбавка').AsInteger:=strtoint(Edit8.Text);
ADOTable3.FieldName('Рейтинг').AsFloat:=strtofloat(Edit5.Text);
ADOTable3.Post ;
end;
procedure TForm16.FormCreate(Sender:
TObject); begin
```



```
ADOTable1.First;
While not ADOTable1.Eof
do begin
Combobox12.Items.Add(ADOTable1.FieldByName('Посада').AsString);
ADOTable1.Next;
end;
ADOTable2.Firs
t;
While not ADOTable2.Eof
do begin
Combobox11.Items.Add(ADOTable2.FieldByName('Відділ').AsString
); ADOTable2.Next;
en
d;
en
d;
en
d.
```

