

**КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
ІМЕНІ ТАРАСА ШЕВЧЕНКА
ФАКУЛЬТЕТ КОМП'ЮТЕРНИХ НАУК ТА КІБЕРНЕТИКИ
КАФЕДРА ІНТЕЛЕКТУАЛЬНИХ ПРОГРАМНИХ СИСТЕМ**

«ЗАТВЕРДЖУЮ»

Заступник декана
з навчальної роботи

_____ Кашпур О.Ф.

«__» _____ 2019 року

**РОБОЧА ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ
ІНТЕЛЕКТУАЛЬНІ СИСТЕМИ НА
ОСНОВІ ЗНАНЬ
для студентів**

галузь знань	12 Інформаційні технології
спеціальність	121 Інженерія програмного забезпечення
освітній рівень	бакалавр
освітня програма	Програмна інженерія
спеціалізація	Інтелектуальні системи
вид дисципліни	вибіркова

Форма навчання	денна
Навчальний рік	2022/2023
Семестр	8
Кількість кредитів ECTS	5
Мова викладання, навчання та оцінювання	українська
Форма заключного контролю	іспит

Викладач: **к. ф.-м. н., асистент Терлецький Д.О.** (лекції, лабораторні заняття).

Пролонговано: на 20__/20__ н. р. _____ (_____) «__» 20__ р.

на 20__/20__ н. р. _____ (_____) «__» 20__ р.

Розробник: Терлецький Дмитро Олександрович, кандидат фізико-математичних наук,
асистент кафедри інтелектуальних програмних систем.

ЗАТВЕРДЖЕНО

Завідувач кафедри інтелектуальних програмних
систем

_____ О.І. Провотар

Протокол № __ від «__» _____2019 р.

Схвалено науково-методичною комісією факультету комп'ютерних наук та кібернетики

Протокол від «__» _____ 2019 року №__

Голова науково-методичної комісії _____ Л.Л. Омельчук

«__» _____ 2019 року

Затверджено вченою радою факультету комп'ютерних наук та кібернетики

Протокол від «__» _____ 2019 року №__

Голова вченої ради факультету _____ А.В. Анісімов

1. Мета дисципліни – вивчення основних моделей подання знань, методів аналізу предметних областей, аналізу, видобування та інтеграції знань, оволодіння сучасними методами проектування та розробки спеціалізованих інтелектуальних програмних систем на основі знань.

2. Попередні вимоги до опанування або вибору навчальної дисципліни. Для успішного вивчення дисципліни «Інтелектуальні системи на основі знань» студенти повинні відповідати наступним вимогам:

1. Успішне опанування курсів:
 1. Дискретна математика.
 2. Теорія алгоритмів та математична логіка.
 3. Організація баз даних та знань.
 4. Об'єктно-орієнтоване програмування.
 5. Програмування мовою Python.
 6. Системи штучного інтелекту.
2. Знання:
 1. Основних концепцій процедурного та об'єктно-орієнтованого програмування.
 2. Основ WEB-програмування.
 3. Основ програмування мовами C++, JavaScript, Java, Python.
 4. Основ проектування реляційних баз даних.
 5. Основ мови структурних запитів SQL.
 6. Основ теорії алгоритмів та теорії складності.
 7. Основ теорії множин та теорії графів.
 8. Основ теорії алгоритмів та математичної логіки.
3. Вміння:
 1. Програмувати у процедурному та об'єктно-орієнтованому стилях.
 2. Програмувати мовами C++/ JavaScript/ Java/ Python.
 3. Формувати запити до реляційних баз даних на мові SQL.
 4. Проектувати класові ієрархії мовами C++/ JavaScript/ Java/ Python.
4. Володіння:
 1. Базовими навичками використання інтегрованих середовищ розробки програмного забезпечення CLion/ WebStorm/ IntelliJ IDEA/ PyCharm/ Eclipse/ NetBeans/ Microsoft Visual Studio/ Microsoft Visual Studio Code.
 2. Англійською мовою на рівні не нижче Intermediate.

3. Анотація навчальної дисципліни. Навчальна дисципліна «Інтелектуальні системи на основі знань» є складовою освітньо-професійної програми підготовки фахівців за першим (бакалаврським) рівнем вищої освіти у галузі знань 12 Інформаційні технології за спеціальністю 121 Інженерія програмного забезпечення в рамках освітньо-професійної програми Програмна інженерія.

Дана дисципліна належить до переліку дисциплін вільного вибору студента, блок вибору «Інтелектуальні системи». Викладається у **8 семестрі в обсязі – 150 год., (5 кредитів ECTS)**, зокрема: лекції – 28 год., лабораторні заняття – 14 год., консультації – 8 год., самостійна робота – 100 год. У курсі передбачено 2 змістових частини, 2 контрольні роботи та 1 лабораторна робота. Завершується дисципліна – **іспитом**.

В результаті вивчення навчальної дисципліни студенти повинні:

Знати:

1. Основні моделі подання знань та їх властивості.
2. Основні типи джерел знань.
3. Основні методи видобування експертних знань.

4. Основні методи автоматичного видобування знань.
5. Основні етапи та методи інтеграції знань.
6. Основні методи аналізу знань.
7. Основні методи проектування та розробки інтелектуальних програмних систем на основі знань.

Вміти:

1. Реалізовувати, аналізувати та оцінювати моделі подання знань.
2. Проектувати та розробляти спеціалізовані інтелектуальні програмні системи на основі знань.

Дисципліна «Інтелектуальні системи на основі знань» є логічним продовженням, доповненням та розширенням дисципліни «Системи штучного інтелекту» першого (бакалаврського) рівня вищої освіти у галузі знань 12 Інформаційні технології за спеціальністю 121 Інженерія програмного забезпечення, в рамках освітньо-професійної програми Програма інженерія.

4. Завдання (навчальні цілі). Основними завданнями дисципліни «Інтелектуальні системи на основі знань» є набуття знань, умінь та навичок (компетентностей) на рівні новітніх досягнень в області інженерії знань та систем на основі знань відповідно до освітньої кваліфікації бакалавр з інженерії програмного забезпечення. Зокрема, розвивати:

- Здатність до абстрактного мислення, аналізу та синтезу (ЗК01).
- Здатність застосовувати знання у практичних ситуаціях (ЗК02).
- Здатність спілкуватися державною мовою як усно, так і письмово (ЗК03).
- Здатність спілкуватися іноземною мовою як усно, так і письмово (ЗК04).
- Здатність вчитися і оволодівати сучасними знаннями (ЗК05).
- Здатність до пошуку, оброблення та аналізу інформації з різних джерел (ЗК06).
- Здатність розробляти архітектури, модулі та компоненти програмних систем (СК03).
- Володіння знаннями про інформаційні моделі даних та системи, здатність створювати програмне забезпечення для зберігання, видобування та опрацювання даних (СК07).
- Здатність до алгоритмічного та логічного мислення (СК14).
- Здатність реалізувати сучасні методи побудови та аналізу ефективних алгоритмів в конкретних застосуваннях (СК15.1).
- Здатність використовувати інтелектуальні інформаційні технології (СК16.1).
- Здатність застосовувати математичний апарат та принципи програмування в процесі розробки програмних систем (СК17.1).

5. Результати навчання за дисципліною.

Результат навчання (1. знати; 2. вміти; 3. комунікація; 4. автономність та відповідальність)		Форми (та/або методи і технології) викладання і навчання	Методи оцінювання та пороговий критерій оцінювання (за необхідності)	Відсоток у підсумковій оцінці з дисципліни
Код	Результат навчання			
PH1.1	Знати основні типи знань, та	Лекції,	Контрольна	3%

	класифікацію моделей подання знань.	самостійна робота, опрацювання рекомендованої літератури.	робота № 1, іспит.	
PH1.2	Знати основні мережеві моделі подання знань та їх властивості.			4%
PH1.3	Знати основні логічні моделі подання знань та їх властивості.			4%
PH1.4	Знати основні процедурні моделі подання знань та їх властивості.			4%
PH1.5	Знати основні структурні моделі подання знань та їх властивості.			4%
PH1.6	Знати загальну архітектуру інтелектуальних систем на основі знань.		Контрольна робота № 2, іспит.	2%
PH1.7	Знати загальну архітектуру баз знань та операції над знаннями.			3%
PH1.8	Знати основні методи видобування та систематизації експертних знань.			6%
PH1.9	Знати основні методи автоматичного видобування знань.			6%
PH1.10	Знати основні методи інтеграції знань.			7%
PH1.11	Знати основні методи аналізу знань та отримання нових знань.			7%
PH2.1	Вміти проектувати та розробляти інтелектуальні програмні системи на основі мережевих моделей подання знань.	Виконання лабораторних робіт, самостійна робота, опрацювання рекомендованої літератури.	Контрольна робота № 1, задача лабораторної роботи, іспит.	5%
PH2.2	Вміти проектувати та розробляти інтелектуальні програмні системи на основі логічних моделей подання знань.			5%
PH2.3	Вміти проектувати та розробляти інтелектуальні програмні системи на основі процедурних моделей подання знань.			5%
PH2.4	Вміти проектувати та розробляти інтелектуальні програмні системи на основі структурних моделей подання знань.		Контрольна робота № 2, задача лабораторної роботи, іспит.	5%
PH2.5	Вміти проектувати та розробляти модулі видобування експертних знань для інтелектуальних програмних систем на основі знань.			5%
PH2.6	Вміти проектувати та розробляти модулі автоматичного видобування			5%

	знань для інтелектуальних програмних систем на основі знань.			
PH2.7	Вміти проектувати та розробляти модулі інтеграції знань для інтелектуальних програмних систем на основі знань.			5%
PH2.8	Вміти проектувати та розробляти модулі аналізу знань та отримання нових знань для інтелектуальних програмних систем на основі знань.			5%
PH3.1	Консультуватися з викладачем стосовно питань що виникають у ході опанування теоретичного матеріалу.	Лекції, лабораторні заняття, консультації, самостійна робота.	Здача лабораторної роботи.	2%
PH3.2	Обговорювати з колегами та викладачем проблемі питання що виникають у ході виконання лабораторної роботи.			2%
PH4.1	Закріплення та поглиблення набутих на лекціях теоретичних знань щодо проектування та розробки інтелектуальних програмних систем на основі знань.	Самостійна робота, опрацювання рекомендованої літератури.		3%
PH4.2	Закріплення та поглиблення набутих під час виконання лабораторної роботи практичних навичок проектування та розробки інтелектуальних програмних систем на основі знань.			3%

6. Співвідношення результатів навчання дисципліни із програмними результатами навчання.

Програмні результати навчання	Результати навчання дисципліни																							
	РН1.1	РН1.2	РН1.3	РН1.4	РН1.5	РН1.6	РН1.7	РН1.8	РН1.9	РН1.10	РН1.11	РН2.1	РН2.2	РН2.3	РН2.4	РН2.5	РН2.6	РН2.7	РН2.8	РН3.1	РН3.2	РН4.1	РН4.2	
ПРН01. Аналізувати, цілеспрямовано шукати і вибирати необхідні для вирішення професійних завдань інформаційно-довідникові ресурси і знання з урахуванням сучасних досягнень науки і техніки.		+	+	+	+				+	+	+	+											+	+
ПРН05. Знати і застосовувати відповідні математичні поняття, методи доменного, системного і об'єктно-орієнтованого аналізу та математичного моделювання для розробки програмного забезпечення.	+	+	+	+	+							+	+	+	+	+	+	+	+					
ПРН06. Уміння вибирати та використовувати відповідну задачі методологію створення програмного забезпечення.												+	+	+	+									
ПРН07. Знати і застосовувати на практиці фундаментальні концепції, парадигми і основні принципи функціонування мовних, інструментальних і обчислювальних засобів інженерії програмного забезпечення.												+	+	+	+									
ПРН08. Вміти розробляти людино-машинний інтерфейс.									+			+	+	+	+	+								
ПРН10. Проводити передпроектне обстеження предметної області, системний аналіз об'єкта проектування.		+	+	+	+							+	+	+	+					+	+			

7. Схема формування оцінки.

7.1 Форми оцінювання студентів.

Семестрове оцінювання: Максимальна/мінімальна кількість балів, які можуть бути отримані студентами протягом семестру, становить **60/36 балів** за яких:

1. Контрольна робота № 1: РН1.1 – РН1.5, РН2.1 – РН2.4 – **10/6 балів**.
2. Контрольна робота № 2: РН1.6 – РН1.11, РН2.5 – РН2.8 – **10/6 балів**.
3. Лабораторна робота (груповий програмний проект): РН2.1 – РН2.8, РН3.1, РН3.2, РН4.1, РН4.2 – **40/24 бали**.

Підсумкове оцінювання (у формі іспиту):

- Максимальна/мінімально-необхідна кількість балів які можуть бути отримані студентами на іспиті: **40 балів/24 бали**.
- Результати навчання які будуть оцінюватись: РН1.1 – РН1.11, РН2.1 – РН2.8.
- Форма проведення іспиту: письмова робота.
- Види завдань: 5 письмових завдань (2 теоретичних питання та 3 практичних завдання).

Студенти які під час семестру набрали менше ніж 20 (двадцять) балів **будуть не недопущені до іспиту**.

Критерії оцінювання на іспиті.

Завдання	Тема завдання	Максимальний відсоток від 40 балів	Всього відсотків
Завдання 1	Питання по теоретичному матеріалу курсу	15%	15%
Завдання 2		15%	15%
Завдання 3	Практичне завдання на основі теоретичного матеріалу курсу	15%	15%
Завдання 4		25%	25%
Завдання 5		30%	30%
			100%

Питання для підготовки до іспиту.

1. Типи знань та їх особливості. Приклади.
2. Класифікація моделей подання знань та їх особливості.
3. Загальна архітектура інтелектуальних систем на основі знань.
4. Бази знань та управління знаннями. Основні особливості. Приклади.
5. Методи видобування експертних знань. Основні особливості. Приклади.
6. Методи автоматичного видобування знань. Основні особливості. Приклади.
7. Системи автоматичного видобування знань. Основні особливості. Приклади.
8. Методи інтеграції знань. Основні особливості. Приклади.
9. Методи аналізу знань. Основні особливості. Приклади.
10. Семантичні мережі. Основні особливості. Приклади.
11. Загальна архітектура інтелектуальних систем на основі семантичних мереж. Основні особливості. Приклади.
12. Методи видобування експертних знань в інтелектуальних системах на основі семантичних мереж. Основні особливості. Приклади.
13. Методи автоматичного видобування знань в інтелектуальних системах на основі семантичних мереж. Основні особливості. Приклади.

42. Методи видобування експертних знань в інтелектуальних системах на основі систем фреймів. Основні особливості. Приклади.
43. Методи автоматичного видобування знань в інтелектуальних системах на основі систем фреймів. Основні особливості. Приклади.
44. Методи інтегрування знань в інтелектуальних системах на основі систем фреймів. Основні особливості. Приклади.
45. Методи аналізу знань в інтелектуальних системах на основі систем фреймів. Основні особливості. Приклади.
46. Об'єктно-орієнтоване програмування на основі класів. Основні особливості. Приклади.
47. Загальна архітектура інтелектуальних систем на основі об'єктно-орієнтованого програмування на основі класів. Основні особливості. Приклади.
48. Методи видобування експертних знань в інтелектуальних системах на основі об'єктно-орієнтованого програмування на основі класів. Основні особливості. Приклади.
49. Методи автоматичного видобування знань в інтелектуальних системах на основі об'єктно-орієнтованого програмування на основі класів. Основні особливості. Приклади.
50. Методи інтегрування знань в інтелектуальних системах на основі об'єктно-орієнтованого програмування на основі класів. Основні особливості. Приклади.
51. Методи аналізу знань в інтелектуальних системах на основі об'єктно-орієнтованого програмування на основі класів. Основні особливості. Приклади.
52. Об'єктно-орієнтоване програмування на основі прототипів. Основні особливості. Приклади.
53. Загальна архітектура інтелектуальних систем на основі об'єктно-орієнтованого програмування на основі прототипів. Основні особливості. Приклади.
54. Методи видобування експертних знань в інтелектуальних системах на основі об'єктно-орієнтованого програмування на основі прототипів. Основні особливості. Приклади.
55. Методи автоматичного видобування знань в інтелектуальних системах на основі об'єктно-орієнтованого програмування на основі прототипів. Основні особливості. Приклади.
56. Методи інтегрування знань в інтелектуальних системах на основі об'єктно-орієнтованого програмування на основі прототипів. Основні особливості. Приклади.
57. Методи аналізу знань в інтелектуальних системах на основі об'єктно-орієнтованого програмування на основі прототипів.. Основні особливості. Приклади.

7.2 Організація оцінювання.

Терміни проведення форм оцінювання:

1. Контрольна робота № 1: до 9 тижня семестру.
2. Контрольна робота № 2: до 18 тижня семестру.
3. Лабораторна робота (груповий програмний проект): до 19 тижня семестру.

У випадку отримання незадовільної оцінки (менше 50% від встановленого максимуму) за контрольну роботу, студенти мають право на одне перескладання цієї роботи у визначений викладачем термін із можливістю отримати не більше 80% балів від встановленої для неї максимальної кількості балів.

У випадку відсутності студентів з поважних причин відпрацювання та перездачі контрольних робіт здійснюються у відповідності до «Положення про порядок оцінювання знань студентів при кредитно-модульній системі організації навчального процесу» від 1 жовтня 2010 року.

У випадку встановлення фактів порушення студентами академічної доброчесності передбачених пунктом 9.8.2 «Положення про організацію освітнього процесу у Київському національному університеті імені Тараса Шевченка» що діє від 07.05.2018, вони будуть притягнуті до відповідальності передбаченої пунктом 9.8.3 цього положення.

7.3 Шкала відповідності оцінок.

Відмінно / Excellent	90-100
Добре / Good	75-89
Задовільно / Satisfactory	60-74
Незадовільно / Fail	0-59

8. Структура навчальної дисципліни. Тематичний план лекцій і лабораторних занять.

№ лекції	Назва лекції	Кількість годин		
		Лекції	Лабораторні заняття	Самостійна робота
Частина 1. Моделі подання знань.				
1	Вступ до курсу. Мета курсу. Зв'язок з іншими курсами. Структура курсу. Рекомендована література. Необхідний інструментарій. Лабораторні роботи. Вимоги до студентів. Форми контролю.	2		3
	Тема 1. Поняття знання. Типи знань. Класифікація моделей подання знань та їх особливості.			
2	Тема 2. Мережеві моделі подання знань: Семантичні мережі.	2		6
3	Тема 3. Мережеві моделі подання знань: Концептуальні графи.	2		7
4	Тема 4. Логічні моделі подання знань: Числення висловлювань. Логіка предикатів першого порядку.	2		7
5	Тема 5. Процедурні моделі подання знань: Продукційні системи.	2		7
6	Тема 6. Процедурні моделі подання знань: Скрипти та динамічна пам'ять.	2		6
7	Тема 7. Структурні моделі подання знань: Системи фреймів.	2		6
8	Тема 8. Структурні моделі подання знань: Об'єктно-орієнтоване програмування на основі класів та прототипів.	2		6
Контрольна робота 1				2
Всього по частині 1		16	0	50

Частина 2. Інженерія знань.				
9	Тема 9. Архітектура інтелектуальних систем на основі знань. Бази знань та управління знаннями.	2	2	7
10	Тема 10. Видобування експертних знань. Методи побудови діалогу. Систематизація знань.	2	2	10
11	Тема 11. Автоматичне видобування знань. Класифікація джерел знань. Методи та системи видобування знань.	2	4	10
12	Тема 12. Інтеграція знань: Розпізнавання. Обробка. Адаптація.	2	2	9
13	Тема 13. Аналіз знань. Перевірка припущень та отримання нових знань.	4	4	12
14				
Контрольна робота 2				2
Всього по частині 2		12	14	50
Консультація		8		
ВСЬОГО		28	14	100

Загальний обсяг – **150** год., в тому числі:

Лекцій – **28** год.

Лабораторні заняття – **14** год.

Консультації – **8** год.

Самостійна робота – **100** год.

Лабораторні роботи.

Лабораторна робота: Проектування та розробка спеціалізованих інтелектуальних систем на основі знань.

9. Рекомендовані джерела.

Основні:

1. R.J. Brachman, H.J. Levesque: Knowledge Representation and Reasoning. // Morgan Kaufmann Publishers, 2004.
2. M. Negnevitsky: Artificial Intelligence: A Guide to Intelligent Systems, 2nd, ed. // Addison-Wesley, 2005.
3. R.A. Akerkar, P.S. Sajja: Knowledge-based Systems. // Jones and Bartlett Publishers, LLC, 2010.
4. T.J.M. Bench-Capon: Knowledge Representation: An Approach to Artificial Intelligence. // Academic Press, 1990.

5. M. Chein, M.-L. Mugnier: Graph-based Knowledge Representation: Computational Foundations of Conceptual Graphs. // Springer, 2009.
6. P. Hitzler, H. Scharfe (eds.): Conceptual Structures in Practice. // Chapman & Hall/CRC, 2009.
7. A. Ligeza: Logical Foundations for Rule-Based Systems, 2nd ed. // Springer, 2006.
8. R.C. Schank, R.P. Abelson: Scripts, Plans, Goals and Understanding: An Inquiry into Human Knowledge Structures. // Lawrence Erlbaum Associates, Publishers, 1977.
9. M. Minsky: A Framework for Representing Knowledge. / Technical Report No. 306, Massachusetts Institute of Technology, A.I. Laboratory, 1974.
10. P.D. Karp: The Design Space of Frame Knowledge Representation Systems. / Technical Note No. 520, 1992.
11. I.D. Craig: Object-Oriented Programming Languages: Interpretation. // Springer, 2007.
12. G. Booch, R.A. Maksimchuk, M.W. Engle, B.J. Young, J. Conallen, K.A. Houston: Object-Oriented Analysis and Design with Applications, 3rd ed. // Addison-Wesley, 2007.
13. M. Weisfeld: The Object-Oriented Thought Process, 5th ed. // Addison-Wesley, 2019.
14. K.B. Bruce: Foundations of Object-Oriented Languages Types and Semantics. // The MIT Press, 2002.
15. B. Meyer: Object-Oriented Software Construction, 2nd ed. // Prentice Hall, 1997.
16. G. Blaschek: Object-Oriented Programming with Prototypes. // Springer, 1994.
17. C. Eichhorn: Rational Reasoning with Finite Conditional Knowledge Bases: Theoretical and Implementational Aspects. // J.B. Metzler, 2018.
18. G. Tecuci, D. Marcu, M. Boicu, D. Schum: Knowledge Engineering: Building Cognitive Assistants for Evidence-Based Reasoning. // Cambridge University Press, 2016.
19. R. Fagin, J.Y. Halpern, Y. Moses, M.Y. Vardi: Reasoning about Knowledge. // MIT Press, 2003.
20. K.S Murray: Learning as Knowledge Integration. // Ph.D. Thesis, Faculty of the Graduate School, University of Texas at Austin, 1995.
21. N.R. Milton: Knowledge Acquisition in Practice: A Step-by-step Guide. // Springer, 2007.
22. T.R. Gruber: The Acquisition of Strategic Knowledge. // Perspectives in Artificial Intelligence, Volume 4, Academic Press, Inc., 1989.
23. A.L. Kidd (ed.): Knowledge Acquisition for Expert Systems: A Practical Handbook. // Plenum Press, 1987.
24. S. Hellmann, J. Unbehauen, A. Zaveri, J. Lehmann, S. Auer, S. Tramp, H. Williams, O. Erling, T. Thibodeau Jr., K. Idehen, A. Blumauer, H. Nagy: Report on Knowledge Extraction from Structured Sources. // LOD2 – Creating Knowledge out of Interlinked Data, Research Project No. 257943.

Додаткові:

1. F. van Harmelen, V. Lifschitz, B. Porter (eds.): Handbook of Knowledge Representation. // Elsevier, 2008.
2. D. Corbett: Reasoning and Unification over Conceptual Graphs. // Springer, 2003.

3. J.F. Sowa: Knowledge Representation: Logical, Philosophical, and Computational Foundations. // Brooks/Cole, 2000.
4. D.S. Touretzky: The Mathematics of Inheritance Systems. // Morgan Kaufmann Publishers Inc., 1986.
5. R. Al-Asady: Inheritance Theory: An Artificial Intelligence Approach. // Ablex Publishing Corporation, 1995.
6. J. Polpinij: Ontology-Based Knowledge Discovery from Unstructured and Semi-Structured Text. // Ph.D. Thesis, School of Computer Science and Software Engineering, University of Wollongong, 2014.
7. N. Takhirov: Extracting Knowledge for Cultural Heritage Knowledge Base Population. // Ph.D. Thesis, Department of Computer and Information Science, Faculty of Information Technology, Mathematics and Electrical Engineering, Norwegian University of Science and Technology, 2013.
8. N.T. Nakashole: Automatic Extraction of Facts, Relations, and Entities for Web-Scale Knowledge Base Population. // Ph.D. Thesis, Faculty of Natural Sciences and Technology University of Saarland, 2012.
9. C.T. Leondes (ed.): Knowledge-Based Systems: Techniques and Applications. 4 Volume set. // Academic Press, 2000.
10. C.T. Leondes (ed.): Intelligent Knowledge-Based Systems: Business and Technology in the New Millennium. Volume 1: Knowledge-Based Systems. // Kluwer Academic Publishers, 2005.