

Рівненський державний гуманітарний університет
Кафедра інформатики та прикладної математики

РОБОЧА ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ
ВК9 «Системне програмування»

Спеціальність **122 Комп'ютерні науки**

освітня програма **Комп'ютерні науки**

факультет **Математики та інформатики**

2019 – 2020 навчальний рік

Опис навчальної дисципліни

Найменування показників	Галузь знань, напрям підготовки, освітньо-кваліфікаційний рівень	Характеристика навчальної дисципліни	
		денна форма навчання	заочна форма навчання
Кількість кредитів – 4,5	Галузь знань <u>12 Інформаційні технології</u>	Вибіркові компоненти ОП. Цикл професійної підготовки.	
	Спеціальність: <u>Комп'ютерні науки</u>		
Модулів – 1	Спеціальність (професійне спрямування): <u>«Комп'ютерні науки»</u>	Рік підготовки:	
Змістовних модулів – 1		3-й	
Індивідуальне завдання		Семестр:	
Загальна кількість годин – 135		5-й	
Тижневих годин для денної форми навчання: аудиторних – 3 самостійної роботи студента – 4,5 для заочної форми навчання: аудиторних – самостійної роботи студента –	Освітньо-кваліфікаційний рівень: бакалавр	Лекцій:	
		26 год.	
		Практичні	
		Лабораторні	
		28 год.	
		Самостійна робота	
		81 год.	
Вид контролю: іспит			
Передумови для вивчення дисципліни: «Програмування», «Комп'ютерна схемотехніка та архітектура ЕОМ», «Об'єктно-орієнтоване програмування», «Теорія програмування», «Комп'ютерні мережі», «Веб-програмування», «Операційні системи», «Алгоритми і структури даних», «Програмування на базі технологій .NET».			

2. Мета та завдання навчальної дисципліни

Метою курсу є вивчення: системних програм; апаратних і програмних засобів сучасних процесорів, призначених для підтримки багатозадачних операційних систем; методології розробки системно-орієнтованих програм з використанням сучасних алгоритмічних мов і систем програмування.

Завдання курсу полягають у формуванні базових знань в області: організації і функціонування сучасних операційних систем; способів розробки системного програмного забезпечення з урахуванням апаратно-програмних особливостей обчислювальної машини; сучасних систем програмування і розробки системного програмного забезпечення.

Для вивчення курсу «Системне програмування» потрібні знання принципів роботи ЕОМ, навички алгоритмізації і розробки програм на алгоритмічних мовах високого рівня і машинно-орієнтованих мовах, роботи з операційними системами і системами програмування на рівні, достатньому для налаштування і виконання програм.

Курс базується на знаннях, отриманих при вивченні курсів «Програмування», «Комп'ютерна схемотехніка та архітектура ЕОМ», «Об'єктно-орієнтоване програмування», «Теорія програмування», «Комп'ютерні мережі», «Веб-програмування», «Операційні системи», «Алгоритми і структури даних», «Програмування на базі технологій .NET».

Матеріали курсу використовуються далі в курсах «Розподілені системи та паралельні обчислення», «Крос-платформне програмування», «Захист інформації», «Адміністрування комп'ютерних систем», «Розробка інтерфейсу користувача на основі технології WPF», «Програмування мобільних пристроїв», «Операційні системи».

Програма побудована за модульним принципом, відповідає вимогам кредитно-трансферної системи навчання.

Після вивчення цієї дисципліни студент оволодіває

загальними компетентностями

ЗК2. Здатність застосовувати знання у практичних ситуаціях.

ЗК7. Здатність вчитися і оволодівати сучасними знаннями.

ЗК8. Здатність до пошуку, оброблення та аналізу інформації з різних джерел.

ЗК12. Вміння виявляти, ставити та вирішувати проблеми.

ЗК13. Здатність приймати обґрунтовані рішення.

ЗК16. Здатність спілкуватися з представниками інших професійних груп різного рівня (з експертами з інших галузей знань/видів економічної діяльності).

ЗК17. Здатність розробляти та управляти проектами.

ЗК19. Здатність оцінювати та забезпечувати якість виконуваних робіт.

фаховими компетентностями:

ФК1. Здатність до проектування інформаційних систем, включаючи формальний опис їх структури та проведення моделювання бізнес-процесів

ФК6. Здатність використовувати сучасні комп'ютерні технології для системного, функціонального, конструкторського та технологічного проектування складних об'єктів і систем.

ФК8. Здатність розв'язувати проблеми масштабованості, підтримки віддалених компонентів і взаємодії різних програмних платформ в розподілених корпоративних інформаційних системах рівня підприємства.

ФК9. Здатність виявляти в даних раніше невідомі знання, які потрібні для прийняття рішень в різних сферах професійної діяльності та зберігати їх у сховищах даних.

ФК13. Здатність забезпечувати знання стандартів, методів і засобів управління процесами життєвого циклу інформаційних систем, продуктів і сервісів інформаційних технологій.

ФК14. Здатність публічно представляти власні і відомі наукові результати виробничо-технологічної діяльності.

ФК15. Здатність використовувати методи математичного та алгоритмічного моделювання при рішенні теоретичних і прикладних завдань.

ФК16. Здатність передавати результат проведених фізико-математичних і прикладних досліджень у вигляді конкретних рекомендацій, сформульованих у термінах предметної області явища, яке вивчалось.

3. Очікувані результати навчання

Згідно ОП «Комп'ютерні науки», дисципліна забезпечує набуття студентами **програмних результатів навчання:**

Знати:

ПРН4. Теоретичні і практичні основи методології та технології моделювання у процесі дослідження, проектування та експлуатації інформаційних систем, продуктів, сервісів інформаційних технологій, інших об'єктів професійної діяльності. Зокрема: типові алгоритми організації взаємодії між процесами; підсистеми ОС; апаратні і програмні засоби сучасних процесорів, призначених для підтримки багатозадачних ОС; завдання, які вирішуються при управлінні віртуальною пам'яттю; архітектуру і підсистеми ОС Windows; засоби синхронізації потоків в ОС сімейства Windows; структурну обробку виняткових ситуацій в ОС сімейства Windows; технологію COM; архітектуру і підсистеми платформи .NET; модульне програмування в середовищі .NET на основі збірок; механізм роботи прибиральника сміття; об'єктну модель платформи .NET; програмування з використанням делегатів і подій; багатопоточне програмування на платформі .NET; технологію графічного виведення Windows Presentation Foundation; технологію віддаленого виклику об'єктів .NET Remoting; технологію сервісної взаємодії програм Windows Communication Foundation;

Вміти:

ПРН 13. Уміння розробляти операційні моделі та здійснювати операційні дослідження в процесі аналізу та синтезу інформаційних систем різного призначення, володіння сучасними технологіями автоматизації проектування складних об'єктів і систем, продуктів і сервісів інформаційних технологій, сучасними парадигмами і мовами програмування. Зокрема: розробляти програми в ОС Windows; розробляти багатопоточні програми з синхронізацією даних; розробляти динамічно під'єднані бібліотеки; використовувати файли, які проектуються в пам'ять; перехоплювати виклики до операційної системи; розробляти програми і збірки для платформи .NET; розробляти багатопоточні програми з синхронізацією даних для платформи .NET; розробляти мережеві сервіси і клієнтські застосування для платформи .NET.

Автономія та відповідальність:

ПРН23. Прийняття рішень у складних і непередбачуваних умовах, що потребує застосування нових підходів та прогнозування.

ПРН25. Здатність до подальшого навчання, яке значною мірою є автономним та самостійним.

Програма розрахована на обсяг **135** навчальних годин, з них – **54** аудиторних. Примірний розподіл аудиторних годин за видами занять: лекцій – **26** годин, лабораторних робіт – **28** години.

4. Програма навчальної дисципліни

Тема 1. Модель виконання коду в середовищі CLR.

Компіляція початкового коду у керовані модулі. Об'єднання керованих модулів у збірку. Завантаження CLR. Виконання коду збірки: IL-код і верифікація; Небезпечний код; IL і захист інтелектуальної власності. NGen.exe. Бібліотека FCL. CTS. CLS. Взаємодія з некерованим кодом.

Тема 2. Вступ у паралельні обчислення.

Мотиви паралелізму. Класифікація обчислювальних систем. Архітектура однопроцесорної машини. Мультикомп'ютери з розподіленою пам'яттю. Мультипроцесор з

розділюваною пам'яттю. Режими виконання незалежних частин програми. Рівні паралелізму у багатоядерній архітектурі. Аналіз ефективності паралельних обчислень. Досягнення максимального паралелізму.

Тема 3. Робота з потоками.

Структура потоку. Стани потоків. Перемикання контексту. Робота з потоками в C#. Передача параметрів. Призупинення потоку. Властивості потоку. Пріоритети потоків. Локальне сховище потоку. Пул потоків.

Тема 4. Делегати, події і потоки виконання

Делегати. Патерн «спостерігач». Операції. Передача делегатів в методи. Обробка виключень при виклику делегатів. Події. Багатопоточні застосування. Клас Thread. Асинхронні делегати. Рекомендації з програмування.

Тема 5. Робота з файлами в C#

Основні поняття. Потоки байтів. Асинхронний ввід-вивід. Потоки символів. Двійкові потоки. Консольний ввід-вивід. Робота з каталогами і файлами

Тема 6. Робота з двійковими файлами в C#

Потоки для роботи з двійковими файлами: класи – Stream, FileStream, MemoryStream, BufferedStream. Класи для читання і запису простих типів у двійковому форматі.

Серіалізація даних: класи – BinaryFormatter, SoapFormatter.

Тема 7. Синхронізація процесів і потоків

Проблеми синхронізації. *Взаємовиключний доступ*: клас Interlocked; критична секція; клас Monitor; ключове слово lock. *Механізми синхронізації ОС: Семафори*, клас Semaphore: особливості використання семафорів; реалізація задачі виробників-споживачів за допомогою семафорів; проблеми використання семафорів; реалізація семафорів в ОС Windows. *Мьютекси*, клас Mutex: правила спрощеного паралелізму; реалізація задачі виробників-споживачів за допомогою мьютексів; реалізація мьютексів в ОС Windows. *Події*. Клас ManualResetEvent. Клас AutoResetEvent. Приклади використання

Тема 8. Асинхронне програмування.

Важливість асинхронного програмування. Асинхронні шаблони. Ключові слова async та await. Створення і використання асинхронних методів. Обробка помилок з допомогою асинхронних методів.

5. Структура навчальної програми

№з/п	Назви змістових модулів і тем	Число годин									
		Денна форма					Заочна форма				
		всього	у тому числі				всього	у тому числі			
л	п		лаб.	с.р.	л	п		лаб.	с.р.		
1	2	3	4	5	6	7	8	9	10	11	12
1.	Модель виконання коду в середовищі CLR.	11	3		2	6					
2.	Вступ у паралельні обчислення.	17	3		4	10					
3.	Робота з потоками.	20	4		6	10					
4.	Делегати, події і потоки виконання	18	4		4	10					
5.	Робота з файлами в C#	14	2		2	10					
6.	Робота з двійковими файлами в C#	14	2		2	10					
7.	Синхронізація процесів і потоків	19	4		4	11					
8.	Асинхронне програмування.	22	4		4	14					
	Контрольна робота										
	Разом:	135	26		28	81					

Усього годин:	135	26	28	81					
----------------------	------------	-----------	-----------	-----------	--	--	--	--	--

6. Теми семінарських занять

Згідно програми не передбачено.

7. Теми практичних занять

Згідно програми не передбачено.

8. Теми лабораторних занять

№ з/п	Назва теми	Число годин
1.	Створення консольного проекту.	2
2.	Управління процесами в ОС Windows.	2
3.	Знайомство з багатопоточною обробкою.	2
4.	Розробка багатопотокових застосувань.	4
5.	Управління пріоритетами потоків.	2
6.	Делегати і події в мові C#.	4
7.	Робота з файлами в C#.	2
8.	Робота з двійковими файлами в C#.	2
9.	Синхронізація процесів і потоків в C#.	4
10.	Асинхронне програмування.	4
Всього:		28

9. Самостійна робота

№ з/п	Назва теми	Число годин
1.	Модель виконання коду в середовищі CLR.	6
2.	Вступ у паралельні обчислення.	10
3.	Робота з потоками.	10
4.	Делегати, події і потоки виконання	10
5.	Робота з файлами в C#	10
6.	Робота з двійковими файлами в C#	10
7.	Синхронізація процесів і потоків	11
8.	Асинхронне програмування.	14
Всього:		81

Самостійна робота є основним засобом засвоєння студентом навчального матеріалу у час, вільний від аудиторних навчальних занять.

Самостійна робота студента над засвоєнням навчального матеріалу з навчальної дисципліни може виконуватися у бібліотеці, навчальних кабінетах, лабораторіях і комп'ютерних класах, а також у домашніх умовах та передбачає:

- вивчення лекційного матеріалу по темі;
- опрацювання літератури по темі;
- підготовку до лабораторних робіт;
- підготовку до самостійних робіт;
- роботу за персональним комп'ютером по темі;
- виконання індивідуальних завдань.

10. Індивідуальні завдання

Індивідуальне завдання – форма організації навчального процесу, яка має на меті поглибити, узагальнити та закріпити знання, які студенти отримують у процесі навчання, а також застосувати ці знання під час вирішення практичних задач. Як індивідуальне завдання

кожен студент має виконати довгострокову роботу за власним варіантом. Потрібно створити системну програму Завдання полягає у створенні програмного виробу.

Результати виконання індивідуального завдання повинні бути представлені в електронному та друкованому виглядах.

11. Засоби діагностики результатів навчання

- стандартизовані тести;
- наскрізні проекти;
- командні проекти;
- звіти, реферати;
- презентації результатів виконаних завдань та досліджень;
- захист індивідуальних завдань на лабораторних заняттях;
- захист індивідуальних завдань;
- екзамен.

12. Шкала оцінювання: національна та ЄКТС

Сума балів за всі види навчальної діяльності	Оцінка за національною шкалою	
	для екзамену, курсового проекту (роботи), практики	для заліку
90-100	відмінно	зараховано
82-89	добре	
74-81		
64-73		
60-63	задовільно	
33-59	незадовільно з можливістю повторного складання	не зараховано з можливістю повторного складання
1-34	незадовільно з обов'язковим повторним вивченням дисципліни	не зараховано з обов'язковим повторним вивченням дисципліни

13. Розподіл балів, які отримують студенти

Поточне тестування та самостійна робота											Екзамен	Всього
Змістовий модуль												
T1	T2	T3	T4	T5	T6	T7	T8	КР	ІНЗ			
6	6	6	6	6	6	6	8	10	10	30	100	

T1, T2 ... – теми змістових модулів

КР – контрольна робота

ІНЗ – індивідуальне завдання

14. Методичне забезпечення

Теоретичні відомості, контрольні запитання, варіативні вправи та вправи за зразком, індивідуальні завдання для самостійної роботи студентів, практичні завдання для перевірки знань студентів розміщені на сайті факультету за адресою: <https://drive.google.com/drive/folders/1yApCroaoeXakB5mzWnaVuKvu5F5baeiT>

15. Рекомендована література

Основна література

1. Павловская Т. А. С#. Программирование на языке высокого уровня: Учебник для вузов. \ Т.А. Павловская — СПб.: Питер, 2014. — 432 с.

2. Дейтел, Х. Операционные системы. Часть 1. Основы и принципы/ Х. Дейтел, П. Дейтел, Д. Чофисес. – М. : Бином, 2013. – 1024 с.
3. Дейтел, Х. Операционные системы. Часть 2. Распределенные системы, сети, безопасность/ Х. Дейтел, П. Дейтел, Д. Чофисес. – М. : Бином, 2013. – 704 с.
4. Дейтел П., Дейтел Х. Как программировать на Visual C# 2012. 5-е изд. \ П. Дейтел, Х. Дейтел — СПб.: Питер, 2014. — 864 с.
5. Шилдт, Герберт. C# 4.0: полное руководство. \ , Герберт Шилдт : Пер. с англ. — М.: ООО "И.Д. Вильямс", 2011. — 1056 с.: ил.
6. Троелсен, Эндрю. Язык программирования C# 2010 и платформа .NET 4.0, 5-е изд. \ Эндрю Троелсен : Пер. с англ. — М. : ООО "И.Д. Вильямс", 2011. — 1392 с. : ил.
7. Нейгел, Кристиан, Ивьен, Билл, Глинн, Джей, Уотсон, Карли. C# 4.0 и платформа .NET 4 для профессионалов. : Пер. с англ. — М. : ООО "И.Д. Вильямс", 2011. — 1440 с.
8. Рихтер Дж. CLR via C#. Программирование на платформе Microsoft .NET Framework 4.5 на языке C#. 4-е изд. \ Дж. Рихтер — СПб.: Питер, 2013. — 896 с.: ил. — (Серия «Мастер-класс»).
9. Гордеев, А. В. Системное программное обеспечение / А. В. Гордеев, А. Ю. Молчанов. – СПб. : Питер, 2002. – 736 с.
10. Макконнелл, С. Совершенный код. Мастер-класс / С. Макконнелл. – СПб. : Питер, 2012. – 896 с.
11. Мартин Р. Чистый код: создание, анализ и рефакторинг. Библиотека программиста. — СПб.: Питер, 2010. — 464 с.
12. Идеальный код / Под редакцией Э. Орама и Г. Уилсона — СПб.: Питер, 2011. — 624 с.
13. Руссинович, М. Внутреннее устройство Microsoft Windows: Windows Server 2003, Windows XP и Windows 2000 / М. Руссинович, Д. Соломон ; пер. с англ. – 4-е изд. – М. : Издат.-торг. дом «Русская Редакция» ; СПб. : Питер, 2005. – 992 с.
14. Харт, Д. Системное программирование в среде Windows / Д. Харт. – М. : Издат. дом «Вильямс», 2005. – 592 с.

16. Інформаційні (Інтернет) ресурси

1. Електронний ресурс. Режим доступу: <https://metanit.com/sharp/>
2. Бібліотека технічної документації для кінцевих користувачів, розробників та IT-спеціалістів, які працюють з продуктами Microsoft. Електронний ресурс. Режим доступу: <https://docs.microsoft.com/ru-ru/>