

**Європейська Комісія
Програма "Темпус"**

**Міністерство освіти і науки України
Запорізький національний технічний університет**

**Міжнародний проект
«Європейсько-український
ступінь магістра
з програмного забезпечення»
(JER 26182 2006)**

**НАВЧАЛЬНИЙ ПЛАН ТА
РОБОЧИ ПРОГРАМИ**

**підготовки магістрів
за спеціальністю 8.080403**

**"Програмне забезпечення автоматизованих систем"
напряму 6.050103 "Програмна інженерія"**

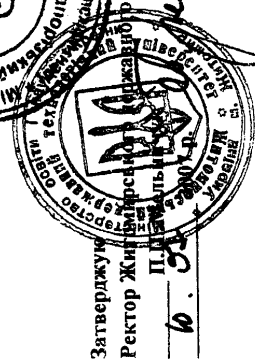
Затверджую

Ректор Запорізького національного технічного університету
Б.Б. Беліков
2007 р.



Затверджую

Ректор Житомирського державного технологічного університету
П.П. Шиньков



Міністерство освіти і науки України
Запорізький національний технічний університет
Житомирський державний технологічний університет

Освітньо-кваліфікаційний рівень-магістр
Термін навчання – 1 рік
Кваліфікація: аналітик комп'ютерних систем (2131.2 – ДК 003-95)
Номер та назва спеціальності 8.080403
«Програмне забезпечення автоматизованих систем»
на базі ОКР «Бакалавр»

НАВЧАЛЬНИЙ ПЛАН
Підготовки по спеціальності «Програмне забезпечення автоматизованих систем»

Напрямок 8.0804 «Комп'ютерні науки»
Магістр 8.080403 «Програмне забезпечення автоматизованих систем»

1. Графік навчального процесу

Курс	Вересень					Жовтень					Листопад					Грудень					Січень					Лютий																			
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45
5																																													
Курс	Березень					Квітень					Травень					Червень																													
5	27	28	29																																										

с

теоретичне навчання
екзаменаційна сесія

к
нп

канікули
наукова
практика

а
ва

державна атестація
виконання атестаційної роботи

Державна атестація

Вид атестації	Семестр
Захист атестаційної роботи у ДЕК	11

Практика	
Види практик	Семестр
Наукова практика	11
	Тижнів
	4

Курс	Теоретичне навчання	Екзаменні	Наукова практика	Виконання атестаційної роботи	Державна атестація	Канікули	Всього
5	27	4	4	4	2	2	43

Теоретичне навчання 37 ECTS
Практика 10 ECTS
Підготовка та захист магістерської роботи 13 ECTS
Разом 60 ECTS

Зведений бюджет часу (в тижнях)

2. План навчального процесу

№ пп	Назва дисципліни	Розподіл по семестрах			Годин						Розподіл аудиторних годин по курсах і по семестрах						
		Екзаменів	Заліків	Курсових проектів і робіт	Зарядний обсяг, годин (ECTS)	Всього	Лекції	Лабораторних занять	Практичних занять	Курсове проектування	Самостійна робота		5 курс				
											9 семестр - 17 тижнів	10 семестр - 10 тижнів				11 семестр	
	1. Цикл гуманітарних та соціально-економічних дисциплін																
1.1	Методологія наукових досліджень		9		72/2	34	18		16					38	2		
1.2.	Вища освіта України та Болонський процес. Педагогіка вищої школи		9		72/2	34	18		16					38	2		
1.3.	Інтелектуальна власність		10		54/1,5	20	10		10					34		2	
1.4.	Охорона праці в галузі та цивільна оборона		10		54/1,5	20	10		10					34		2	
	Всього з циклу 1		4		252/7	108	56		52					144	4	4	
	2. Цикл дисциплін фахової підготовки																
2.1.	Обов'язкові фахові дисципліни																
2.1.1.	Проектування, моделювання та аналіз інформаційних систем	9			108/3	68	34		34					40	4		
2.1.2.	Розподілені та паралельні системи	9			108/3	34	18		16					74	2		
2.1.3.	Менеджмент програмних систем	9			108/3	34	18		16					74	2		
2.1.4.	Сучасний штучний інтелект	9		КП	108/3	34	18		16					74	2		
2.1.5.	Інтелектуальні агенти		9		108/3	34	18		16					74	2		
2.1.6.	Архітектури та технології web- служб	9			108/3	34	18		16					74	2		

№ пп	Назва дисципліни	Розподіл по семестрах			Загальний обсяг, годин (ECTS)	1 один						годин по курсах і по семестрах					
		Екзаменів	Заліків	Курсових проектів і робіт		Аудиторних занять						Самостійна робота					
						Всього	Лекції	Лабораторних занять	Практичних занять	Курсове проектування	Всього	9 семестр - 17 тижнів	10 семестр - 10 тижнів	11 семестр			
2.2.	Фахові дисципліни за вибором студента																
2.2.A	Модуль А																
2.2.1.A	Моделі та метрика програмних систем	10			108 / 3	40	20	20			68				4		
2.2.2.A	Сучасні бази даних та інтелектуальний аналіз даних	10			72 / 2	20	10	10			52				2		
2.2.3.A	Технології розробки мережних додатків	10			72 / 2	20	10	10			52				2		
2.2.4.A	Безпека інформаційних систем	10			72 / 2	20	10	10			52				2		
2.2.5.A	Розробка програмних систем	10			108 / 3	40	20	20			68				4		
2.2.B	Модуль В																
2.2.1.B	Менеджмент безпеки інформаційних технологій	10			72 / 2	20	10	10			68				2		
2.2.2.B	Моделювання бізнес-процесів	10			108 / 3	40	20	20			52				4		
2.2.3.B	Проектування комп'ютерних мереж	10			72 / 2	20	10	10			52				2		
2.2.4.B	Геоінформаційні системи	10			72 / 2	20	10	10			52				2		
2.2.5.B	Розробка Agile програмного забезпечення	10			108 / 3	40	10	10			68				4		
	Всього з циклу	10	1	1	1080 / 30	378	194	184			702				14	14	
	Разом по 1, 2 циклам	10	5	1	1332 / 37	486	250	184	52		846				18	18	

№ пп	Назва дисципліни	Розподіл по семестрах			Годин						Розподіл аудиторних годин по курсах і по семестрах				
		Іспитів	Заліків	Курсових проектів і робіт	Загальний обсяг, годин (ECTS)	Всього	Аудиторних занять					Самостійна робота			
							Лекції	Лабораторних занять	Практичних занять	Курсове проектування	9 семестр – 17 тижнів	10 семестр – 10 тижнів	11 семестр		
III.	НАУКОВО-ПРАКТИЧНА ПІДГОТОВКА				216 / 10										
IV.	ПЕДАГОГІЧНА ПРАКТИКА				10										
	Разом	10	5	1	1558	486	250	184	52	846	18	18			

ЖДТУ

Проректор

Декан ФІКТ

Зав. кафедри ПЗОТ

Г.М.Виговський

А.П. Громовий

О.М. Данильченко

[Signatures]

ЗНТУ

Директор ІПРЕ

Начальник НІМВ

Декан ІОТ

Зав. кафедри ПЗ

Д.М. Піза

І.В. Ткаченко

М.М. Касьян

А.В. Дригула

[Signatures]

ПОГОДЖЕНО

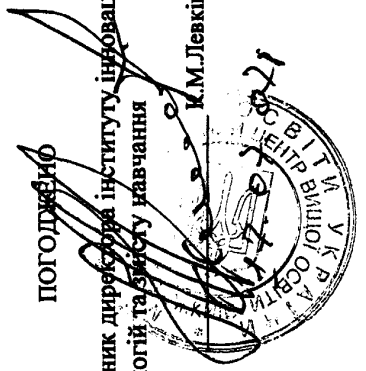
[Signature]
 Начальник департаменту вищої освіти
 Міністерства освіти і науки України



ПОГОДЖЕНО

Заступник директора інституту інноваційних технологій та змісту навчання

К.М. Левківський



ПОГОДЖЕНО

Голова Науково-методичної комісії МОНУ
 з напрямку „Програмна інженерія”

М.Ф. Бондаренко



**Міністерство освіти і науки України
Запорізький національний технічний університет
Інститут інформатики та радіоелектроніки**

“ЗАТВЕРДЖУЮ”

Ректор
Запорізького національного
технічного університету,
д.т.н., проф.

_____ /С.Б. Беліков/
“ _____ ” _____ 2008 р

**Робочі програми
підготовки магістрів за спеціальністю
"Програмне забезпечення автоматизованих систем"
напряму 6.050103 "Програмна інженерія"**

Схвалено методичною комісією факультету інформатики та обчислювальної техніки. Протокол № 1 від " 27 " вересня 2007 р.

Голова методкомісії, декан ФІОТ

_____ М.М. Касьян

Затверджено на засіданні кафедри програмних засобів.

Протокол № 1 від "28" серпня 2007 р.

Зав. кафедри програмних засобів

_____ А.В. Притула

РОБОЧА НАВЧАЛЬНА ПРОГРАМА

дисципліни

“Проектування, моделювання та аналіз інформаційних систем”

для студентів магістратури напрямку підготовки

6.050103 “Програмна інженерія”

спеціальності 8.080403 “Програмне забезпечення автоматизованих систем”

Факультет інформатики та обчислювальної техніки

Кафедра програмних засобів

Форма навчання	Курс	Семестр	Усього (год)	Лекції (год)	Лабор. роб. (год)	Практ. (год)	Самост. під керівн.(год)	Самост. (год)	КП, КР, РГЗ (сем.)	Залік (сем.)	Іспит (сем.)
Денна	V	9	108/3	34	34	-	-	40	-	-	9
Заочна											

Робоча програма складена на основі Навчального плану підготовки магістрів по спеціальності "Програмне забезпечення автоматизованих систем" напрямку 6.050103 "Програмна інженерія" від 17.07.2007 р. за Міжнародним проектом "Європейсько-український ступінь магістра з програмного забезпечення" (JEP 26182 2006) програми "Темпус" Європейської Комісії.

Робоча програма складена
кандидатом технічних наук, доцентом

Табунщик Г.В.

1 МЕТА І ЗАДАЧІ ДИСЦИПЛІНИ, ЇЇ МІСЦЕ В НАВЧАЛЬНОМУ ПРОЦЕСІ

1.1 Мета викладання дисципліни

Метою даного курсу є закріплення теоретичних основ та практичних навичок об'єктно-орієнтованого аналізу та проектування. Дисципліна "Проектування, моделювання та аналіз інформаційних систем" спрямована на поглиблення знань та практичних навичок студентів з основ сучасної технології створення складних програмних продуктів на базі ідей і принципів об'єктно-орієнтованого методу. Такі знання призначені для використання при проектуванні складних корпоративних програмних додатків з урахуванням сучасних вимог у відношенні до надійності, якості та ефективності програмних продуктів, що створюються.

1.2 Завдання вивчення дисципліни:

На основі вивчення дисципліни " Проектування, моделювання та аналіз інформаційних систем " студент повинен

з н а т и:

– етапи життєвого циклу програмного забезпечення, елементи об'єктної моделі, процедури об'єктно-орієнтованого аналізу, процедури об'єктно-орієнтованого проектування, основи уніфікованої мови моделювання UML, основні концепції уніфікованого процесу розробки програмного забезпечення UP, архітектуру програмного забезпечення, CASE –засоби формування вимог до програмного забезпечення та проектування програмного забезпечення.

в м і т и:

– розробляти *об'єктно-орієнтовану* *декомпозицію* предметної області в умовах проектування складних об'єктів і систем за допомогою процедур об'єктно-орієнтованого аналізу, об'єктно-орієнтованого проектування, використовуючи визначення класів, поведінки об'єктів, структури даних та їх взаємозв'язки;

– робити *об'єктно-орієнтований синтез* інформаційної та функціональної моделі в умовах проектування складних об'єктів і систем за допомогою процедур об'єктно-орієнтованого аналізу, об'єктно-орієнтованого проектування, об'єктно-орієнтованого програмування, використовуючи діаграми класів, процесів, об'єктів;

– розробляти *об'єктно-орієнтовану модель* предметної галузі за допомогою мов об'єктно-орієнтованого моделювання в умовах проектування складних об'єктів та систем, використовуючи об'єктно-орієнтовану нотацію складних систем;

– виконувати *об'єктно-орієнтовану* *декомпозицію* та *об'єктно-орієнтований* аналіз предметної області в умовах формалізації предметної

області за допомогою процедур об'єктно-орієнтованого аналізу, проектування та програмування., використовуючи ієрархію класів, діаграми процесів та об'єктів;

– визначати *структуру програмного забезпечення* комп'ютерних інформаційних систем в умовах ескізного і технічного проектування на основі аналізу функціональної структури системи, використовуючи інформацію про математичне, технічне, інформаційне забезпечення;

– розробляти *проект програмної системи* в умовах технічного проектування за допомогою технології об'єктно-орієнтованого проектування, використовуючи діаграми класів та об'єктів, діаграми процесів та переходів.

1.3 Зв'язок з іншими дисциплінами

Курс базується на поняттях, що вивчаються в дисциплінах «Об'єктно-орієнтоване програмування» та «Технології програмування та створення програмних продуктів».

Отриманні по розглянутій дисципліні знання будуть доповнюватися у курсах “Менеджмент програмних систем” та “Моделі та метрики програмних систем”.

2 ЗМІСТ ДИСЦИПЛІНИ

Навчальний процес в ЗНТУ в умовах кредитно-модульної системи організації навчання здійснюється у таких організаційних формах: навчальні заняття, виконання індивідуальних завдань, самостійна робота студентів, практична підготовка, контрольні заходи. Основні види занять, що входять до складу модулів: лекція, лабораторне, семінарське заняття, консультація та самостійна робота під керівництвом викладача.

Дисципліна "Проектування, моделювання та аналіз інформаційних систем" вивчається на 5 курсі у 9 семестрі і завершується іспитом. Навчальним планом передбачено:

Всього:

- лекцій – 34 годин;
- лабораторних робіт – 34 годин;
- самостійної роботи – 40 годин;

Загальна кількість аудиторних годин – 108 годин.

В умовах кредитно-модульної системи навчальна дисципліна "Проектування, моделювання та аналіз інформаційних систем" відповідає 3 заліковим кредитам.

Дисципліна складається з 6 блоків змістових модулів – основних її розділів, які разом зі змістовими модулями наведені у таблиці 2.1.

Таблиця 2.1 – Склад, обсяг і терміни виконання змістових модулів дисципліни "Проектування, моделювання та аналіз інформаційних систем"

Модулі (блоки змістових модулів)	Короткий зміст модулів	Розподіл навчального часу за елементами модуля (видами занять), години			Самостійна робота студента		Обсяг навантаження студента	
		Л	ЛР	ПР	СРВ	СРС	Години	Кредити
1	2	3	4	5	7	8	9	10
1.Характеристики процесу розробки програмного забезпечення	Вступ	4				14	18	0,5
	Характер процесу розробки програмного забезпечення (ПЗ)							
	Планування розробки системи							
	Етапи життєвого циклу ПЗ							
2. Основи об'єктних технологій	Історичний огляд	2				7	9	0,25
	Сутність об'єктно-орієнтованого підходу							
	Об'єктно-орієнтовані мови програмування							

Продовження таблиці 2.1

1	2	3	4	5	7	8	9	10
3. Засоби аналізу, проектування та моделювання інформаційних систем	Основні компоненти мови UML	5				4	9	0,25
	CASE-засоби для розробки діаграм							
	Можливості та особливості інформаційних технологій Rational							
4. Ітеративна розробка та уніфікований процес	Основна характеристика ітеративного процесу, фаз та дисципліни	7	6			5	18	0,5
	Особливості та характеристика RUP							
5. Аналіз та специфікація вимог	Визначення вимог	8	14			5	27	0,75
	Керування вимогами							
	Принципи специфікації вимог							
	Специфікація функціональних вимог							
	Моделювання класів та асоціацій							
	Моделювання відношень агрегації та композиції							
Моделювання об'єктів								
6. Моделювання та проектування систем	Шаблони розподілення обов'язків	8	14			5	27	0,75
	Шаблони проектування							
	Моделювання станів об'єктів							
	Загальні принципи побудови компонентних моделей							
	Загальні методи архітектурного аналізу							
	Архітектурний аналіз в UP							
Всього		34	34			40	108	3,00

Позначення у таблиці 2.1: Л–лекції, ЛР – лабораторні роботи, ПР – практична робота, СРС – самостійна робота студента

2.1 Змістові модулі

2.1.1 *Характеристики процесу розробки програмного забезпечення*

Вступ. Мета і завдання курсу. Зміст курсу, його місце у навчальному плані спеціальності. Види та об'єми занять, форми контролю.

Характер процесу розробки програмного забезпечення. Планування розробки системи. Етапи життєвого циклу ПЗ.

Лекцій – 4 год.

Самостійна робота – 10 год.

Література: [1,4,8].

2.1.2 *Основи об'єктних технологій*

Історичний огляд розвитку методології об'єктно-орієнтованого аналізу та проектування складних систем.

Сутність об'єктно-орієнтованого підходу. Елементи об'єктної моделі: об'єкт-екземпляр, клас, асоціації, агрегація і композиція, об'єкт-клас.

Основні визначення об'єктно-орієнтованого аналізу та проектування. Приклад об'єктно-орієнтованого аналізу та проектування: визначення прецедентів, визначення моделі предметної області, діаграми взаємодій, діаграми класів.

Огляд об'єктно-орієнтованих мов програмування: Object Pascal, C++, Java та інші.

Лекцій – 2 год.

Самостійна робота – 7 год.

2.1.3 *Засоби аналізу, проектування та моделювання інформаційних систем*

Мова UML. Специфіка опису метамоделі мови UML. Особливості зображення діаграм мови UML. Основні етапи розвитку UML.

Діаграми в UML: діаграма класів, діаграма прецедентів, діаграма стану, діаграма взаємодії, діаграма послідовностей, діаграма кооперації, діаграма розгортання, діаграма класів.

CASE-засоби для розробки діаграм UML. Критерії вибору CASE-засобів розробки.

Можливості та особливості інформаційних технологій Rational

Лекцій – 3 год.

Самостійна робота – 6 год.

Література: [1–4, 8, 10].

2.1.4 Ітеративна розробка програмного забезпечення та шаблони проектування

Основна характеристика ітеративного процесу, фаз та дисципліни. Особливості та характеристика RUP. Артефакти.

Початкова фаза: модель бізнес-прецедентів, визначення інших вимог, артефакти початкової фази.

Лекцій – 7 год.

Лабораторна робота – 6 год.

Самостійна робота – 5 год.

Література: [1–4, 8, 10].

2.1.5 Аналіз та специфікація вимог

Визначення вимог: функціональні та нефункціональні вимоги. Поняття рамок системи. Процес керований прецедентами. Документ концепція проекту.

Керування вимогами. Специфікація функціональних вимог.

Моделювання класів та асоціацій. Моделювання відношень агрегації та композиції. Моделювання об'єктів

Лекцій – 7 год.

Лабораторна робота – 6 год.

Самостійна робота – 5 год.

Література: [1–4, 8, 10].

2.1.6 Моделювання та проектування систем

Основні переваги використання шаблонів проектування: Шаблони розподілення обов'язків, Шаблони проектування Façade, Adapter, Bridge, Abstract Factory.

Практичне використання шаблонів проектування: вирішення завдань САПР з допомогою шаблонів проектування, обробка варіацій за допомогою шаблонів проектування.

Моделювання станів об'єктів.

Компонентний аналіз систем.

Архітектурний аналіз.

Перелік принципів об'єктно-орієнтованого проектування

Лекцій – 6 год.

Лабораторна робота – 14 год.

Самостійна робота – 7 год.

Література: [1–4, 8, 10].

2.2 Перелік лабораторних робіт

Мета проведення лабораторних занять посідає в придбанні досвіду та навичок для проведення об'єктно-орієнтованого аналізу із застосуванням мови UML

Лабораторна робота № 1. Початкова фаза ітераційного процесу

Мета роботи: Ознайомитись з основними етапами RUP та виконати роботи відповідно до дисциплін.

Обсяг – 6 годин

Лабораторна робота № 2. Специфікація функціональних вимог

Мета роботи: Навчитись знаходити та задавати специфікацію для функціональних вимог системи.

Обсяг – 4 години

Лабораторна робота № 3. Конструювання моделі предметної області

Мета роботи: Навчитись формулювати вимоги до предметної області за допомогою моделі класів

Обсяг – 6 годин

Лабораторна робота № 4. Проектування взаємодії

Мета роботи: Вивчити базові прийоми проектування інформаційних систем

Обсяг – 5 годин

Лабораторна робота № 5. Проектування обов'язків

Мета роботи: Навчитись використовувати шаблони проектування обов'язків

Обсяг – 5 годин

Лабораторна робота № 6. Шаблони проектування

Мета роботи: Навчитись використовувати шаблони при проектуванні систем.

Обсяг – 5 годин

Лабораторна робота № 7. Моделювання поведінки

Мета роботи: Навчитись моделювати динамічні об'єкти інформаційних систем

Обсяг – 5 годин

2.3 Перелік тем для самостійної роботи

Нижче наведено перелік тем для самостійної роботи студентів:

- | | |
|--|---------|
| 1. Методи вдосконалення програмних засобів | 10 год. |
| 2. Засоби планування розробки систем | 4 год. |
| 3. Об'єктно-орієнтовані мови програмування | 7 год. |
| 4. Програмні засоби фірми Rational | 6 год. |
| 5. Приклад змісту ітерацій RUP та приклад артефактів | 5 год. |
| 6. Діаграми UML | 14 год. |

3 НАВЧАЛЬНО-МЕТОДИЧНІ МАТЕРІАЛИ З ДИСЦИПЛІНИ

3.1 Література основна

1. Буч Г. Объектно-ориентированный анализ и проектирование с примерами приложений на C++. – М.:Бином, 1999. –560 с.
2. Крачтен Я. Введение в Rational Unified Process. Пер. с англ. – М.:Издательский дом «Вильямс», 2002. – 240 с.: ил
3. Кватрани Ю. Визуальное моделирование с помощью Rational Rose 2002 и UML. Пер. с англ. –М.:Издательский дом «Вильямс», 2002. – 240 с.: ил
4. Ларман К. Применение UML и шаблонов проектирования. 2-е изд. Пер. с англ. –М.:Издательский дом «Вильямс», 2004. –625 с.
5. Мацяшек Л.А. Анализ требований и проектирование систем. Разработка информационных систем с использованием UML. Пер. с англ. – М.:Изд. дом «Вильямс», 2002. –432 с.
6. Шаллоуей А. Шаблоны проектирования. Новый подход к объектно-ориентированному анализу и проектированию. Пер. с англ. –М.: Изд. дом «Вильямс», 2002. – 288 с.

3.2 Література додаткова

7. Бергстрём С., Роберг Л. Rational Unified Process – путь к успеху. Руководство по внедрению RUP. – М.: Кудиц-образ, 2004. – 256 с.
8. Трофимов С.А. Case-технологии. Практическая работа в Rational Rose. – М.: «Бином», 2001. –

3.3 Методичні матеріали з дисципліни

9. Методичні вказівки до виконання лабораторних робіт з дисципліни "Проектування, моделювання та аналіз інформаційних систем " для студентів для студентів магістратури напряму підготовки 6.050103 "Програмна інженерія" спеціальності 8.080403 "Програмне забезпечення автоматизованих систем" / Укл. Табунщик Г.В. – Запоріжжя: ЗНТУ, 2007.– 20 с.

3.4 Програмні засоби

Пакети Microsoft Visual C 6.0., IBM Rational Rose, IBM Requisite Pro

4 ПЕРЕЛІК ПИТАНЬ З ДИСЦИПЛІНИ

1. Характеристики процесу розробки програмного забезпечення. (ПЗ). Сутність та властивості, що характерні програмній інженерії.
2. Характеристики процесу розробки ПЗ. Методи вдосконалення процесу розробки.
3. Характеристики процесу розробки ПЗ. Сучасні процеси розробки ПЗ.
4. Характеристики процесу розробки ПЗ. Засоби автоматизованого проектування та створення програм.
5. Планування розробки інформаційної системи.
6. Етапи життєвого циклу ПЗ.
7. Огляд підходів до розробки ПЗ.
8. Елементи об'єктної моделі. Абстрагування. Інкапсуляція.
9. Елементи об'єктної моделі. Відношення класів.
10. Елементи об'єктної моделі. Типи структурних ієрархій.
11. Елементи об'єктної моделі ієрархія та модулі.
12. Основні компоненти мови UML,
13. Діаграми мови UML. Діаграма класів.
14. Етапи розробки програмного забезпечення.
15. Основні концепції аналізу предметної області.
16. Основні концепції аналізу додатків.
17. Основні концепції проектування системи.
18. Основні концепції реалізації інформаційних систем.
19. Основні концепції уніфікованого процесу та ітеративній розробці ПЗ.
20. Модель предметної області
21. Аналіз та специфікація вимог.
22. Модель стану об'єктів.
23. Шаблони проектування.
24. Розробка програмного забезпечення на базі компонентів.
25. Введення в архітектурний аналіз.

РОБОЧА НАВЧАЛЬНА ПРОГРАМА
дисципліни
“ РОЗПОДІЛЕНІ ТА ПАРАЛЕЛЬНІ СИСТЕМИ”
для студентів магістратури напряму підготовки
0501 “Програмна інженерія”

спеціальності **8.080403 “Програмне забезпечення автоматизованих систем”**

Факультет інформатики та обчислювальної техніки
Кафедра програмних засобів

Форма навчання	Курс	Семестр	Усього (год)	Лекції (год)	Лабор. роб. (год)	Практ. (год)	Самост. під керівн.(год)	Самост. (год)	КП, КР, РГЗ (сем.)	Залік (сем.)	Іспит (сем.)
Денна	V	9	108	18	16	–	–	74	–	–	10
Заочна											

Робоча програма складена на основі Навчального плану підготовки магістрів по спеціальності "Програмне забезпечення автоматизованих систем" напряму 6.050103 "Програмна інженерія" від 17.07.2007 р. за Міжнародним проектом "Європейсько-український ступінь магістра з програмного забезпечення" (JEP 26182 2006) програми "Темпус" Європейської Комісії.

Робоча програма складена
канд. техн. наук, доцентом

Кудерметовим Р.К.

1 МЕТА І ЗАДАЧІ ДИСЦИПЛІНИ, ЇЇ МІСЦЕ В НАВЧАЛЬНОМУ ПРОЦЕСІ

1.1 Мета викладання дисципліни

Засвоєння студентами основних теоретичних відомостей та практичних вмінь з курсу. Підготувати студента до ефективного використання сучасних методів паралельного та розподіленого обчислення з використанням комп'ютерних кластерів корпоративних комп'ютерних мереж і міжплатформного програмного забезпечення.

1.2 Завдання вивчення дисципліни:

Внаслідок вивчення дисципліни студенти повинні:

- сформувані знання та отримати практичні навички для використання методів паралельного і розподіленого програмування;
- отримати уяву про стан і перспективи розвитку високопродуктивних та розподілених комп'ютерних систем.

На основі вивчення дисципліни студент повинен

знати:

- основні поняття та визначення паралельних обчислень;
- основні поняття та визначення розподілених обчислень;
- сучасні програмні засоби для проектування і розробки паралельних програм та розподілених додатків;

вміти:

- використовувати сучасні програмні засоби для проектування паралельних програм;
- аналізувати складність, ефективність та прискорення паралельних обчислень;
- використовувати міжплатформне програмне забезпечення для створення розподілених програмних додатків.

1.3 Зв'язок із іншими дисциплінами

Курс базується на поняттях, що вивчаються в дисциплінах:

- програмування;
- системне програмування;
- операційні системи;
- комп'ютерні мережі.

2 ЗМІСТ ЛЕКЦІЙ (18 ГОДИН)

2.1 Паралельні обчислення

Паралельні алгоритми та їх характеристики. Закон Амдала. Архітектура та класифікація суперкомп'ютерних систем. Основні положення стандарту MPI (Message Passing Interface). Функції обміну даними бібліотеки mpich 1.2.4. Функції управління групами і комунікаторами. Методи паралельної обробки матриць. Прямі та ітераційні методи паралельного розв'язання лінійних рівнянь.

Лекцій – 9 год.

Лаб. роб. – 8 год.

Самостійна робота – 37 год.

Література [1, 2, 3, 6]

2.2 Розподілені обчислення

Розподілені комп'ютерні системи. Міжплатформне програмне забезпечення для створення розподілених програмних додатків (middleware). Забезпечення інтегруєбельності за допомогою мови IDL (Interface Definition Language). Створення розподілених систем за допомогою технології RPC (Remote Procedure Calls). Архітектура CORBA (Common Object Request Broker Architecture). Структура CORBA-додатку. Інтегруєбельність в Java. Архітектура Java/RMI (Remote Method Invocation). Структура Java/RMI-додатку.

Лекцій – 9 год.

Лаб. роб. – 8 год.

Самостійна робота – 37 год.

Література [4, 5, 6]

3 ПЕРЕЛІК ЛАБОРАТОРНИХ РОБІТ

Лабораторна робота № 1. Знайомство з бібліотекою MPI (2 год.).

Мета роботи: вивчити каркас паралельних програм на базі функцій бібліотеки стандарту MPI (реалізація mpich 1.2.4), навчитися створювати паралельні програми та запускати їх на виконання в операційному середовищі Red Hat Linux.

Лабораторна робота № 2. Функції обміну в MPI (2 год.).

Мета роботи: продовжити вивчення функцій обміну бібліотеки MPI, зокрема, функцій колективного обміну. Засвоїти деякі прийоми їх використання для розподілу даних та обчислень між паралельними процесорами.

Лабораторна робота № 3. Паралельні методи інтегрування. Використання функцій колективного обміну MPI (4 год.).

Мета роботи: продовжити вивчення функцій обміну бібліотеки MPI, зокрема, функцій колективного обміну. Вивчити деякі прийоми їх використання для розподілення даних та обчислень між паралельними процесорами.

Лабораторна робота № 4. Віддалений виклик процедур (4 год.).

Мета роботи: вивчити механізм віддаленого виклику процедур (Remote Procedure Calls, RPC). Навчитися створювати розподілені програми, які використовують RPC, за допомогою утиліти gprgen.

Лабораторна робота № 5. CORBA (4 год.).

Мета роботи: познайомитись зі специфікацією CORBA (Common Object Request Broker Architecture). Навчитися створювати прості розподілені програми за допомогою відкритої реалізації CORBA 2.3 – MICO (MICO IS CORBA).

4 ПЕРЕЛІК ТЕМ ДО САМОСТІЙНОЇ РОБОТИ

1. Класифікації комп'ютерних систем (6 год.).
2. Комп'ютерні архітектури WLIV, SMP, MPP, NUMA, ccNUMA (15 год.).
3. Функції управління топологіями в MPI (16 год.).
4. Віртуальна машина JVM (4 год.).
5. Міжплатформне програмне забезпечення Microsoft DCOM (16 год.).
6. Сервіс-орієнтована архітектура розподілених систем SOA. (17 год.).

Контроль самостійної роботи передбачає написання рефератів або контрольних робіт.

5 НАВЧАЛЬНО-МЕТОДИЧНІ МАТЕРІАЛИ З ДИСЦИПЛІНИ

5.1 Література основна

1. Немнюгин С.А., Стесик О.Л. Параллельное программирование для многопроцессорных вычислительных систем. – СПб.: БХВ – Санкт-Петербург, 2002. – 400 с.
2. Воеводин В.В., Воеводин Вл.В. Параллельные вычисления. – СПб.: БХВ-Петербург, 2002. – 608 с.
3. Ортега Дж. Введение в параллельные и векторные методы решения линейных систем. – М.: Мир, 1991. – 367 с.
4. Бэкон Д., Харрис Т. Операционные системы. Параллельные и распределенные системы. – СПб.: Питер; Киев: Издательская группа ВНУ, 2004. – 800 с.
5. Хьюз К., Хьюз Т. Параллельное и распределенное программирование на C++. – М.: Издательский дом "Вильямс", 2004. – 672 с.
6. Parallel.ru – информационно-аналитический центр по параллельным вычислениям – <http://www.parallel.ru>.

5.2 Література додаткова

7. Корнеев В.Д. Параллельное программирование в MPI. – 2-е изд., испр. – Новосибирск: Изд-во ИВМиМГ СО РАН, 2002. – 215 с.
8. Эндрюс Г.Р. Основы многопоточного, параллельного и распределенного программирования.: Пер. с англ. – М.: Издательский дом "Вильямс", 2003. – 512 с.
9. Гергель В.П., Стронгин Р.Г. Основы параллельных вычислений для многопроцессорных вычислительных систем. Учебное пособие – Нижний Новгород; Изд-во ННГУ им. Н.И. Лобачевского, 2000. –176 с.
10. Бройнль Т. Паралельне програмування: Початковий курс: Навч. посібник / Вступ. Слово А.Ройтера; Пер. з нім. В.А. Святного. – К.: Вища школа, 1997. – 358 с.
11. Букатов А.А., Дацюк В.Н., Жегуло А.И. Программирование многопроцессорных вычислительных систем. – Ростов-на-Дону: Изд-во ООО "ЦВВР", 2003. – 208 с.
12. Корнеев В.В. Параллельные вычислительные системы. – М.: Нолидж, 1999. – 320 с.
13. Теренс Чан. Системное программирование на C++ для UNIX. – К.: Издательская группа ВНУ, 1999. – 592 с.
14. Митчелл М, Оулдем Дж. Самьюэл А. Программирование для Linux. Профессиональный подход. – М.: Издательский дом "Вильямс", 2002. – 288 с.
15. Примеры RPC-приложений.
<ftp://ftp.oreilly.com/pub/examples/nutshell/rpc/rpc.tar.z>
16. Сигел Дж. CORBA 3. –М.: МАЛИП, 2002. – 412 с.

РОБОЧА НАВЧАЛЬНА ПРОГРАМА

дисципліни

“Менеджмент програмних систем”

для студентів магістратури напряму підготовки

6.050103 “Програмна інженерія”

спеціальності 8.080403 “Програмне забезпечення автоматизованих систем”

Факультет інформатики та обчислювальної техніки

Кафедра програмних засобів

Форма навчання	Курс	Семестр	Усього (год)	Лекції (год)	Лабор. роб. (год)	Практ. (год)	Самост. під керівн.(год)	Самост. (год)	КП, КР, РГЗ (сем.)	Залік (сем.)	Іспит (сем.)
Денна	V	9	108/3	18	16	-	-	74	-	-	9
Заочна											

Робоча програма складена на основі Навчального плану підготовки магістрів по спеціальності "Програмне забезпечення автоматизованих систем" напряму 6.050103 "Програмна інженерія" від 17.07.2007 р. за Міжнародним проектом "Європейсько-український ступінь магістра з програмного забезпечення" (JEP 26182 2006) програми "Темпус" Європейської Комісії.

Робоча програма складена
кандидатом технічних наук, доцентом

Табунщик Г.В.

1 МЕТА І ЗАДАЧІ ДИСЦИПЛІНИ, ЇЇ МІСЦЕ В НАВЧАЛЬНОМУ ПРОЦЕСІ

1.1 Мета викладання дисципліни

Метою даного курсу є вивчення теоретичних основ та практичних аспектів менеджменту програмних систем. Дисципліна «Менеджмент програмних систем» спрямована на отримання студентом базових знань та практичних навичок з основ керування програмними проектами. Такі знання призначені для використання при розробці об'ємних корпоративних програмних додатків.

1.2 Завдання вивчення дисципліни

Внаслідок вивчення дисципліни студенти повинні вміти:

- розробляти загальний план проекту, що вимагає значного об'єму робіт;
- використовувати методи керування проектами, як «гнучкими» так і «традиційними»;
- ефективно оцінювати проектні витрати, використовуючи декілька різних методів одночасно;
- вимірювати прогрес проекту, продуктивність та інші аспекти процесу розробки програмного забезпечення;
- використовувати методи аналізу вартості(earned-value analysis);
- керувати ризиками, динамічно регулювати плани проекту;
- ефективно використовувати засоби керування конфігураціями та коректно використовувати процеси керування змінами;
- використовувати стандарти керування проектами.

1.3 Зв'язок з іншими дисциплінами

Дисципліна «Менеджмент програмних систем» базується на знаннях отриманих під час вивчення наступних дисциплін:

- технології програмування та створення програмних продуктів;
- економічна теорія;
- економіка підприємства.

Знання, що студенти отримають під час цієї дисципліни будуть використовуватися під час вивчення дисциплін:

- проектування, моделювання та аналіз інформаційних систем;
- модулі та метрики програмних систем;
- розробка гнучких систем.

2 ЗМІСТ ДИСЦИПЛІНИ

Дисципліна "Менеджмент програмних систем" вивчається на 5 курсі у 9 семестрі і завершується іспитом. Навчальним планом передбачено:

Всього:

- лекцій – 18 годин;
 - лабораторних робіт – 16 годин;
 - самостійної роботи – 74 години;
- Загальна кількість – 108 годин.

Таблиця 2.1 – Склад, обсяг і терміни виконання змістових модулів дисципліни “Менеджмент програмних систем”

Модулі (блоки змістових модулів)	Короткий зміст модулів	Розподіл навчального часу за елементами модуля (видами занять), години			Самостійна робота студента		Обсяг навантаження студента	
		Л	ЛР	ПР	СРВ	СРС	Години	Кредити
1	2	3	4	5	7	8	9	10
Основи керування проектом	Вступ	1	4			11	18	0,5
	Типи та види проектів	2						
	Організаційна структура проекту. Учасники проекту							
	Програмні засоби керування проектами							
Життєвий цикл програмного забезпечення	Основні етапи життєвого циклу програмних продуктів	2				7	9	0,25
	Порівнянні етапів життєвого циклу для різних типів програмних проектів							
Оцінювання програмного забезпечення	Методи оцінювання програмного забезпечення	2				16	18	0,5
Планування заходів проекту	Розробка WBS структури проекту	2	6			10	18	0,5
	Операції проекту. Метод критичного шляху.							
	Планування ресурсів проекту							
Моніторинг проекту	Вартісний аналіз проекту	2	3			4	9	0,25
	Методи прогнозування витрат							

Продовження таблиці 2.1

1	2	3	4	5	7	8	9	10
Керування ризиками проекту	Ситуації невизначеності та фактори ризику	2	3			13	18	0,5
	Класифікація ризиків.							
	Методи боротьби з ризиками							
Керування якістю проекту	Стандарти керування якістю	2				7	9	0,25
	Методи контролю та забезпечення якості							
	Стандарти ISO10006 та ISO 12207							
Людські фактори	Формування команди проекту. Мотивація. Розподілення зобов'язань	2				6	9	0,25
	Висновки							
Всього		18	16			74	108	3,00

2.1 Змістові модулі

2.1.1 Вступ

Мета та завдання дисципліни. Місце дисципліни в навчальному процесі.

Лекцій – 1 год.

Самостійна робота – 2 год.

2.1.2 Основи керування проектом

Типи та види проектів. Організаційна структура проекту. Учасники проекту

Лекцій – 2 год.

Лабораторна робота – 4 год

Самостійна робота – 9 год.

Література: [3, 4].

2.1.3 Життєвий цикл програмного забезпечення

Основні етапи життєвого циклу програмних продуктів. Порівнянні етапів життєвого циклу для різних типів програмних проектів (RUP, RAD, Agile)

Лекцій – 2 год.

Самостійна робота – 7 год.

Література: [1–3].

2.1.4 Оцінювання програмного забезпечення

Методи оцінювання програмного забезпечення

Лекцій – 2 год.

Самостійна робота – 16 год.

Література: [4].

2.1.5 Планування заходів проекту

Мета та план виконання проектних робіт, послідовні та паралельні події, мережна модель планування, додавання розмірності часу, ідентифікація критичного путі, діаграма діяльності.

Розподілення ресурсів: типи ресурсів, ідентифікація вимог до ресурсів, розробка критичного путі, розрахунки коштів, розробка графіків ресурсів, коштів та послідовностей дій.

Лекцій – 2 год.

Лабораторна робота – 6 годин

Самостійна робота – 10 год.

Література: [7, 11, 13].

2.1.6 Моніторинг проекту

Обов'язки, оцінка прогресу, визначення точок контролю, збір даних, візуалізація прогресу, моніторинг коштів, встановлення пріоритетів, керування характеристиками проекту, контроль внесених змін до проекту.

Лекцій – 2 год.

Лабораторна робота – 3 години

Самостійна робота – 4 год.

Література: [7, 11–13].

2.1.7 Керування ризиками проекту

Ситуації невизначеності та фактори ризику. Класифікація ризиків. Методи боротьби з ризиками. Керування ризиками.

Лекцій – 2 год.

Самостійна робота – 3

Лабораторна робота – 3 години

Література: [7, 11–13].

2.1.8 Керування якістю проекту

Стандарти керування якістю. Методи контролю та забезпечення якості. Стандарти ISO10006 та ISO 12207

Лекцій – 2 год.
Самостійна робота – 7 год.
Література: [3, 7, 8].

2.1.9 Людські фактори

Методологія і мотивація формування команди проекту, створення команди, методи забезпечення взаємозв'язків в середині команди, лідерство.

Лекцій – 2 год.
Самостійна робота – 6 год.
Література: [7, 9, 10].

2.1.10 Висновки

Аналіз Інтернет ресурсів, присвячених керуванням програмними проектами.

Лекцій – 1 год.
Література: [7, 9, 10].

2.2 Перелік лабораторних робіт

Лабораторна робота № 1 Основні можливості Microsoft Project 2003

Мета – Ознайомитись з основними можливостями Microsoft Project 2003, визначити мету та рамки проекту та навчитись будувати ієрархічну структуру проекту.

Обсяг – 4 години

Лабораторна робота № 2 Визначення діяльності

Мета – навчитись визначати види діяльності та будувати діаграму Гранта

Обсяг – 3 години

Лабораторна робота № 3 Визначення та розподілення ресурсів проекту

Мета – визначити ресурси та календар ресурсів

Обсяг – 3 години

Лабораторна робота № 4 Аналіз плану проектних робіт та вартість проекту

Мета – навчитись використовувати метод PERT для аналізу проектних робіт

Обсяг – 3 години

Лабораторна робота № 5 Моніторинг проекту

Мета – вивчити різні методи, що можуть бути використані для моніторингу.

Обсяг – 3 години

2.3 Теми для самостійної роботи студента

Нижче наведено перелік тем для самостійної роботи:

1. Програмні засоби для керування проектами – 2 години
2. Методи класифікації проектів – 5 годин
3. Метрики, що використовуються при керуванні якістю програмного проекту – 14 годин
4. Діаграма Ганта – 4 години
5. Життєвий цикл проектів відповідно до RAD – 3 години
6. Життєвий цикл Agile-проектів – 4 години
7. Характеристика учасників проекту – 2 години
8. Загальний зміст стандартів PMBOK та SWEBOOK – 7 годин
9. Математичні методи визначення та зменшення ризику – 10 годин
10. Методи вартісного аналізу – 16
11. Керування конфігураціями та змінами – 4
12. Формування команди проекту. Мотивація персоналу – 6 годин

3 НАВЧАЛЬНО-МЕТОДИЧНІ МАТЕРІАЛИ З ДИСЦИПЛІНИ

3.1 Література основна

1. Cotterell M., Hughes B. Software Project Management. London: International Tomson Publishing, 1995. – 276 p.
2. Kenneth R. Baine. Integrated IT Project Management: A Model-Centric Approach. Artech House. –2004. – 502p
3. Maciaszek L.A., Liong B.L. Practical software Engineering. A case study Approach. Addison Wesley. –2005. – 826p.
4. Stephen S. Bonham IT Project Portfolio Management. Artech House. – 2005. – 286p.
5. Paul E Harris. Planning and Control Using Microsoft® Project and PMBOK® Guide Third Edition. Eastwood Harris Pty Ltd. – 2005. – 300 p
6. Royse W. Software Project Management: a Unified FrameWork. Addison Wesley, 1998. – 448p. (Ройс У. Управление проектами по созданию программного обеспечения. М.: Лори. – 2002. – 424с.)
7. Беркун С. Искусство управления IT-проектами. –СПб.: Питер, – 2007. – 400с.
8. Кантор М. Управление программными проектами. Практическое руководство по разработке успешного программного обеспечения. М.: Вильямс, – 2002. – 176с.
9. <http://www.ipma.ch/Pages/IPMA.aspx> – International Project Management Association
10. <http://www.upma.kiev.ua/> – Ukrainian project management association "Ukrnet"

3.2 Додаткова література

11. Богданов В. Управление проектами в Microsoft Project 2002/ Учебный курс. – СПб.:Питер, 103. –640с.
12. Гультяев А. К. MS Project 2002. Управление проектами. Русифицированная версия: Самоучитель. – СПб.: КОРОНА принт, 2003. – 592 с.

3.3 Методичні матеріали з дисципліни

13. Методичні вказівки до виконання лабораторних з дисципліни “Управління програмними проектами” для студентів магістратури напряму підготовки 6.050103 “Програмна інженерія” спеціальності 8.080403 “Програмне забезпечення автоматизованих систем”/ Укл.: Г.В. Табунщик.– Запоріжжя: ЗНТУ, 2007. – 14 с.

3.4 Програмні засоби

Microsoft Project 2003, Microsoft Visio 2003

4 ПЕРЕЛІК ПИТАНЬ З ДИСЦИПЛІНИ

1. Дайте визначення проекту.
2. Виконайте класифікацію проектів за видом та типами.
3. Наведіть та дайте характеристику учасників проекту.
4. Життєвий цикл проекту.
5. Наведіть порівняльну характеристику життєвого циклу проектів для різних шаблонів проекту (RAD, RUP, Agile).
6. Функції менеджменту проектами
7. Процеси керування проектом.
8. Стандарти якості ISO 9000.
9. Стандарти якості програмних систем.
10. Ієрархічні структура проекту.
11. Які основні поняття використовують при конструюванні календарного графіку?
12. Модель перетворень для операції.
13. Календарі операцій.
14. Методи планування операцій.
15. Метод критичного шляху.
16. Класифікація ресурсів проекту, їх планування та керування їми.
17. Вартісна оцінка елементів проекту.
18. Методи моніторингу проектних процесів.
19. Класифікація ризиків проекту
20. Методи боротьби з ризиками. Метод Монте-Карло.
21. Методи боротьби з ризиками. Метод Pert.
22. Методи боротьби з ризиками. Дерево рішень.
23. Методи керування ризиками.
24. Імовірнісні оцінки показників ризику.
25. Мотивація персоналу проекту.
26. Розподілення зобов'язань.
27. Дайте характеристику програмних засобів, що використовуються при керуванні проектами

РОБОЧА НАВЧАЛЬНА ПРОГРАМА
дисципліни
“СУЧАСНИЙ ШТУЧНИЙ ІНТЕЛЕКТ”

для студентів магістратури напрямку підготовки

6.050103 “Програмна інженерія”

спеціальності **8.080403 “Програмне забезпечення автоматизованих систем”**

Факультет інформатики та обчислювальної техніки
Кафедра програмних засобів

Форма навчання	Курс	Семестр	Усього (год)	Лекції (год)	Лабор. роб. (год)	Практ. (год)	Самост. під керівн.(год)	Самост. (год)	КП, КР, РГЗ (сем.)	Залік (сем.)	Іспит (сем.)
Денна	V	9	108	18	16	–	–	74	9	–	9
Заочна											

Робоча програма складена на основі Навчального плану підготовки магістрів по спеціальності "Програмне забезпечення автоматизованих систем" напрямку 6.050103 “Програмна інженерія” від 17.07.2007 р. за Міжнародним проектом "Європейсько-український ступінь магістра з програмного забезпечення" (JEP 26182 2006) програми "Темпус" Європейської Комісії.

Робоча програма складена
кандидатом технічних наук, доцентом

Субботіним С.О.

1 МЕТА І ЗАДАЧІ ДИСЦИПЛІНИ, ЇЇ МІСЦЕ В НАВЧАЛЬНОМУ ПРОЦЕСІ

1.1 Мета викладання дисципліни

Засвоєння студентами основних теоретичних відомостей та практичних вмінь з курсу. Ознайомлення студентів з окремими галузями штучного інтелекту, зокрема, машинним навчанням. Подання студентам широкого кола методів та алгоритмів у контексті сприйняття та навчання. Підготувати студента до ефективного використання сучасних методів штучного інтелекту для створення автоматизованих систем у подальшій професійній діяльності; допомогти набути навички практичної роботи із програмними засобами для побудови інтелектуальних моделей.

1.2 Завдання вивчення дисципліни:

Внаслідок вивчення дисципліни студенти повинні:

- сформувати знання та отримати практичні навички для використання методів машинного навчання;
- отримати уяву про стан і перспективу розвитку теорії штучного інтелекту та програмного забезпечення для проектування й розробки інтелектуальних систем.

На основі вивчення дисципліни студент повинен
з н а т и:

- основні поняття та визначення теорії штучного інтелекту;
- методи машинного навчання та прийняття рішень у системах штучного інтелекту;
- сучасні програмні засоби для проектування і розробки систем штучного інтелекту;
- критерії порівняння моделей і методів штучного інтелекту.

в м і т и:

- обґрунтовувати й аналізувати вибір конкретного методу машинного навчання при вирішенні відповідних практичних задач;
- використовувати сучасні програмні засоби для проектування та дослідження систем штучного інтелекту;
- обирати та використовувати еволюційні методи для вирішення задач машинного навчання та оптимізації;
- створювати програми на мові макросів пакету MATLAB для побудови та використання сучасних інтелектуальних систем;
- аналізувати результати побудови та використання штучного інтелекту при вирішенні прикладних задач.

1.3 Зв'язок із іншими дисциплінами

Курс базується на поняттях, що вивчаються в дисциплінах:

- Математичні методи оптимізації та дослідження операцій;
- Обробка та інтерпретація соціально-економічної інформації;
- Математичні основи представлення знань;
- Системи штучного інтелекту;
- Нейроінформатика та еволюційні алгоритми;
- Теорія прийняття рішень.

Отриманні по розглянутій дисципліні знання будуть використовуватися та доповнюватися в курсах "Сучасні бази даних та інтелектуальний аналіз даних" та "Інтелектуальні агенти", а також у курсовому проектуванні.

2 ЗМІСТ ЛЕКЦІЙ (18 ГОДИН)

2.1 Основні принципи машинного навчання та їхнє використання в системах штучного інтелекту

Машинне навчання. Стратегії та методи навчання. Психологічні моделі вирішення проблем. Виведення на основі прикладів. Методи виведення. Дії та планування.

Лекцій – 8 год.

Самостійна робота – 30 год.

Література [1, 26, 27, 29, 32, 33, 35, 38, 41, 43, 44, 46, 47, 55–58, 60, 67–69, 71]

2.2 Еволюційні методи в задачах штучного інтелекту та оптимізації

Загальне уявлення про еволюційну оптимізацію. Еволюційні оператори. Застосування еволюційних методів в задачах машинного навчання. Вирішення оптимізаційних задач за допомогою еволюційних методів.

Лекцій – 10 год.

Лаб. роб. – 16 год.

Самостійна робота – 42 год.

Література [2–25, 28, 30, 31, 34, 36, 37,39, 40,42, 45, 48–54, 59,61–66, 70, 72]

3 ПЕРЕЛІК ЛАБОРАТОРНИХ РОБІТ

Лабораторна робота № 1. Методи еволюційного пошуку (6 год.)

Мета роботи: вивчити основні методи еволюційного пошуку, навчитися використовувати еволюційні методи для розв'язку оптимізаційних задач.

Лабораторна робота № 2. Статистичний аналіз результатів еволюційної оптимізації (4 год.)

Мета роботи: навчитися обирати оптимальні параметри для роботи еволюційних методів, ознайомитися із статистичними методами дослідження результатів роботи методів еволюційного пошуку.

Лабораторна робота № 3. Комбінаторна оптимізація за допомогою еволюційних методів (6 год.)

Мета роботи: вивчити основні еволюційні оператори, що призначені для вирішення комбінаторних задач, навчитися розв'язувати задачі комбінаторної оптимізації за допомогою методів еволюційного пошуку.

4 ПЕРЕЛІК ТЕМ ДО САМОСТІЙНОЇ РОБОТИ

1. Історія розвитку штучного інтелекту (4 год.).
2. Узагальнена модель інтелектуальної системи (2 год.).
3. Властивості інтелектуальних систем (2 год.).
4. Принципи вирішення задач штучного інтелекту у пакеті MATLAB (6 год.).
5. Класифікація методів машинного навчання (2 год.).
6. Чітке (булеве) логічне виведення (6 год.).
7. Нечітке виведення (4 год.).
8. Дії та планування (4 год.).
9. Класифікація та моделі еволюційних методів (6 год.).
10. Еволюційні оператори (6 год.).
11. Застосування еволюційних методів у задачах відбору інформативних ознак (6 год.).
12. Застосування еволюційних методів у задачах машинного навчання (6 год.).
13. Застосування еволюційних методів у задачах комбінаторного пошуку (6 год.).
14. Аналіз еволюційних методів (6 год.).
15. Еволюційні методи у пакеті MATLAB (6 год.).

Контроль самостійної роботи передбачає написання рефератів або контрольних робіт.

5 ТЕМИ КУРСОВИХ ПРОЕКТІВ

1. Методи відбору інформативних ознак.
2. Багатокритеріальні методи еволюційного пошуку.
3. Полімодальний еволюційний пошук.
4. Керування параметрами еволюційного пошуку.
5. Теорема схем. Теоретичний аналіз функціонування методів еволюційного пошуку.
6. Імунні системи.
7. Метод бджолої колонії та його застосування до синтезу моделей складних об'єктів та систем.
8. PSO-метод і багатокритеріальний PSO-метод та їх застосування до синтезу моделей складних об'єктів та систем.
9. Методи генетичного та еволюційного програмування.
10. Паралельні та багаторівневі еволюційні методи.
11. Методи синтезу інформативних ознак. Методи та критерії оцінки інформативності ознак.
12. Еволюційні стратегії та їх застосування до синтезу моделей складних об'єктів та систем.
13. Метод мурашиних колоній та його застосування до синтезу моделей складних об'єктів та систем.
14. Синтез моделей складних об'єктів та систем за допомогою інтелектуальних методів мультиагентної оптимізації.
15. Метод імітації відпалу та його застосування до синтезу моделей складних об'єктів та систем.

6 ПИТАННЯ З ДИСЦИПЛІНИ

1. Машинне навчання.
2. Стратегії та методи навчання.
3. Психологічні моделі вирішення проблем.
4. Виведення на основі прикладів.
5. Індуктивне виведення.
6. Дедуктивне виведення.
7. Чітке (булеве) логічне виведення.
8. Нечітке виведення.
9. Методи виведення.
10. Дії та планування.
11. Які методи відносять до еволюційних?
12. Порівняйте методи еволюційного пошуку з іншими методами оптимізації. В чому переваги еволюційних методів?
13. Проаналізуйте умови ефективного використання методів еволюційного пошуку.
14. Назвіть особливості еволюційних методів.
15. Які недоліки еволюційного пошуку та в чому вони полягають?
16. Дайте визначення основних термінів, що відносяться до теорії еволюційного пошуку: популяція, розмір популяції, число поколінь, хромосома, ген, локус, алель, фенотип, генотип.
17. Проаналізуйте узагальнену схему роботи еволюційних методів.
18. Наведіть послідовність виконання узагальненого еволюційного пошуку.
19. Які параметри необхідно визначати для роботи еволюційних методів?
20. Виконайте порівняльний аналіз канонічних моделей еволюційного пошуку.
21. В чому полягають особливості моделі Genitor?
22. Що таке гібридний еволюційний метод? Які існують стратегії взаємодії класичних та еволюційних методів?
23. Назвіть відмінності моделі СНС від класичних еволюційних методів.
24. Які особливості еволюційного методу із змінним часом життя хромосом?
25. Порівняйте мобільний еволюційний метод з класичними методами еволюційного пошуку. Для чого призначені оператори CUT та SPLICE?
26. Проаналізуйте паралельні та багаторівневі еволюційні методи.
27. Наведіть послідовність виконання еволюційного пошуку із зменшенням розміру популяції.
28. Які існують способи кодування параметрів, що оптимізуються, при використанні еволюційних методів?
29. Що таке фітнесс-функція?
30. Порівняйте стратегії створення початкової популяції.

31. Виконайте порівняльний аналіз операторів відбору (пропорційний відбір, відбір за допомогою ранжирування, турнірний відбір та відбір з використанням порогу).

32. Які способи формування батьківської пари використовуються в еволюційних методах?

33. Проаналізуйте оператори схрещування (n-точкове, рівномірне, порівняльне, арифметичне, діагональне).

34. Для чого призначений оператор мутації? Які оператори мутації використовуються в еволюційних методах?

35. Яким чином відбувається формування нового покоління?

36. Які критерії зупинення використовуються при еволюційному пошуку?

37. Для чого призначена теорема схем? Дайте визначення основних понять, що використовуються в теоремі схем. В чому полягає теорема Холанда про схеми?

38. Порівняйте еволюційні стратегії з генетичними алгоритмами та методом імітації відпалу.

39. Виконайте порівняльний аналіз генетичного та еволюційного програмування.

40. Проаналізуйте внутрішню структуру функції `ga` пакету Matlab: основні змінні, параметри, методи та допоміжні функції, їх призначення та використання.

41. Які параметри можна використовувати в функції `ga`? Яким чином вони задаються? Як отримати поточні параметри функції `ga`?

42. Проаналізуйте візуальний модуль для роботи з методами еволюційного пошуку `gatoool`: призначення, використання, параметри, візуальні компоненти, методи представлення результатів еволюційного пошуку.

43. З якою метою виконують статистичний аналіз результатів еволюційної оптимізації?

44. Порівняйте поняття генеральної сукупності та вибірки. Що таке об'єм вибірки? Яка вибірка репрезентативною?

45. Дайте визначення випадкової величини. Чим відрізняється дискретна випадкова величина від неперервної?

46. Що таке ряд розподілу випадкової величини? Як визначається кількість та ширина інтервалів при побудові ряду розподілу?

47. Чим відрізняються поняття “настройка параметрів” та “управління параметрами” еволюційного пошуку? Проаналізуйте методи управління параметрами еволюційного пошуку (адаптивні та не-адаптивні).

48. Поясніть, яким чином впливають ймовірності виконання еволюційних операторів відбору, схрещування та мутації на його ефективність.

49. Обґрунтуйте вплив кількості елітних хромосом, розміру популяції та максимально допустимої кількості ітерацій на результати еволюційного пошуку.

50. Проаналізуйте засоби пакету Matlab, що можуть бути використані для виконання статистичного аналізу результатів еволюційної оптимізації.

51. Який вид кодування хромосом використовується в еволюційних методах при розв'язку комбінаторних задач?
52. В чому полягає суть оператора впорядковуючого схрещування? Які відмінності одноточкового від двоточкового впорядковуючого схрещування?
53. Проаналізуйте схрещування із частковим відображенням.
54. Яким чином відбувається схрещування із частковим відображенням?
55. Наведіть послідовність виконання жадібного схрещування. Які переваги та недоліки такого оператора?
56. За рахунок чого реалізується схрещування методом дихотомії?
57. Чим відрізняється оператор сегрегації від інших операторів схрещування? Яким чином реалізується такий оператор?
58. Дайте порівняльну характеристику операторів схрещування, яку використовуються методах при розв'язку задач комбінаторної оптимізації.
59. Порівняйте класичну із одноточковою мутацією обміну.
60. Яка особливість мутації золотого розрізу?
61. Наведіть спільні та відмінні риси мутації золотого перетину та мутації на основі чисел Фібоначчі. Які вони мають переваги та недоліки?
62. В чому полягає нечітка мутація на основі методу дихотомії?
63. В яких випадках застосовується оператор мутації Монте-Карло?
64. Яка мета оператора інвертування?
65. Наведіть послідовність виконання класичного інвертування.
66. Проаналізуйте інвертування із зсувом.
67. Порівняйте інвертування з використанням методу Фібоначчі з іншими видами інвертування.
68. Які особливості має інвертування на основі методу золотого перетину?
69. Порівняйте оператори мутації негомологічних числових хромосом.
70. Чому оператори транслокації, вставки та делеції не набули широкого поширення? Яким чином вони виконуються? Порівняйте їх з іншими еволюційними операторами.
71. Проаналізуйте складність еволюційних методів (теоретично).
72. Чим відрізняються задачі комбінаторної від задач неперервної оптимізації?
73. В чому полягає задача комівояжера. Які існують методи її вирішення?
74. Проаналізуйте задачу складання оптимального розкладу та методи її вирішення.
75. Метод гілок та меж. Задача про рюкзак.
76. Порівняйте транспортну задачу з іншими задачами комбінаторної оптимізації.
77. Яким чином методи еволюційного пошуку можуть бути застосовані до розв'язку задачі відбору інформативних ознак?

7 НАВЧАЛЬНО-МЕТОДИЧНІ МАТЕРІАЛИ З ДИСЦИПЛІНИ

7.1 Література основна

1. Дубровин В.И., Субботин С.А., Богуслаев А.В., Яценко В.К. Интеллектуальные средства диагностики и прогнозирования надежности авиадвигателей: Монография. – Запорожье: ОАО "Мотор-Сич", 2003. – 279 с.
2. Дубровін В.І., Субботін С.О. Методи оптимізації та їх застосування в задачах навчання нейронних мереж: Навчальний посібник. – Запоріжжя: ЗНТУ, 2003. – 136 с.
3. Круглов В.В., Борисов В.В. Искусственные нейронные сети: теория и практика.–М.: Горячая линия-Телеком, 2001. – 382 с.
4. Рутковская Д., Пилиньский М., Рутковский Л. Нейронные сети, генетические алгоритмы и нечеткие системы: Пер. с польск. И.Д. Рудинского. – М.: Горячая линия-Телеком, 2004. – 452 с.

7.2 Література додаткова

5. Bryden K.M., McCorkle D.S. Evolutionary optimization of energy systems using population graphing and neural networks // Advances in Engineering Software. – 2004. – №35. – P. 289–299.
6. Cantu–Paz E. Efficient and Accurate Parallel Genetic Algorithms. – Massachusetts: Kluwer Academic Publishers, 2001. – 162p.
7. Gen M., Cheng R. Genetic algorithms and engineering design. – New Jersey: John Wiley & Sons, 1997. – 352 p.
8. Haupt R., Haupt S. Practical Genetic Algorithms. – New Jersey: John Wiley & Sons, 2004. – 261 p.
9. Hoffmann A.G. Paradigms of Artificial Intelligence: a methodological and computational analysis. – Singapore: Springer-Verlag, 1998. – 234 p.
10. Holland J.H. Adaptation in natural and artificial systems. – Ann Arbor: The University of Michigan Press, 1975. – 97 p.
11. Hyvarinen A., Karhunen J., Oja E. Independent Component Analysis. – New York: John Wiley & Sons, 2001. – 481 p.
12. Poli R. Exact schema theory for genetic programming and variable-length genetic algorithms with one-point crossover // Genetic Programming and Evolvable Machines. – Las Vegas: Morgan Kaufmann. – 2001. – P. 469–476.
13. Radcliffe N.J. Equivalence class analysis of genetic algorithms //Complex Systems. – 1990. – № 2. – P. 183–205.
14. Roshdy S.Y., Carla N.P. Combining genetic algorithms and neural networks to build a signal pattern classifier // Soft Computing Systems – Design, Management and Applications: Proceedings of the International Conference (1-4 December 2002). – Santiago: IOS Press, 2002. – P. 735–744.

15. Sarimveis H., Alexandridis A., Mazarakis S., Bafas G. A new algorithm for developing dynamic radial basis function neural network models based on genetic algorithms // *Computers and Chemical Engineering*. – 2004. – №28. – P. 209 – 217.
16. Siedlecki W., Sklansky J. A note on genetic algorithms for large-scale feature selection // *Pattern Recognition Letters*. – 1989. – №10. – P. 335–347.
17. Subbotin S., Oleynik A. Entropy Based Evolutionary Search for Feature Selection // *The experience of designing and application of CAD systems in Microelectronics: Proceedings of the IX International Conference CADSM-2007 (20–24 February 2007)*. – Lviv: Publishing house of Lviv Polytechnic, 2007. – P. 442–443.
18. Subbotin S., Oleynik A. The feature selection method based on the evolutionary approach with a fixation of a search space // *Modern problems of radio engineering, telecommunications and computer science: Proceedings of the IX International Conference TCSET'2006 (21–25 February 2006)*. – Lviv: Publishing house of Lviv Polytechnic, 2006. – P. 574–575.
19. *The Practical Handbook of Genetic Algorithms. Volume I. Applications* / Ed. by L.D. Chambers. – Florida: CRC Press, 2000. – 520 p.
20. *The Practical Handbook of Genetic Algorithms. Volume II. New Frontiers* / Ed. by L.D. Chambers. – Florida: CRC Press, 2000. – 421 p.
21. *The Practical Handbook of Genetic Algorithms. Volume III. Complex Coding Systems* / Ed. by Lance D. Chambers. – Florida: CRC Press LLC, 2000. – 659p.
22. Vafaie H., Imam I. Feature Selection Methods: Genetic Algorithms vs. Greedy-like Search // *Fuzzy and Intelligent Control Systems: Proceedings of the Third International Conference (22–26 March 1994)*. – Louisville: IST, 1994. – P. 381–390.
23. Whitley D. A Genetic Algorithm Tutorial // *Statistics and Computing*. – 1994. – № 4. – P. 65–85.
24. Yang J., Honavar V. Feature subset selection using a genetic algorithm // *Genetic Programming: Proceedings of the II International Conference GP-97 (13-16 July 1997)*. – Stanford: Publishing house of Stanford University, 1997. – P. 58–63.
25. Zhengjun L., Aixia L., Changyao W., Zheng N. Evolving neural network using real coded genetic algorithm for multispectral image classification // *Future Generation Computer Systems*. – 2004. – №20. – P. 1119–1129.
26. Андрейчиков А.В., Андрейчикова О.Н. Интеллектуальные информационные системы: Учебник. – М.: Финансы и статистика, 2004. – 424 с.
27. Белецкий Б.А., Вагис А.А., Васильев С.В., Гупал Н.А. Сложность байесовской процедуры индуктивного вывода. Дискретный случай // *Проблемы управления и информатики*. – 2006. – № 6. – С. 55–70.
28. Богуслаев А.В., Олейник А.А., Пухальская Г.В., Субботин С.А. Отбор геометрических параметров и синтез модели частотной характеристики лопаток компрессора на основе эволюционного поиска // *Вісник двигунобудування*. – 2006. – № 1. – С. 14–17.
29. Бондарев В.Н., Аде Ф.Г. Искусственный интеллект. – Севастополь: СевНТУ, 2002. – 615 с.

30. Васильев В.И., Ильясов Б.Г. Интеллектуальные системы управления с использованием генетических алгоритмов // Информационные технологии. – 2002. – №12 – С. 29–34.
31. Вороновский Г.К., Махотило К.В., Петрашев С.Н., Сергеев С.А. Генетические алгоритмы, искусственные нейронные сети и проблемы виртуальной реальности.–Харьков: Основа, 1997.– 112 с.
32. Галуев Г.А. Интеллектуальные среды нового поколения: состояние проблемы и перспективы решения // Искусственный интеллект. – 2004. – № 3. – С. 523–533.
33. Герасимов Б.М., Тарасов В.А., Токарев И.В. Человеко-машинные системы принятия решений с элементами искусственного интеллекта. – К.: Наукова думка, 1993. – 180 с.
34. Герман О.В. Получение выводов в противоречивых системах // Кибернетика и системный анализ. – 2005. – № 5. – С. 29–41.
35. Дюк В., Самойленко А. Data mining: учебный курс.–СПб.: Питер, 2001.– 368 с.
36. Дьяконов В. MATLAB 6: учебный курс. – СПб.: Питер, 2001. – 592 с.
37. Емельянов В.В. Курейчик В.В. Курейчик В.М. Теория и практика эволюционного моделирования. – М.: ФИЗМАТЛИТ, 2003. – 432с.
38. Зайченко Ю.П. Основи проектування інтелектуальних систем. Навчальний посібник. – К.: Слово, 2004. – 352 с.
39. Кисляков А.В. Генетические алгоритмы: математический анализ некоторых схем репродукции // Информационные технологии. – 2000. – № 12. – С. 9–14.
40. Кисляков А.В. Генетические алгоритмы: операторы скрещивания и мутации // Информационные технологии. – 2001. – № 1. – С. 29–34.
41. Кричевский М.Л. Интеллектуальные методы в менеджменте. – СПб.: Питер, 2005. – 304 с.
42. Курейчик В.М. Генетические алгоритмы: Монография. – Таганрог: ТРТУ, 1998. – 242 с.
43. Лорьер Ж.-Л. Системы искусственного интеллекта: Пер. с франц.. – М.: Мир, 1991. – 568 с.
44. Люгер Дж.Ф. Искусственный интеллект: стратегии и методы решения сложных проблем / Пер. с англ. – М.: Вильямс, 2005. – 864 с.
45. Митюшкин Ю.И., Мокин Б.И., Ротштейн А.П. Soft Computing: идентификация закономерностей нечеткими базами знаний. – Винница: УНИВЕРСУМ-Винница, 2002. – 145 с.
46. Нильсон Н. Принципы искусственного интеллекта / Пер. с англ. – М.: Радио и связь, 1985. – 376 с.
47. Нильсон Н.Дж. Искусственный интеллект. Методы поиска решений. – М.: Мир, 1973. – 270 с.
48. Олейник А.А., Субботин С.А. Метод эволюционного поиска с сокращением размера популяции // Сучасні проблеми і досягнення в галузі радіотехніки, телекомунікацій та інформаційних технологій: Матеріали

Міжнародної науково-практичної конференції (13–15 квітня 2006 р.). – Запоріжжя: ЗНТУ, 2006. – С. 179–181.

49. Олейник Ал.А., Олейник Ан.А., Субботин С.А., Яценко В.К. Синтез моделей коеффициента упрочнения деталей авиадвигателей после алмазного выглаживания на основе оптимизационного подхода // Вісник двигунобудування. – 2005. – №3. – С. 25–30.

50. Олейник Ан.А. Эволюционный отбор заданного количества признаков // Радіоелектроніка і молодь в ХХІ сторіччі: Матеріали 11-го міжнародного молодіжного форуму (10–12 квітня 2007 р.). – Харків: ХНУРЕ, 2007. – Ч. 2. – С.142.

51. Олейник Ан.А., Олейник Ал.А. Эволюционный синтез моделей сложных объектов и процессов // Радіоелектроніка і молодь в ХХІ сторіччі: Матеріали 10-го міжнародного молодіжного форуму (10–12 квітня 2006 р.). – Харків: ХНУРЕ, 2006. – С. 417.

52. Олейник Ан.А. Выбор системы информативных признаков для классификации транспортных средств на основе эволюционного поиска // Комп'ютерне моделювання та інтелектуальні системи: Збірник наукових праць / За рад. Д.М. Пізи, С.О. Субботіна. – Запоріжжя: ЗНТУ, 2007. – С. 134–146.

53. Осовский С. Нейронные сети для обработки информации. – М.: Финансы и статистика, 2004. – 344 с.

54. Пат. 18294 Україна, МПК G06F 19/00. Спосіб відбору інформативних ознак для діагностики виробів / С.О. Субботін, А.О. Олійник; Запорізький національний технічний університет. – № u200603087; Заявл. 22.03.06; Опубл. 15.11.06, Бюл. № 11. – 4 с.

55. Поспелов Г.С. Искусственный интеллект – основа новой информационной технологии. – М.: Наука, 1988. – 280 с.

56. Поспелов Г.С., Поспелов Д.А. Искусственный интеллект – прикладные системы. – М.: Знание, 1985. – 48 с.

57. Поспелов Д.А. Моделирование рассуждений. Опыт анализа мыслительных актов. – М.: Радио и связь, 1989. – 184 с.

58. Рассел С., Норвиг П. Искусственный интеллект: современный подход, 2-е изд.: Пер с англ.. – М.: Вильямс. – 2006. –1408 с.

59. Риоло Р.Л. Естественный отбор в мире битов // В мире науки. – 1992. – № 9. – С. 160–163.

60. Рідкокаша А.А., Голдер К.К. Основи систем штучного інтелекту. Навчальний посібник. – Черкаси: "ВІДЛУННЯ-ПЛЮС", 2002. – 240 с.

61. Ротштейн А.П. Интеллектуальные технологии идентификации: нечеткие множества, генетические алгоритмы, нейронные сети. – Винница: УНІВЕРСУМ-Вінниця, 1999. – 320 с.

62. Сергиенко И.В., Гупал А.М. Принципы построения процедур индуктивного вывода // Кибернетика и системный анализ. – 2006. – № 4. – С. 51-63.

63. Субботин С.А., Олейник А.А. Ускоренный метод эволюционного отбора признаков // Автоматика-2006: Тези доповідей тринадцятої міжнародної

науково-технічної конференції (25–28 вересня 2006 р.). – Вінниця: УНІВЕРСУМ–Вінниця, 2006. – С. 409.

64. Субботин С.А., Олейник А.А. Выбор набора информативных признаков для синтеза моделей объектов управления на основе эволюционного поиска с группировкой признаков // Искусственный интеллект. – 2006. – № 4. – С. 488-494.

65. Субботин С.А., Олейник А.А. Модифицированный оператор жадного кроссовера для эволюционного отбора информативных признаков // Нейроинформатика и ее приложения: Материалы XIV Всероссийского семинара (6–8 октября 2006 г.). – Красноярск: ИВМ СО РАН, 2006. – С. 116-118.

66. Субботин С.А., Олейник А.А. Критерии сравнения эволюционных методов синтеза нейромоделей // Нейроинформатика-2007: Материалы IX Всероссийской научно-технической конференции (23–26 января 2007 г.). – М.: МИФИ, 2007. – Ч. 2. – С. 177-184.

67. Тимофеев А.В. Роботы и искусственный интеллект. – М.: Наука, 1978. – 192 с.

68. Усков А.А., Кузьмин А.В. Интеллектуальные технологии управления. Искусственные нейронные сети и нечеткая логика. – М.: Горячая линия-Телеком, 2004. – 143 с.

69. Хант Э. Искусственный интеллект / под ред. Стефанюка. – М.: Мир, 1978. – 558 с.

70. Холланд Х.Д. Генетические алгоритмы // В мире науки. – 1992. – № 9. – С. 32–40.

71. Ямпольський Л.С., Лавров О.А. Штучний інтелект у плануванні та управлінні виробництвом. – К.: Вища школа, 1995. – 255 с.

72. Ярушкина Н.Г. Основы теории нечетких и гибридных систем. – М.: Финансы и статистика, 2004. – 320 с.

РОБОЧА НАВЧАЛЬНА ПРОГРАМА

дисципліни

“ІНТЕЛЕКТУАЛЬНІ АГЕНТИ”

для студентів магістратури напряму підготовки

6.050103 “Програмна інженерія”

спеціальності 8.080403 “Програмне забезпечення автоматизованих систем”

Факультет інформатики та обчислювальної техніки

Кафедра програмних засобів

Форма навчання	Курс	Семестр	Усього (год)	Лекції (год)	Лаб. роб. (год)	Практ. (год)	Самост. під керівн. (год)	Самост. (год)	КП, КР, РГЗ (сем.)	Залік (сем.)	Іспит (сем.)
Денна	V	9	108	18	16	–	–	74	–	9	–
Заочна											

Робоча програма складена на основі Навчального плану підготовки магістрів по спеціальності "Програмне забезпечення автоматизованих систем" напряму 6.050103 "Програмна інженерія" від 17.07.2007 р. за Міжнародним проектом "Європейсько-український ступінь магістра з програмного забезпечення" (JER 26182 2006) програми "Темпус" Європейської Комісії.

Робоча програма складена
кандидатом технічних наук, доцентом

Субботіним С.О.

1 МЕТА І ЗАДАЧІ ДИСЦИПЛІНИ, ЇЇ МІСЦЕ В НАВЧАЛЬНОМУ ПРОЦЕСІ

1.1 Мета викладання дисципліни

Засвоєння студентами основних теоретичних відомостей та практичних вмінь з курсу. Підготувати студента до ефективного використання сучасних засобів створення інтелектуальних програмних агентів; допомогти набутти навички практичної роботи із програмними засобами для проектування та розробки інтелектуальних агентних систем.

1.2 Завдання вивчення дисципліни:

Внаслідок вивчення дисципліни студенти повинні:

- сформувати знання та отримати практичні навички для використання мультиагентних систем та інтелектуальних агентів при вирішенні практичних задач;
- отримати уяву про стан і перспективу розвитку агентних технологій та програмного забезпечення для проектування й розробки агентних систем.

На основі вивчення дисципліни студент повинен

з н а т и:

- основні поняття та визначення агентних систем;
- моделі та методи побудови агентних систем та їхні властивості;
- сучасні програмні засоби для проектування і розробки агентних систем;
- критерії порівняння моделей агентів.

в м і т и:

- обґрунтовувати й аналізувати вибір конкретного типу моделі та методу розробки агентних систем при вирішенні відповідних практичних задач;
- використовувати сучасні програмні засоби для проектування та дослідження агентних систем;
- створювати програми на мові макросів пакету MATLAB для побудови та використання сучасних агентних систем;
- аналізувати результати побудови та використання агентних систем при вирішенні прикладних задач.

1.3 Зв'язок із іншими дисциплінами

Курс базується на поняттях, що вивчаються в дисциплінах:

- Математичні методи оптимізації та дослідження операцій;
- Обробка та інтерпретація соціально-економічної інформації;
- Математичні основи представлення знань;
- Системи штучного інтелекту;
- Нейроінформатика та еволюційні алгоритми;
- Теорія прийняття рішень.

Отриманні по розглянутій дисципліні знання будуть використовуватися та доповнюватися в курсах "Сучасні бази даних та інтелектуальний аналіз даних" та "Сучасний штучний інтелект", а також у курсовому проектуванні.

2 ЗМІСТ ЛЕКЦІЙ (18 ГОДИН)

2.1 Основні поняття та визначення інтелектуальних агентних систем

Агенти як інструмент комп'ютерних наук. Узагальнена модель та класифікація інтелектуальних агентів. Анатомія агента. Архітектури інтелектуальних агентів. Моделювання поведінки. Властивості програмних агентів. Розумні агенти. Поняття штучного життя. Моделювання харчових ланцюгів. Енергія та метаболізм агента. Відтворення агента. Смерть агента. Зв'язок та взаємодія між агентами. Основні поняття про програмних агентів. Web-агент. Розробка Web-агента.

Лекцій – 10 год.

Лаб. роб. – 10 год.

Самостійна робота – 36 год.

Література [2, 5, 7]

2.2 Swarm Intelligence – біонічні мультиагентні інтелектуальні системи

Метод мурашиної колонії. Застосування методу мурашиних колоній у задачах комбінаторного пошуку. Метод бджолоїної колонії. Застосування методу бджолоїної колонії для вирішення оптимізаційних задач. Відбір інформативних ознак в задачах діагностики: застосування Swarm Intelligence.

Лекцій – 8 год.

Лаб. роб. – 6 год.

Самостійна робота – 38 год.

Література [1, 3, 4, 6, 8–15]

3 ПЕРЕЛІК ЛАБОРАТОРНИХ РОБІТ

Лабораторна робота № 1. Метод мурашиних колоній (6 год.).

Мета роботи: вивчити метод мурашиних колоній, навчитися розв'язувати задачі комбінаторної оптимізації за допомогою методу мурашиних колоній.

Лабораторна робота № 2. Моделювання природних систем (4 год.).

Мета роботи: навчитися моделювати природні системи, вивчити методи, які застосовуються до моделювання природних систем.

Лабораторна робота № 3. Розробка Web-агента (6 год.).

Мета роботи: вивчити основні принципи роботи програмних агентів, навчитися створювати Web-агента, який взаємодіє із певними службами в середовищі Internet.

4 ПЕРЕЛІК ТЕМ ДО САМОСТІЙНОЇ РОБОТИ

1. Поняття штучного життя (3 год.).
2. Моделювання харчових ланцюгів (3 год.).
3. Опис моделі та навколишнє середовище (3 год.).
4. Анатомія агента (3 год.).
5. Енергія та метаболізм агента (3 год.).
6. Відтворення агента (3 год.).
7. Смерть агента (3 год.).
8. Web-агент (3 год.).
9. Розробка Web-агента (3 год.).
10. Основні поняття про програмні агенти (3 год.).
11. Властивості програмних агентів (3 год.).
12. Розумні агенти (3 год.).
13. Застосування Swarm Intelligence (8 год.).
14. Метод мурашиної колонії (7 год.).
15. Метод бджолоїної колонії (7 год.).
16. Програмні засоби для оптимізації на основі методу мурашиної колонії (8 год.).
17. Програмні засоби для оптимізації на основі методу бджолоїної колонії (8 год.).

Контроль самостійної роботи передбачає написання рефератів або контрольних робіт.

5 НАВЧАЛЬНО-МЕТОДИЧНІ МАТЕРІАЛИ З ДИСЦИПЛІНИ

5.1 Література основна

1. Subbotin S., Oleynik A. Modifications of Ant Colony Optimization Method for Feature Selection // The experience of designing and application of CAD systems in Microelectronics: Proceedings of the IX International Conference CADSM – 2007 (20–24 February 2007). – Lviv: Publishing house of Lviv Polytechnic, 2007. – P. 493–494.
2. Джонс М.Т. Программирование искусственного интеллекта в приложениях. – М.: ДМК Пресс, 2004. – 312 с.
3. Дубровин В.И., Субботин С.А., Богуслаев А.В., Яценко В.К. Интеллектуальные средства диагностики и прогнозирования надежности авиадвигателей: Монография. – Запорожье: ОАО "Мотор – Сич", 2003. – 279 с.
4. Дубровін В.І., Субботін С.О. Методи оптимізації та їх застосування в задачах навчання нейронних мереж: Навч. пос. – Запоріжжя: ЗНТУ, 2003. – 136 с.
5. Люгер Дж.Ф. Искусственный интеллект: стратегии и методы решения сложных проблем / Пер. с англ.. – М.: Вильямс, 2005. – 864 с.
6. Олейник Ал.А. Сравнительный анализ методов оптимизации на основе метода муравьиных колоний // Комп'ютерне моделювання та інтелектуальні системи: Збірник наукових праць / За ред. Д.М. Пізи, С.О. Субботіна. – Запоріжжя: ЗНТУ, 2007. – С. 147–159.
7. Рассел С., Норвиг П. Искусственный интеллект: современный подход, 2 – е изд.: Пер с англ. – М.: Вильямс. – 2006. – 1408 с.

5.2 Література додаткова

8. Bullnheimer B., Hartl R.F., Strauss C. Applying the ant system to the vehicle routing problem // Meta – Heuristics: Advances and Trends in Local Search Paradigms for Optimization. – Boston: Kluwer, 1998. – P. 109–120.
9. Costa D., Hertz A. Ants can colour graphs // Journal of the Operational Research Society. – 1997. – №48. – P. 295–305.
10. Dorigo M. Optimization, Learning and Natural Algorithms. – Milano: Politecnico di Milano, 1992. – 140 p.
11. Dorigo M., Gambardella L.M. Ant colonies for the traveling salesman problem // BioSystems. – 1997. – №43. – P. 73–81.
12. Gambardella L.M., Dorigo M. HAS – SOP: An hybrid ant system for the sequential ordering problem. – Lugano: CH, 1997. – P. 237–255.
13. Gambardella L.M., Taillard E., Agazzi G. Macs – vrptw: A multiple ant colony system for vehicle routing problems with time windows // New Methods in Optimisation. – McGraw – Hill, 1999. – P. 63–79.

14. Maniezzo V. Exact and approximate nondeterministic tree – search procedures for the quadratic assignment problem. – Bologna: Universita di Bologna, 1998 – 102 p.

15. Michel R., Middendorf M. An ACO algorithm for the shortest common supersequence problem // New Methods in Optimisation. – McGraw – Hill, 1999. – P. 525–537.

РОБОЧА НАВЧАЛЬНА ПРОГРАМА

дисципліни

“Моделі та метрика програмних систем”

для студентів магістратури напряму підготовки

6.050103 “Програмна інженерія”

спеціальності 8.080403 “Програмне забезпечення автоматизованих систем”

Факультет інформатики та обчислювальної техніки

Кафедра програмних засобів

Форма навчання	Курс	Семестр	Усього (год)	Лекції (год)	Лабор. роб. (год)	Практ. (год)	Самост. під керівн.(год)	Самост. (год)	КП, КР, РГЗ (сем.)	Залік (сем.)	Іспит (сем.)
Денна	V	10	108/3	20	20	-	-	68	-	-	10
Заочна											

Робоча програма складена на основі Навчального плану підготовки магістрів по спеціальності "Програмне забезпечення автоматизованих систем" напряму 6.050103 "Програмна інженерія" від 17.07.2007 р. за Міжнародним проектом "Європейсько-український ступінь магістра з програмного забезпечення" (JEP 26182 2006) програми "Темпус" Європейської Комісії.

Робоча програма складена
кандидатом технічних наук, доцентом

Табунщик Г.В.

1 МЕТА І ЗАДАЧІ ДИСЦИПЛІНИ, ЇЇ МІСЦЕ В НАВЧАЛЬНОМУ ПРОЦЕСІ

1.1 Мета викладання дисципліни

Метою даного курсу є отримання теоретичних основ та практичних навичок з оцінки та забезпечення якості програмних продуктів. Дисципліна "Моделі та метрика програмних систем" спрямована на поглиблення знань та практичних навичок студентів з сучасної технології створення складних програмних продуктів на базі ідей і принципів об'єктно-орієнтованого методу. Такі знання призначені для використання на всіх етапах життєвого циклу корпоративних програмних додатків з урахуванням сучасних вимог у відношенні до надійності, якості та ефективності програмних продуктів, що створюються.

1.2 Завдання вивчення дисципліни:

На основі вивчення дисципліни "Моделі та метрика програмних систем" студент повинен

з н а т и:

- етапи життєвого циклу програмного забезпечення;
- метрики програмного забезпечення;
- методи керування якістю відносно до розробки проектування та експлуатації програмних продуктів;
- CASE-засоби для тестування програмного забезпечення.

в м і т и:

- оцінювати якість та ефективність програмного забезпечення при рішенні завдань розробки;
- перевірки коректності;
- тестування;
- вибору та вдосконалення програмного забезпечення.

1.3 Зв'язок з іншими дисциплінами

Курс базується на поняттях, що вивчаються в дисциплінах «Проектування, моделювання та аналіз інформаційних систем» та «Менеджмент програмних систем».

2 ЗМІСТ ДИСЦИПЛІНИ

Навчальний процес в ЗНТУ в умовах кредитно-модульної системи організації навчання здійснюється у таких організаційних формах: навчальні заняття, виконання індивідуальних завдань, самостійна робота студентів, практична підготовка, контрольні заходи. Основні види занять, що входять до складу модулів: лекція, лабораторне, семінарське заняття, консультація та самостійна робота під керівництвом викладача.

Дисципліна " **Моделі та метрика програмних систем** " вивчається на 5 курсі у 10 семестрі і завершується іспитом. Навчальним планом передбачено:

Всього:

- лекцій – 20 годин;
- лабораторних робіт – 20 годин;
- самостійної роботи – 68 годин;

Загальна кількість аудиторних годин – 108 годин.

В умовах кредитно-модульної системи навчальна дисципліна “Розробка програмних систем” відповідає 3 заліковим кредитам.

Дисципліна складається з 5 блоків змістових модулів – основних її розділів, які разом зі змістовими модулями наведені у таблиці 2.1.

Таблиця 2.1 – Склад, обсяг і терміни виконання змістових модулів дисципліни “Моделі та метрика програмних систем”

Модулі (блоки змістових модулів)	Короткий зміст модулів	Розподіл навчального часу за елементами модуля (видами занять), години			Самостійна робота студента		Обсяг навантаження студента	
		Л	ЛР	ПР	СРВ	СРС	Години	Кредити
1	2	3	4	5	7	8	9	10
1 Основні поняття керування та забезпечення якості програмного забезпечення	Основні поняття загального керування якістю (TQM)	4	4			14	22	0,6
	Якість програмного забезпечення							
	Основні поняття теорії вимірювань							
2 Основні метрики програмного забезпечення	Мультиплікативні метрики	4	4			13	21	0,6
	Динамічні метрики							
	Метрики проектів							
3 Методи керування якістю ПЗ	Сім простих методів керування якістю	4	4			14	22	0,6
	Прості методи при керуванні якістю ПЗ							

Продовження таблиці 2.1

1	2	3	4	5	7	8	9	10
4 Метрики та вимірювання для об'єктно-орієнтованих систем	Метрики для оцінювання проектної складності проекту, оцінки якості, продуктивності	4	4			13	21	0,6
5 Методи оцінки якості програмного забезпечення	Оцінювання якості проекту	4	4			14	22	0,6
	Оцінювання якості процесу							
Всього		20	20			68	108	3

Позначення у таблиці 2.1: Л – лекції, ЛР – лабораторні роботи, ПР – практична робота, СРС – самостійна робота студента

2.1 Змістові модулі

2.1.1 Основні поняття керування та забезпечення якості програмного забезпечення

Основні поняття загального керування якістю (TQM).

Життєвий цикл програмного забезпечення. Якість програмного забезпечення. Стандарти для керування якістю програмного забезпечення.

Основні поняття теорії вимірювань: основні визначення, шкали вимірювань, надійність ті точність, помилки вимірювань, методи визначення взаємозв'язків.

Лекцій – 4 год.

Лабораторна робота – 4 год.

Самостійна робота – 14 год.

Література: [1, 5].

2.1.2 Основні метрики програмного забезпечення

Мультиплікативні метрики: середній час до виникнення помилки, щільність помилок, проблеми та задоволеність користувачів.

Динамічні метрики, метрики етапу експлуатації та підтримки програмного забезпечення.

Лекцій – 4 год.

Лабораторна робота – 4 год.

Самостійна робота – 13 год.

Література: [1–3, 5].

2.1.3 Основні методи при керуванні якістю ПЗ

Сім простих методів керування якістю: причинно-наслідкова діаграма, гістограми, контрольні карти, кореляційні таблиці.

Використання простих методів для керування якістю програмних засобів.

Лекцій – 4 год.

Лабораторна робота – 4 год.

Самостійна робота – 14 год.

2.1.4 Метрики та вимірювання для об'єктно-орієнтованих систем

Метрики для оцінювання проектної складності проекту, оцінки якості, продуктивності. Використання емпіричних метрик оцінювання якості програм.

Лекцій – 4 год.

Лабораторна робота – 4 год.

Самостійна робота – 13 год.

Література: [1–3].

2.1.5 Методи оцінки якості програмного забезпечення

Оцінювання якості процесу

Лекцій – 4 год.

Лабораторна робота – 4 год.

Самостійна робота – 14 год.

Література: [1, 4, 5].

2.2 Перелік лабораторних робіт

Мета проведення лабораторних занять посідає в придбанні досвіду оцінювання якості корпоративних програмних додатків

Лабораторна робота № 1.

Мета роботи: ознайомитись з case-засобами призначеними для керування якістю програмних проектів

Обсяг – 4 год.

Лабораторна робота № 2.

Мета роботи: навчитись розраховувати характеристик якості програмного забезпечення відповідно до метрик Холстеда

Обсяг – 4 год.

Лабораторна робота № 3.

Мета роботи: навчитись використовувати прості методи для контролю якості програмних продуктів

Обсяг – 4 год.

Лабораторна робота № 4.

Мета роботи: навчитись використовувати емпіричні метрики оцінювання якості програм.

Обсяг – 4 год.

Лабораторна робота № 5.

Мета роботи: навчитись реалізувати розподілену обробку даних

Обсяг – 4 год.

2.3 Перелік тем для самостійної роботи

Нижче наведено перелік тем для самостійної роботи студентів:

- | | |
|---|---------|
| 1. Життєвий цикл програмного забезпечення.
Різновид. Порівняння. | 13 год. |
| 2. Метрики оцінювання задоволеності споживача | 14 год. |
| 3. Контрольні карти | 13 год. |
| 4. Емпіричні методи | 14 год. |
| 5. Стандарти керування якістю програмного забезпечення | 13 год. |

3 НАВЧАЛЬНО-МЕТОДИЧНІ МАТЕРІАЛИ З ДИСЦИПЛІНИ

3.1 Література основна

1. Hutcheson M. L. Software Testing Fundamentals: Methods and Metrics. John Wiley & Sons, – 2003. –408 p.
2. Kan H.N. Metrics and Models in Software Quality Engineering, Second Edition, 2002. – 560.
3. Pandian C. R. Software Metrics: A Guide to Planning, Analysis, and Application Auerbach Publications, 2004. –312 p.

3.2 Література додаткова

4. IEEE Standard for a Software Quality Metrics Methodology Copyright ©, Institute of Electrical and Electronics Engineers, Inc. 2005.

3.3 Методичні матеріали з дисципліни

Методичні вказівки до виконання лабораторних робіт з дисципліни "Моделі та метрика програмних систем" для студентів для студентів магістратури напряму підготовки 6.050103 "Програмна інженерія" спеціальності 8.080403 "Програмне забезпечення автоматизованих систем" / Укл. Табунщик Г.В.– Запоріжжя: ЗНТУ, 2007.– 20 с.

РОБОЧА НАВЧАЛЬНА ПРОГРАМА
дисципліни
“СУЧАСНІ БАЗИ ДАНИХ ТА ІНТЕЛЕКТУАЛЬНИЙ АНАЛІЗ ДАНИХ”
для студентів магістратури напряму підготовки
6.050103 “Програмна інженерія”
спеціальності **8.080403 “Програмне забезпечення автоматизованих систем”**

Факультет інформатики та обчислювальної техніки
Кафедра програмних засобів

Форма навчання	Курс	Семестр	Усього (год)	Лекції (год)	Лабор. роб. (год)	Практ. (год)	Самост. під керівн.(год)	Самост. (год)	КП, КР, РГЗ (сем.)	Залік (сем.)	Іспит (сем.)
Денна	V	9	72	10	10	–	–	52	–	–	10
Заочна											

Робоча програма складена на основі Навчального плану підготовки магістрів по спеціальності "Програмне забезпечення автоматизованих систем" напряму 6.050103 "Програмна інженерія" від 17.07.2007 р. за Міжнародним проектом "Європейсько-український ступінь магістра з програмного забезпечення" (JEP 26182 2006) програми "Темпус" Європейської Комісії.

Робоча програма складена
кандидатом технічних наук, доцентом

Субботіним С.О.

1 МЕТА І ЗАДАЧІ ДИСЦИПЛІНИ, ЇЇ МІСЦЕ В НАВЧАЛЬНОМУ ПРОЦЕСІ

1.1 Мета викладання дисципліни

Засвоєння студентами основних теоретичних відомостей та практичних вмінь з курсу. Підготувати студента до ефективного використання сучасних методів інтелектуального аналізу даних та обробки інформації з використанням м'яких обчислень задля створення автоматизованих систем у подальшій професійній діяльності; допомогти набути навички практичної роботи із програмними засобами для інтелектуального аналізу даних.

1.2 Завдання вивчення дисципліни

Внаслідок вивчення дисципліни студенти повинні:

- сформулювати знання та отримати практичні навички для використання методів і моделей інтелектуального аналізу даних при вирішенні задач обробки великих масивів даних;
- отримати уяву про стан і перспективу розвитку методів м'яких обчислень як напряму теорії штучного інтелекту.

На основі вивчення дисципліни студент повинен

з н а т и:

- основні поняття та визначення інтелектуального аналізу даних;
- моделі та методи побудови моделей та аналізу залежностей у великих масивах даних;
- сучасні програмні засоби для проектування і розробки систем інтелектуального аналізу даних;
- критерії порівняння моделей і методів інтелектуального аналізу даних.

в м і т и:

- обґрунтовувати й аналізувати вибір конкретного типу моделі та методу інтелектуального аналізу даних при вирішенні відповідних практичних задач;
- використовувати сучасні програмні засоби для проектування та дослідження систем інтелектуального аналізу даних;
- створювати програми на мові макросів пакету MATLAB для побудови та використання моделей на основі м'яких обчислень;
- аналізувати результати побудови та використання систем інтелектуального аналізу даних при вирішенні прикладних задач.

1.3 Зв'язок із іншими дисциплінами

Курс базується на поняттях, що вивчаються в дисциплінах:

- Організація баз даних і знань;
- Математичні методи оптимізації та дослідження операцій;
- Обробка та інтерпретація соціально-економічної інформації;
- Математичні основи представлення знань;
- Системи штучного інтелекту;
- Нейроінформатика та еволюційні алгоритми;
- Теорія прийняття рішень.

Отриманні по розглянутій дисципліні знання будуть використовуватися та доповнюватися в курсах "Сучасний штучний інтелект" та "Інтелектуальні агенти", а також у курсовому проектуванні.

2 ЗМІСТ ЛЕКЦІЙ (10 ГОДИНИ)

2.1 Технології інтелектуального аналізу даних

Організація великих масивів даних. Задачі інтелектуального аналізу даних. Основні парадигми м'яких обчислень. Таксономія інтелектуального аналізу даних. Методи класифікації. Дерева вирішальних правил: основні поняття та застосування. Методи побудови дерев вирішальних правил. Асоціативні правила: основні поняття та застосування. Методи видобування асоціативних правил з великих масивів даних.

Лекцій – 6 год.

Лаб. роб. – 6 год.

Самостійна робота – 26 год.

Література [3, 12, 15]

2.2 Нейро-нечіткі мережі в задачах інтелектуального аналізу даних

Нечітка логіка та системи нечіткого виведення у прийнятті рішень та інтелектуальному аналізі даних. Штучні нейронні мережі в інтелектуальному аналізі даних та прийнятті рішень. Нечітке логічне виведення як інструмент інтелектуального аналізу даних. Моделі нейро-нечітких мереж. Методи синтезу та навчання нейро-нечітких мереж. Нечіткий кластер-аналіз. Застосування нейро-нечітких мереж. Порівняльна характеристика нейро-нечітких мереж.

Лекцій – 4 год.

Лаб. роб. – 4 год.

Самостійна робота – 26 год.

Література [1–19]

3 ПЕРЕЛІК ЛАБОРАТОРНИХ РОБІТ

Лабораторна робота № 1. Методи видобування асоціативних правил з великих масивів даних (3 год.).

Мета роботи: вивчити основні методи видобування асоціативних правил з великих масивів даних, навчитися використовувати спеціалізовані програмні засоби для видобування знань з масивів даних.

Лабораторна робота № 2. Деревя вирішальних правил (3 год.).

Мета роботи: вивчити основні методи побудови дерев вирішальних правил, навчитися використовувати спеціалізовані програмні засоби для побудови дерев рішальних правил.

Лабораторна робота № 3. Нейро-нечіткі системи (4 год.).

Мета роботи: вивчити основні моделі та методи синтезу нейро-нечітких систем, освоїти принципи побудови нейро-нечітких мереж за допомогою програмних засобів, навчитися використовувати нейро-нечіткі моделі для інтелектуального аналізу даних.

4 ПЕРЕЛІК ТЕМ ДО САМОСТІЙНОЇ РОБОТИ

1. Задачі інтелектуального аналізу даних (8 год.).
2. Методи побудови дерев вирішальних правил (10 год.).
3. Методи видобування асоціативних правил з даних (8 год.).
4. Моделі нейро-нечітких мереж (6 год.).
5. Методи синтезу та навчання нейро-нечітких мереж (8 год.).
6. Нечіткий кластер-аналіз (6 год.).
7. Порівняльна характеристика нейро-нечітких мереж (6 год.).

Контроль самостійної роботи передбачає написання рефератів або контрольних робіт.

5 НАВЧАЛЬНО-МЕТОДИЧНІ МАТЕРІАЛИ З ДИСЦИПЛІНИ

5.1 Література основна

1. Андрейчиков А.В., Андрейчикова О.Н. Интеллектуальные информационные системы: Учебник. – М.: Финансы и статистика, 2004. – 424 с.
2. Дубровин В.И., Субботин С.А., Богуслаев А.В., Яценко В.К. Интеллектуальные средства диагностики и прогнозирования надежности авиадвигателей: Монография. – Запорожье: ОАО "Мотор-Сич", 2003. – 279 с.
3. Дюк В., Самойленко А. Data mining: учебный курс.–СПб.: Питер, 2001.– 368 с.
4. Круглов В.В., Борисов В.В. Искусственные нейронные сети: теория и практика.–М.: Горячая линия-Телеком, 2001. – 382 с.
5. Рутковская Д., Пилиньский М., Рутковский Л. Нейронные сети, генетические алгоритмы и нечеткие системы: Пер. с польск. И.Д. Рудинского. – М.: Горячая линия-Телеком, 2004. – 452 с.

5.2 Література додаткова

6. Бондарев В.Н., Аде Ф.Г. Искусственный интеллект. – Севастополь: СевНТУ, 2002. – 615 с.
7. Вороновский Г.К., Махотило К.В., Петрашев С.Н., Сергеев С.А. Генетические алгоритмы, искусственные нейронные сети и проблемы виртуальной реальности.–Харьков: Основа, 1997. – 112 с.
8. Дубровін В.І., Субботін С.О. Методи оптимізації та їх застосування в задачах навчання нейронних мереж: Навчальний посібник. – Запоріжжя: ЗНТУ, 2003. – 136 с.
9. Дьяконов В. MATLAB 6: учебный курс. – СПб.: Питер, 2001. – 592 с.
10. Зайченко Ю.П. Основы проектирования интеллектуальных систем. Навчальний посібник. – К.: Слово, 2004. – 352 с.
11. Кричевский М.Л. Интеллектуальные методы в менеджменте. – СПб.: Питер, 2005. – 304 с.
12. Люгер Дж.Ф. Искусственный интеллект: стратегии и методы решения сложных проблем / Пер. с англ. – М.: Вильямс, 2005. – 864 с.
13. Митюшкин Ю.И., Мокин Б.И., Ротштейн А.П. Soft Computing: идентификация закономерностей нечеткими базами знаний. – Винница: УНИВЕРСУМ-Винница, 2002. – 145 с.
14. Осовский С. Нейронные сети для обработки информации. – М.: Финансы и статистика, 2004. – 344 с.
15. Рассел С., Норвиг П. Искусственный интеллект: современный подход, 2-е изд.: Пер с англ.. – М.: Вильямс. – 2006. – 1408 с.
16. Рідкокаша А.А., Голдер К.К. Основы систем штучного інтелекту. Навчальний посібник. – Черкаси: "ВІДЛУННЯ-ПЛЮС", 2002. – 240 с.

17. Ротштейн А.П. Интеллектуальные технологии идентификации: нечеткие множества, генетические алгоритмы, нейронные сети. – Винница: УНІВЕРСУМ-Вінниця, 1999. – 320 с.

18. Усков А.А., Кузьмин А.В. Интеллектуальные технологии управления. Искусственные нейронные сети и нечеткая логика. – М.: Горячая линия-Телеком, 2004. – 143 с.

19. Ярушкина Н.Г. Основы теории нечетких и гибридных систем. – М.: Финансы и статистика, 2004. – 320 с.

РОБОЧА НАВЧАЛЬНА ПРОГРАМА
дисципліни
“ТЕХНОЛОГІЇ РОЗРОБКИ МЕРЕЖНИХ ДОДАТКІВ”
для студентів магістратури напряму підготовки
6.050103 “Програмна інженерія”

спеціальності **8.080403 “Програмне забезпечення автоматизованих систем”**

Факультет інформатики та обчислювальної техніки
 Кафедра програмних засобів

Форма навчання	Курс	Семестр	Усього (год)	Лекції (год)	Лаб. роб. (год)	Практ. (год)	Самост. під керівн.(год)	Самост. (год)	КП, КР, РГЗ (сем.)	Залік (сем.)	Іспит (сем.)
Денна	V	10	72/2	10	10	-		52	-	-	10
Заочна											

Робоча програма складена на основі Навчального плану підготовки магістрів по спеціальності "Програмне забезпечення автоматизованих систем" напряму 6.050103 "Програмна інженерія" від 17.07.2007 р. за Міжнародним проектом "Європейсько-український ступінь магістра з програмного забезпечення" (JEP 26182 2006) програми "Темпус" Європейської Комісії.

Робоча програма складена
 кандидатом фізико-математичних наук, доцентом

Онищенко В.Ф.

1 МЕТА І ЗАДАЧІ ДИСЦИПЛІНИ, ЇЇ МІСЦЕ В НАВЧАЛЬНОМУ ПРОЦЕСІ

1.1 Мета викладання дисципліни

Основна мета курсу – сформувати фундаментальні теоретичні знання і практичні навички у області базових принципів створення клієнтських і серверних додатків з використанням інтерфейсу «сокетів» в середовищі Linux/Unix. В курсі розглядаються принципи створення одне і багато – поточкових TCP і UDP серверів з використанням різних методів вводу-виводу. У той же час також розглядаються створення додатків, з використанням комплексних принципів, наприклад, видалений виклик процедур і створення захищеного каналу SSL.

1.2 Основні задачі предмету:

В результаті вивчення дисципліни студенти повинні:

- на алгоритмічному рівні знати функціонування протоколів IP, TCP, UDP, ICMP;
- на мові програмування C уміти створювати клієнтські і серверні додатки з використанням протоколів IP, TCP, UDP і інтерфейсу «сокетів» в середовищі ОС Linux;
- на мові програмування C уміти створювати багатопотокові серверні додатки з використанням методик синхронного і асинхронного вводу-виводу в середовищі ОС Linux;
- на мові програмування C уміти створювати захищені клієнтські і серверні додатки з використанням інтерфейсу «сокетів» і бібліотеки SSL;
- уміти створювати клієнтські і серверні додатки з використанням технології програмного інтерфейсу RPC в ОС Linux;
- уміти створювати клієнтські і багатопотокові серверні додатки на об'єктно-орієнтованій мові Java;
- уміти використовувати сучасні інструменти для відладки клієнтських і серверних додатків в середовищі ОС Linux.

2 ЗМІСТ ДИСЦИПЛІНИ

№	Назва теми	Кільк.годин	
		Лекц.	Лб.
1	Введення в стек протоколів TCP/IP.	1	
2	Введення в елементарні сокети.	2	
3	Багатозадачність. Розподіл навантаження і механізми вводу-виводу.	3	4
4	Основні відомості про сокети UDP. Принципи створення клієнтських і серверних додатків.	2	2
5	Основні відомості про сокети TCP. Принципи створення клієнтських і серверних додатків.	2	2
6	Створення клієнтських і серверних додатків з використанням об'єктно-орієнтованих сокетів.	3	2
7	Створення клієнтських і серверних додатків з використанням складних мережевих методик.	3	6
8	Основи техніки відладки мережевих додатків.	2	
	Всього	18	16

Тема №1. Введення в стек протоколів TCP/IP.

Клієнт-серверна архітектура. Введення в мережеве програмування. Мережевий рівень IP, ICMP. Транспортний рівень TCP потік, UDP датаграми і надійний орієнтований на повідомлення SCTP протокол. Орієнтовані і не орієнтовані на попереднє встановлення каналу зв'язку протоколи. Порти. APIs: Бібліотека «сокети» Берклі і WinSock.

Тема №2. Введення в елементарні сокети.

Іменовані канали. Структура IPv4 адреси сокета. Спеціальні адреси. Комунікаційний домен. Типи сокетів. Порядок проходження байтів і функції управління ними. Створення і прив'язка сокетів. Параметри сокетів. Елементарні перетворення імен і адрес.

Тема №3. Багатозадачність. Розподіл навантаження і механізми вводу-виводу.

Потоки і процеси. Створення потоків і процесів. Взаємодія потоків і процесів. Семафори, що виключають, і гонки. Управління дочірніми завданнями і завдання «зомбі». Блокуючий і не блокуючий ввід-вивід. Операції функції ioctl. Мультиплексування вводу-виводу. Опит каналів вводу-виводу.

Асинхронний ввід-вивід. Використовування функцій poll, select. Модель вводу-виводу, керованого сигналом. Сигнал SIGIO в ОС Linux

Тема №4. Основні відомості про сокети UDP. Принципи створення клієнтських і серверних додатків.

Узагальнена структура клієнтських і серверних UDP додатків. Клієнти та сервери без попереднього встановлення каналу. Прив'язка до «сокета». Скріплення «сокета» з напрямленими і широкомовними адресами. Приєднаний UDP «сокет». Організація взаємодії клієнта і серверу. Проблема втрати датаграмм. Організація тайм-аутів, використання виклику alarm в ОС Linux. Організація процесу запису і читання клієнтом даних з «сокета». Проблема обрізання UDP датаграмм. Проблема надійності UDP датаграмм. Послідовні і паралельні UDP сервери.

Тема №5. Основні відомості про сокети TCP. Принципи створення клієнтських і серверних додатків.

Узагальнена структура клієнтських і серверних TCP додатків. Логічне встановлення каналу. Прив'язка адреси і порту до «сокета». Черги очікування запитів на з'єднання. Прийом запитів від клієнтів. Внепоточні дані. Проблеми фрагментації пакетів. Заборона маршрутизації пакету. Взаємодія з клієнтом. Управління потоком TCP даних за допомогою ковзаючого вікна. Проблеми запису TCP даних. Багатопотокові і однопотокові сервери. Обмеження числа клієнтських з'єднань. Попереднє галуження (дублювання) серверу (сервлети). Архітектура з одним і декількома TCP з'єднаннями. Закриття каналів читання і запису. Обробка помилок і аварійних ситуацій. Запуск програм через inetd в ОС Linux.

Тема №6. Створення клієнтських і серверних додатків з використанням об'єктно-орієнтованих сокетів.

Мережеве програмування в Java. Адреса IP і клас InetAddress. Методи класу InetAddress. Універсальна адреса ресурсів URL. Клас URL в бібліотеці класів Java. Конструктори класу URL. Методи класу URL. Клас Socket. Конструктори класу Socket. Методи класу Socket. Робота з поточковими сокетами. Ініціалізація серверу. Ініціалізація клієнта. Передача даних між клієнтом і сервером. Завершення роботи серверу і клієнта. Використовування датаграмних сокетів. Клас DatagramSocket. Конструктори і методи класу DatagramSocket. Клас DatagramPacket. Конструктори і методи класу DatagramPacket. Зв'язок додатків Java з розширеннями серверу Web. Взаємодія між додатком Java і розширенням серверу Web, таким як CGI або ISAPI. Клас URLConnection. Методи класу URLConnection.

Ввод і вивід в Java. Класифікація класів вводу-виводу. Багатозадачні програми в Java. Створення потокового класу. Додавання потоку до класу. Синхронізація методів. Обмеження об'єктний – орієнтованого програмування.

Тема №7. Створення клієнтських і серверних додатків з використанням складних мережевих методик.

Видалений виклик процедур (RPC). Поняття мережевих заглушок. Поняття формату даних, не залежного від архітектури комп'ютера. Поняття номера програми, номери версії програми і номери процедури. Формат XDR. Рівні інтерфейсу програмування RPC. Інтерфейс високого рівня. Бібліотечні функції RPC бібліотеки librpcsvc.a. Створення мережевих компонентів з використанням компілятора грсген в ОС Linux. Мова транслятора грсген. Призначення і формат X-файлів. Призначення *_svc.c і *_clnt.c файлів, створених утилітою грсген. Недоліки компілятора грсген. Низькорівневий інтерфейс програмування RPC. Функції XDR перетворення. Класи і методи RPC_svc і RPC_cls для створення об'єктів RPC клієнтів і серверів. Управління набором RPC програм і версій. Методи RPC аутентифікації. Широкомовний режим RPC. Зворотний виклик RPC (call back).

Створення безпечних мережевих додатків. Використовування бібліотеки OpenSSL. Широкомовна і групова передача повідомлень. Використовування неструктурованих «сокетів» SOCK_RAW. Використовування маршрутизуючих «сокетів» в ОС UNIX.

Тема №8. Основи техніки відладки мережевих додатків.

Використовування ping, tcpdump, netstat і traceroute в ОС Linux. Використовування бібліотеки потокових «сокетів» SVR4. Використовування засобів трасування системних викликів.

3 ТЕМИ ЛАБОРАТОРНИХ РОБІТ

1. Створення програм клієнта і одне потокового (одне процесного) серверу TCP з використанням блокуючих сокетів.
2. Створення програм клієнта і одне потокового (одне процесного) серверу UDP з використанням не блокуючих сокетів і широкомовної розсилки.
3. Створення багато потокового (багато процесного) серверу з використанням мультиплексування вводу – виводу.
4. Створення багато потокового (багато процесного) серверу з використанням вводу–виводу, керованого сигналом SIGIO.
5. Створення програм клієнта і серверу з використанням безпечного SSL каналу.
6. Створення програми шлюзу-серверу з використанням маршрутизуючих «сокетів».
7. Створення програм RPC клієнта і серверу з використанням низькорівневого інтерфейсу.
8. Створення TCP/UDP програм клієнта і серверу з використанням Java орієнтованих сокетів.

4 НАВЧАЛЬНО-МЕТОДИЧНІ МАТЕРІАЛИ З ДИСЦИПЛІНИ

4.1 Література основна

1. У. Стивенс.: UNIX: Разработка сетевых приложений. – СПб.: Питер, 2003. – 1088 с.
2. Снейдер Й.: Эффективное программирование TCP/IP. Библиотека программиста. – СПб.:Питер, 2002. – 320 с.
3. Уолтон, Шон.: Создание сетевых приложений в среде Linux.: Пер. с англ.–М.: Издательский дом “Вильямс”, 2001.–464с.
4. Теренс Чан.: Системное программирование на C++ для UNIX.: Пер. с англ.–К.: Издательская группа ВНУ., Киев, 1997. – 592с.

4.2 Література додаткова

5. Митчелл, Марк, Оулдем, Джеффри, Самьюэл, Алекс.: Программирование для Linux. Профессиональный подход.: Пер. с англ.–М.: Издательский дом “Вильямс”, 2003.–288с.
6. Т. Паркер, К. Сиян.:TCP/IP. Для профессионалов. 3-е изд. – СПб.: Питер, 2004.– 859 с.
7. Стаханов А.А.: Сетевое администрирование Linux.–СПб.:БХВ-Петербург, 2004.–480с.
8. Смит, Родерик, В. Сетевые средства Linux.: Пер. с англ. – М.: Издательский дом "Вильямс.

РОБОЧА НАВЧАЛЬНА ПРОГРАМА
дисципліни
“БЕЗПЕКА ІНФОРМАЦІЙНИХ СИСТЕМ”
для студентів магістратури напряму підготовки
6.050103 “Програмна інженерія”

спеціальності **8.080403 “Програмне забезпечення автоматизованих систем”**

Факультет інформатики та обчислювальної техніки
Кафедра програмних засобів

Форма навчання	Курс	Семестр	Усього (год)	Лекції (год)	Лабор. роб. (год)	Практ. (год)	Самост. під керівн. (год)	Самост. (год)	КП, КР, РГЗ (сем.)	Залік (сем.)	Іспит (сем.)
Денна	V	10	72	10	10	–	–	74	–	10	–
Заочна											

Робоча програма складена на основі Навчального плану підготовки магістрів по спеціальності "Програмне забезпечення автоматизованих систем" напряму 6.050103 "Програмна інженерія" від 17.07.2007 р. за Міжнародним проектом "Європейсько-український ступінь магістра з програмного забезпечення" (JEP 26182 2006) програми "Темпус" Європейської Комісії.

Робоча програма складена

Неласою Г.В.

1 МЕТА І ЗАДАЧІ ДИСЦИПЛІНИ, ЇЇ МІСЦЕ В НАВЧАЛЬНОМУ ПРОЦЕСІ

1.1 Мета викладання дисципліни

Основна мета дисципліни – формування у студентів базових знань, необхідних для розуміння широкого кола реальних проблем захисту інформації у комп'ютерних системах та мережах.

1.2 Задачі вивчення дисципліни

Внаслідок вивчення дисципліни студенти повинні:

з н а т и: загальні принципи організації захисту інформації в комп'ютерних системах та мережах на базі сучасних концепцій.

в м і т и: впроваджувати системний підхід к захисту інформації в комп'ютерних системах та мережах, використовувати стандартні криптографічні алгоритми та протоколи, використовувати сучасні засоби та методи криптографічного захисту інформації, виконувати розрахунки характеристик безпечного використання паролів та ключів шифрування.

1.3 Перелік дисциплін, засвоєння яких необхідне для вивчення даної дисципліни

Дисципліна пов'язана з попередніми дисциплінами: "Дискретна математика", "Основи програмування та алгоритмічні мови", "Архітектура комп'ютерів", "Комп'ютерні мережі".

2 ЗМІСТ ДИСЦИПЛІНИ

Дисципліна вивчається у 9 семестрі. Загальний обсяг: 72год., ауд. заняття 20 год., лекції 10 год., лаб. роб. 10 год., самостійна робота – годин.

По дисципліні передбачений іспит, теоретичний залік.

2.1 Найменування тем, їх зміст, об'єм в годинах лекційних занять

2.1.1 Вступ. Предмет та задачі курсу. Основні визначення та термінологія. Література, що рекомендується.

Лекції – 1 година.

2.1.2 Загрози безпеки та методи захисту. Основні шляхи витоку інформації та несанкціонованого доступу в каналах телекомунікацій. Політика захисту інформації. Свідомість працівників. Окреслення небезпеки і потенційно можливих атак. Паролі. Firewall.

Лекції – 1 години.

2.1.3 Криптографічний захист інформації. Основні принципи шифрування. Поточкові алгоритми. Криптографічні системи з закритим ключем. Односторонні функції. Геш-функції. Криптографічні системи з відкритим ключем. Стандарти шифрування. Алгоритми цифрового підпису на еліптичних кривих.

Лекцій – 6 годин.

Лабораторних занять – 10 годин.

2.1.4 Нормативна і правова бази захисту інформації в системах та мережах телекомунікації України. Законодавча, нормативно-методична, наукова, нормативно-правова бази України із забезпечення інформаційної безпеки в системах телекомунікацій; стандартизація методів забезпечення та сертифікація компонентів систем телекомунікацій за вимогами інформаційної безпеки.

Лекцій – 2 години.

2.2 Перелік лабораторних робіт

Лабораторна робота № 1 Базові шифри. Частотний криптоаналіз
Обсяг – 2 години.

Лабораторна робота № 2 Режими шифрування блокових шифрів
Обсяг – 2 години.

Лабораторна робота № 3 Криптографія з відкритим ключем.
Функція гешування
Обсяг – 2 години.

Лабораторна робота № 4 Криптографічні перетворення на еліптичних кривих
Обсяг – 2 години.

Лабораторна робота № 5 Стеганографічний захист
Обсяг – 2 години.

2.3 Перелік тем для самостійної роботи

1. Віртуальні приватні мережі.
2. Архітектура і конфігурація Firewall.
3. Захищені операційні системи.
4. Смарт-карти: архітектура, технології, галузі застосування.
5. Банківські технології.
6. Комп'ютерні віруси. Класифікація. Методи протидії.
7. Захищені бази даних.
8. Система безпеки Windows XP.
9. Безпека Internet – бізнесу.
10. Технічні засоби захисту інформації.

3 НАВЧАЛЬНО-МЕТОДИЧНІ МАТЕРІАЛИ ЗА ДИСЦИПЛІНОЮ

3.1 Основна література

1. Дж. Уолрэнд. Телекоммуникационные и компьютерные сети. Вводный курс. Москва: Постмаркет, 2001. – 480с.
2. Вербіцкий О.В. Вступ до криптології. – Видавництво наук.-техн. Літератури. Львів, 1998.
3. Теоретические основы компьютерной безопасности: Учеб. пособие для вузов/ П.Н. Деревянин, О.О.Михальский, Д.И. Правиков и др. – М.: Радио и связь, 2000. – 192 с.
4. Коржик В.И., Кушнир Д.В. Теоретические основы информационной безопасности телекоммуникационных систем: учеб. пособие (спец. 200900, 201000, 060800)/ СПбГУТ. – Спб, 2000. – 134с.
5. Зегжда Д.П., Ивашко А.М. Основы информационной безопасности. – М.: Горячая линия – Телеком, 2000. – 452с.
6. Щербаков А.Ю. Введение в теорию и практику компьютерной безопасности. – М.: издатель Молгачева С.В., 2001, 352с.
7. Романец Ю.В., Тимофеев П.А., Шаньгин В.Ф. Защита информации в компьютерных системах и сетях / Под ред. В.Ф. Шаньгина. – 2-е изд., перераб. и доп. – М.: Радио и связь, 2001. – 376 с.
8. Завгородний В.И. Комплексная защита информации в компьютерных системах: Учебное пособие. – М.: Логос, ПБОЮЛ Н.А. Егоров, 2001. – 264 с.
9. Устинов Г.Н. Основы информационной безопасности систем и сетей передачи данных. Учебное пособие. Серия «Безопасность». – М.: СИНТЕГ, 2000. – 248 с.
10. Иванов М.А. Криптографические методы защиты информации в компьютерных системах и сетях. М.Ж КУДИЦ-ОБРАЗ, 2001 – 386с.

3.2 Методичні матеріали з дисципліни

Методичні вказівки до лабораторних робіт з дисципліни “Захист інформації в інформаційних системах” для студентів магістратури напряму підготовки 6.050103 “Програмна інженерія” спеціальності 8.080403 “Програмне забезпечення автоматизованих систем” всіх форм навчання/Укл.: Неласа Г.В., Козіна Г.Л. – Запоріжжя: ЗНТУ, 2007. – 49с.

РОБОЧА НАВЧАЛЬНА ПРОГРАМА

дисципліни

“Розробка програмних систем”

для студентів магістратури напряму підготовки

6.050103 “Програмна інженерія”

спеціальності 8.080403 “Програмне забезпечення автоматизованих систем”

Факультет інформатики та обчислювальної техніки
Кафедра програмних засобів

Форма навчання	Курс	Семестр	Усього (год)	Лекції (год)	Лаб. роб. (год)	Практ. (год)	Самост. під керівн. (год)	Самост. (год)	КП, КР, РГЗ (сем.)	Залік (сем.)	Іспит (сем.)
Денна	V	10	108/3	20	20	–	–	68	–	–	10
Заочна											

Робоча програма складена на основі Навчального плану підготовки магістрів по спеціальності "Програмне забезпечення автоматизованих систем" напряму 6.050103 "Програмна інженерія" від 17.07.2007 р. за Міжнародним проектом "Європейсько-український ступінь магістра з програмного забезпечення" (JEP 26182 2006) програми "Темпус" Європейської Комісії.

Робоча програма складена
кандидатом технічних наук, доцентом

Табунщик Г.В.

1 МЕТА І ЗАДАЧІ ДИСЦИПЛІНИ, ЇЇ МІСЦЕ В НАВЧАЛЬНОМУ ПРОЦЕСІ

1.1 Мета викладання дисципліни

Метою даного курсу є закріплення теоретичних основ та практичних навичок об'єктно-орієнтованого проектування та розробки програмних продуктів. Дисципліна "Розробка програмних систем" спрямована на поглиблення знань та практичних навичок студентів з основ сучасної технології створення складних програмних продуктів на базі ідей і принципів об'єктно-орієнтованого методу. Такі знання призначені для використання при розробці архітектури складних корпоративних програмних додатків з урахуванням сучасних вимог у відношенні до надійності, якості та ефективності програмних продуктів, що створюються.

1.2 Завдання вивчення дисципліни:

На основі вивчення дисципліни "Розробка програмних систем" студент повинен знати: етапи життєвого циклу програмного забезпечення, елементи об'єктної моделі, процедури об'єктно-орієнтованого аналізу, процедури об'єктно-орієнтованого проектування, основи уніфікованої мови моделювання UML, основні концепції уніфікованого процесу розробки програмного забезпечення UP, архітектуру програмного забезпечення, CASE –засоби формування вимог до програмного забезпечення та проектування програмного забезпечення.

вміти: визначати *структуру програмного забезпечення* комп'ютерних інформаційних систем в умовах ескізного і технічного проектування на основі аналізу функціональної структури системи, використовуючи інформацію про математичне, технічне, інформаційне забезпечення; розробляти *корпоративні програмні додатки* з використанням об'єктно-орієнтованих мов програмування C++, Java або систему програмування C#.

1.3 Зв'язок з іншими дисциплінами

Курс базується на поняттях, що вивчаються в дисциплінах «Об'єктно-орієнтоване програмування» та «Технології програмування та створення програмних продуктів», «Проектування, моделювання та аналіз інформаційних систем».

2 ЗМІСТ ДИСЦИПЛІНИ

Навчальний процес в ЗНТУ в умовах кредитно-модульної системи організації навчання здійснюється у таких організаційних формах: навчальні заняття, виконання індивідуальних завдань, самостійна робота студентів, практична підготовка, контрольні заходи. Основні види занять, що входять до складу модулів: лекція, лабораторне, семінарське заняття, консультація та самостійна робота під керівництвом викладача.

Дисципліна "Розробка програмних систем" вивчається на 5 курсі у 10 семестрі і завершується іспитом. Навчальним планом передбачено:

Всього:

- лекцій – 20 годин;
- лабораторних робіт – 20 годин;
- самостійної роботи – 68 годин;

Загальна кількість аудиторних годин – 108 годин.

В умовах кредитно-модульної системи навчальна дисципліна “Розробка програмних систем” відповідає 3 заліковим кредитам.

Дисципліна складається з 4 блоків змістових модулів – основних її розділів, які разом зі змістовими модулями наведені у таблиці 2.1.

Таблиця 2.1 – Склад, обсяг і терміни виконання змістових модулів дисципліни “Розробка програмних систем”

Модулі (блоки змістових модулів)	Короткий зміст модулів	Розподіл навчального часу за елементами модуля (видами занять), години			Самостійна робота студента		Обсяг навантаження студента	
		Л	ЛР	ПР	СРВ	СРС	Години	Кредити
1	2	3	4	5	7	8	9	10
Огляд сучасних засобів розробки корпоративних додатків	Вступ							
	Особливості корпоративних програмних додатків	5	5			17	27	0,75
	Платформи та засоби							
Об’єктні моделі та реляційні бази даних	Архітектурні рішення							
	Організація доступу до даних	5	5			17	27	0,75
	Типові рішення							
Уява даних у WEB	Модель подання даних							
	Засоби відображення	5	5			17	27	0,75
	Типові рішення							
Керування розподіленням завданнями	Стратегії паралельних завдань							
	Інтерфейс розподіленої обробки	5	5			17	27	0,75
	Типові рішення							
Всього		20	20			68	108	3

Позначення у таблиці 2.1: Л – лекції, ЛР – лабораторні роботи, ПР – практична робота, СРС – самостійна робота студента

2.1 Змістові модулі

2.1.1 Огляд сучасних засобів розробки корпоративних додатків

Вступ. Мета і завдання курсу. Зміст курсу, його місце у учбовому плані спеціальності. Види та об'єми занять, форми контролю.

Розробка об'єктно-орієнтованих програм рівня підприємства. Особливості корпоративних програмних додатків. Розробка програмного забезпечення на базі компонентів.

Платформи та засоби: С++, Java2EE, .NET. Java2EE та архітектура системи підприємства.

Лекцій – 5 год.

Лабораторна робота – 5 год.

Самостійна робота – 17 год.

Література: [2, 7, 8].

2.1.2 Об'єктні моделі та реляційні бази даних

Архітектурні рішення. Відображення об'єктів та реляційних структур – відображення зв'язків,. Використання мета даних.

Організація доступу до даних. З'єднання з базою даних.

Типові рішення: подання логіки, типові рішення джерел даних, моделювання поведінки, моделювання структури, використання мета даних.

Лекцій – 5 год.

Лабораторна робота – 5 год.

Самостійна робота – 17 год.

Література: [1, 7, 8].

2.1.3 Уява даних у WEB

Розподілені комунікації. Мережна взаємодія. Web взаємодія.

Web взаємодія у системі підприємства: Java-сервлети, JavaServerPages

Типові рішення:

Лекцій – 5 год.

Лабораторна робота – 5 год.

Самостійна робота – 17 год.

Література: [1–3, 6].

2.1.4 Керування розподіленими завданнями

Паралельне виконання завдань. Стратегії розподілених розрахунків. Транзакції. Паралельні операції та сервери програмних додатків.

Лекцій – 5 год.

Лабораторна робота – 5 год.

Самостійна робота – 17 год.

Література: [6–9].

2.2 Перелік лабораторних робіт

Мета проведення лабораторних занять посідає в придбанні досвіду розробки великих корпоративних програмних додатків

Лабораторна робота № 1. Знайомство з платформами розробки корпоративних програмних додатки

Мета роботи: Вивчити основні можливості Java та Java2EE

Обсяг – 5 годин

Лабораторна робота № 2. Розробка об'єктних баз даних

Мета роботи: Навчитись реалізувати об'єктні бази даних та організувати доступ до даних

Обсяг – 5 годин

Лабораторна робота № 3. Подання даних у Web

Мета роботи: Навчитись використовувати JSP при розробці Web-додатків

Обсяг – 5 годин

Лабораторна робота № 4. Реалізація розподіленої обробки даних

Мета роботи: навчитись реалізувати розподілену обробку даних

Обсяг – 5 годин

2.3 Перелік тем для самостійної роботи

Нижче наведено перелік тем для самостійної роботи студентів:

- | | |
|--------------------------------------|---------|
| 1. Основні конструкції мови Java | 17 год. |
| 2. Засоби JDBC | 17 год. |
| 3. Мережева взаємодія | 17 год. |
| 4. Інтеграція корпоративних додатків | 17 год. |

3 НАВЧАЛЬНО-МЕТОДИЧНІ МАТЕРІАЛИ З ДИСЦИПЛІНИ

3.1 Література основна

1. Мацяшек Л.А. Анализ требований и проектирование систем. Разработка информационных систем с использованием UML. Пер. с англ. – М.:Изд. дом «Вильямс», 2002. –432 с.

2. Перроун П. Дж., Чаганти В.С.Р. Создание корпоративных ситем на основе Java 2 Enterprise Edition. –М.:Вильямс, 2001. –1184 с.

3. Фаулер М. архитектура корпоративных программных приложений. – М.:Изд. дом «Вильямс», 2004. –544с.

3.2 Література додаткова

4. Буч Г. Объектно-ориентированный анализ и проектирование с примерами приложений на С++. – М.:Бином, 1999. –560 с.

5. Крачтен Я. Введение в Rational Unified Process. Пер. с англ. – М.:Издательский дом «Вильямс», 2002. – 240 с.: ил

6. Шеферд Д. Программирование на Microsoft Visual C++.NET. Мастер класс.– М.:Русская редакция, СПб.:Питер, 2005. –928с.

7. Хорстман К., Корнелл Г. Java 2. Библиотека профессионала, том 1. Основы, 7-е изд. М.: Вильямс, 2006. –896 с.

8. Хорстман К., Корнелл Г. Java 2. Библиотека профессионала, том 2. Тонкости программирования, 7-е изд. М.: Вильямс, 2006. –1168 с.

3.3 Методичні матеріали з дисципліни

9. Методичні вказівки до виконання лабораторних робіт з дисципліни "Розробка програмних систем" для студентів для студентів магістратури напряму підготовки 6.050103 "Програмна інженерія" спеціальності 8.080403 "Програмне забезпечення автоматизованих систем" / Укл. Табунщик Г.В.– Запоріжжя: ЗНТУ, 2007.– 20 с.

3.4 Програмні засоби

Пакети Visual Studio 2005, Java 2EE