

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
КАРПАТСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
ІМЕНІ ВАСИЛЯ СТЕФАНІКА**



Фізико-технічний факультет

Кафедра комп'ютерної інженерії та електроніки

СИЛАБУС НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

Системне програмування

Рівень вищої освіти: перший (бакалаврський)

Освітня програма: Комп'ютерна інженерія

Спеціальність: F7 Комп'ютерна інженерія

Галузь знань: F Інформаційні технології

Затверджено на засіданні кафедри
Протокол № 13 від 26 серпня 2025 р.

м. Івано-Франківськ – 2025 р.

ЗМІСТ

1. Загальна інформація	3
2. Опис дисципліни	3
3. Структура курсу	5
4. Система оцінювання курсу	7
5. Оцінювання відповідно до графіку навчального процесу	7
6. Ресурсне забезпечення	8
7. Контактна інформація	10
8. Політика навчальної дисципліни	10

1. Загальна інформація

Назва дисципліни	Системне програмування
Освітня програма	Комп'ютерна інженерія
Спеціалізація (за наявності)	
Спеціальність	F7 Комп'ютерна інженерія
Галузь знань	F Інформаційні технології
Освітній рівень	бакалавр
Статус дисципліни	основна
Курс / семестр	3/5,6
Розподіл за видами занять та годинами навчання (якщо передбачені інші види, додати)	6 кредитів Лекції – 36 год. Лабораторні заняття – 42 год. Самостійна робота – 102 год.
Мова викладання	українська
Посилання на сайт дистанційного навчання	https://d-learn.pnu.edu.ua/

2. Опис дисципліни

Мета та цілі курсу

Метою вивчення навчальної дисципліни “Системне програмування” є отримання теоретичних знань і практичних навичок з програмування мовою асемблера (Nasm, ОС Linux) мікропроцесорів Intel x86-32/64 та їх апаратних розширень (CPU, FPU, MMX, XMM, SSE, AVX), отримання теоретичних та практичних знань з побудови основних структурних блоків трансляторів (лексичного аналізатора, синтаксичного аналізатора, семантичного аналізатора, генератора машинних кодів).

Основним **завданням** навчальної дисципліни є:

- ознайомлення з сучасним станом, тенденціями та перспективами розвитку мікропроцесорів Intel x86-32/64 та їх програмування на мові асемблера;

- вивчення системи команд мікропроцесорів Intel x86-32/64 та їх апаратних розширень (CPU, FPU, MMX, SSE, AVX);

- вивчення принципів організації і побудови одно- та багатомодульних програм на мові асемблера NASM в середовищі ОС Linux;

- вивчення загальних принципів організації і побудови основних структурних блоків трансляторів.

- вивчення принципів та методів лексичного, синтаксичного і семантичного аналізу, генерації машинного коду.

У результаті вивчення навчальної дисципліни студент повинен

знати:

- основні принципи розроблення системних програм;

- програмну архітектуру процесорів Intel x86-32/64 та їх апаратних розширень (CPU, FPU, MMX, SSE, AVX);

- систему команд цілочислового процесора CPU та процесора з

плаваючою крапкою FPU;

- системи команд MMX, SSE, AVX;
- визначення і властивості скінченних автоматів;
- методи проектування лексичних аналізаторів;
- принципи низхідного і висхідного синтаксичного аналізу;
- способи подання семантики;
- методи генерації машинного коду.

вміти:

- створювати системні програми на мові асемблері NASM в ОС Linux для мікропроцесорів Intel x86-32/64 та їх апаратних розширень (CPU, FPU, MMX, SSE, AVX);
- розрізняти лексичні, синтаксичні і семантичні помилки в програмах;
- проектувати лексичні аналізатори на основі скінченних автоматів;
- будувати формальні граматики;
- побудувати дерево синтаксичного розбору;
- проектувати низхідний LL(1) висхідний LR синтаксичні аналізатори спуску;
- генерувати машинні коди.

Компетентності

Інтегральна компетентність.

Здатність розв'язувати складні спеціалізовані задачі та практичні проблеми під час професійної діяльності в комп'ютерній галузі або навчання, що передбачає застосування теорій та методів комп'ютерної інженерії і характеризується комплексністю та невизначеністю умов.

Загальні компетентності

ЗК2. Здатність вчитися і оволодівати сучасними знаннями.

ЗК3. Здатність застосовувати знання у практичних ситуаціях.

ЗК12. Здатність до розуміння предметної галузі та професійної діяльності.

ЗК13. Здатність розв'язувати поставлені задачі та приймати відповідні рішення.

Спеціальні компетентності.

ФК2. Здатність використовувати сучасні методи і мови програмування для розроблення алгоритмічного та програмного забезпечення.

ФК3. Здатність створювати системне та прикладне програмне забезпечення комп'ютерних систем та мереж.

ФК 9. Здатність системно адмініструвати, використовувати, адаптувати та експлуатувати наявні інформаційні технології та системи.

ФК13. Здатність вирішувати проблеми у галузі комп'ютерних та інформаційних технологій, визначати обмеження цих технологій.

Програмні результати навчання

ПРН1. Знати і розуміти наукові положення, що лежать в основі функціонування комп'ютерних засобів, систем та мереж.

ПРН3. Знати новітні технології в галузі комп'ютерної інженерії.

ПРН4. Знати та розуміти вплив технічних рішень в суспільному, економічному, соціальному і екологічному контексті.

ПРН10. Вміти розробляти програмне забезпечення для вбудованих і розподілених застосувань, мобільних і гібридних систем, розраховувати, експлуатувати, типове для спеціальності обладнання.

ПРН15. Вміти виконувати експериментальні дослідження за професійною тематикою.

ПРН23. Вміти розробляти та супроводжувати алгоритмічне та програмне забезпечення для спеціалізованих обчислювальних пристроїв та їх компонентів використовуючи технології автоматизованого проектування комп'ютерних систем.

3. Структура курсу

№	Тема	Результати навчання	Кількість годин	Завдання
5 семестр				
1	Вступ. Системне програмування у контексті МК Intel x86-64.	Знати особливості системного програмування, призначення асемблера і машинного коду, машинне подання чисел в МК Intel x86-64, основні етапів розвитку і характеристики МК Intel.	2	Лабораторна робота (звіт); контрольні запитання.
2	Режими роботи та програмна модель МК Intel x86-64.	Знати режими роботи та програмну модель МК Intel x86-64, регістри загального та спеціального призначення, сегментні регістри.	2	Лабораторна робота (звіт); контрольні запитання.
3	Асемблер NASM.	Знати засоби компіляції, налагодження і виконання асемблерних програм, синтаксис асемблера Nasm, типи даних, директивами, способами адресації пам'яті.	2	Лабораторна робота (звіт); контрольні запитання.
4	Макропроцесор і макродирективи.	Розуміти можливості макропроцесора і макродиректив і вміти їх застосовувати для спрощення програмування на асемблері.	2	Лабораторна робота (звіт); контрольні запитання.
5	Команди CPU.	Знати функціональну класифікації команд CPU, команди роботи з цілими числами, команд пересилання даних, арифметичні команди, логічні команди і операції, ланцюгові команди.	2	Лабораторна робота (звіт); контрольні запитання.
6	Команди переходів.	Знати команди умовних та	2	Лабораторна

	Стек. Системні виклики.	безумовних переходів, організації циклів. Розуміти роботу стеку і стекового фрейму при виклику підпрограм. Вміти застосовувати системні виклики.		робота (звіт); контрольні запитання.
7	Співпроцесор FPU. MMX розширення.	Розуміти принцип роботи FPU. Знати архітектуру, програмну модель і команди співпроцесора FPU. Знати команди MMX розширення. Вміти обробляти цілочислові дані за допомогою команд MMX розширення.	2	Лабораторна робота (звіт); контрольні запитання.
8	SSE розширення.	Розуміти можливості команд SSE, SSE2, SSE3, SSSE3, SSE4 розширень. Вміти застосовувати SSE команди для потокової обробки даних.	2	Лабораторна робота (звіт); контрольні запитання.
9.	AVX розширення.	Знати команди x86-AVX, AVX2, AVX-512 розширень для потокової обробки даних та виконання арифметичних операцій над запакованими цілими і дійсними числами.	2	Лабораторна робота (звіт); контрольні запитання; тести.
6 семестр				
10.	Структура транслятора, елементи формальних граматик.	Знати структуру транслятора, елементи формальних граматик, операції над мовами, регулярні вирази і визначення.	2	Лабораторна робота (звіт); контрольні запитання.
11.	Лексичний аналізатор.	Знати структуру лексичного аналізатора, вміти будувати лексичний аналізатор на основі діаграм переходів	2	Лабораторна робота (звіт); контрольні запитання.
12.	Недермінові і детерміновані скінченні автомати.	Знати поняття токени, шаблони, лексеми. Вміти перетворювати недетермінований скінченний автомат в детермінований.	2	Лабораторна робота (звіт); контрольні запитання.
13.	Контекстно вільні граматики.	Знати визначення формальної граматики, типи граматик. Вміти виявляти неоднозначність граматик, усувувати ліву рекурсію та виконувати ліву факторизацію.	2	Лабораторна робота (звіт); контрольні запитання.
14.	Низхідний рекурсивний синтаксичний аналізатор.	Знати класифікацію синтаксичних аналізаторів, дерева розбору, абстрактне синтаксичне дерево. Вміти будувати низхідний рекурсивний синтаксичний аналізатор.	2	Лабораторна робота (звіт); контрольні запитання.
15.	Передбачаючий синтаксичний аналізатор LL(1).	Знати як працює передбачаючий синтаксичний аналізатор. Вміти будувати множини First, Follow, таблиці синтаксичного аналізу.	2	Лабораторна робота (звіт); контрольні запитання.

16.	Висхідний LR-аналізатор. Побудова таблиць SLR аналізу.	Знати базові концепції LR-аналізу і роботу висхідного LR-аналізатор. Вміти виконувати синтаксичний аналіз "перенесення-згортка" з використанням буфера і стеку, будувати таблиці SLR аналізу.	2	Лабораторна робота (звіт); контрольні запитання.
17.	Синтаксично керовані визначення і трансляція.	Знати призначення синтаксично керованого визначення і трансляції. Вміти застосовувати семантичні правила до граматичних правил і будувати синтаксичні дерева.	2	Лабораторна робота (звіт); контрольні запитання.
18.	Генерація проміжного і машинного коду.	Знати як генерується і оптимізується проміжний код. Вміти перетворювати проміжний код у машинний код з врахуванням наявних регістрів.	2	Лабораторна робота (звіт); контрольні запитання; тести.

4. Система оцінювання курсу

Накопичування балів під час вивчення дисципліни	
Види навчальної роботи	Максимальна кількість балів
Лекція	9
Лабораторні заняття	30
Самостійна робота	11
Залік/Екзамен	50
Максимальна кількість балів	100

5. Оцінювання відповідно до графіку навчального процесу

Види навчальної роботи	Номер навчального заняття (5 семестр)											Разом
	1	2	3	4	5	6	7	8	9	10	11	
Лекції	2	2	2	2	2	2	2	2	2			18
Лабораторні з-тя	6	6	6	6	6	6	6	6	6	6	6	66
Самостійна робота	1	1	1	1	1	1	2	2	2	2	2	16
Залік												
Всього за заняття	9	9	9	9	9	9	10	10	10	10	8	100

Види навчальної роботи	Номер навчального заняття (6 семестр)											Разом
	1	2	3	4	5	6	7	8	9	10	11	
Лекції	1	1	1	1	1	1	1	1	1			9
Лабораторні з-тя	3	3	3	3	3	3	3	3	3	3		30
Самостійна робота	1	1	1	1	1	1	1	1	1	1	1	11
Екзамен												50
Всього за заняття	9	9	9	9	9	9	9	9	10	10	8	100

Примітка: не рекомендується на один тиждень планувати кілька форм контролю.

6. Ресурсне забезпечення

Матеріально-технічне забезпечення	Мультимедіа, комп'ютерна лабораторія для виконання лабораторних робіт.
Література	
Основна (5 сем)	
<ol style="list-style-type: none">1. Методичні рекомендації до виконання курсової роботи з навчальної дисципліни "Системне програмування". [Електронний ресурс] / КНУВС; укл., Голота В.І. Когут І.Т., Грига В.М. – Електронні текстові дані (1 файл: 0,25 Мбайт). – Івано-Франківськ: КНУВС, 2025. – 37 с. Режим доступу https://lib-repo.pnu.edu.ua. – Назва з екрану.2. Системне програмування. Програмування на асемблері: комп'ютерний практикум [Електронний ресурс]: навч. посіб. для студ. освітньої програми «Комп'ютерні системи та мережі» спеціальності 123 «Комп'ютерна інженерія» / КПІ ім. Ігоря Сікорського; уклад. Порєв В.М. – Електронні текстові дані. – Київ: КПІ ім. Ігоря Сікорського, 2022. – 146 с.3. Архітектура комп'ютерних систем: мова асемблера [Електронний ресурс] : навчальний посібник / КПІ ім. Ігоря Сікорського; уклад.: Л. Ю. Гальчинський, О. В. Козленко. – Київ: КПІ ім. Ігоря Сікорського, 2022. – 160 с.4. Bartlett J. Learn to Program with Assembly: Foundational Learning for New Programmers / Jonathan Bartlett. – Apress, 2021. – 348 p.5. Hyde R. The Art of 64-Bit Assembly, Volume 1: x86-64 Machine Organization and Programming / Randall Hyde. – No Starch Press, 2021. – 1032 p.6. Kusswurm D. Modern X86 Assembly Language Programming: Covers x86 64-bit, AVX, AVX2, and AVX-512 / Daniel Kusswurm. – 2nd ed. – Apress, 2018. – 625 p.7. Hennessy J., Patterson D. – «Computer Architecture: A Quantitative Approach», Morgan Kaufmann, 2018. – 1–104 с.8. Lemieux G. – «Advanced Computer Architecture». Cambridge University Press, 2020. – 210–267 с.	
Додаткова (5 сем)	
<ol style="list-style-type: none">9. NASM - The Netwide Assembler [Електронний ресурс]. – Режим доступу: http://www.nasm.us/.10. Мікропроцесори Intel. Архів офіційного сайту Intel. [Електронний ресурс]. – Режим доступу: https://www.intel.com.11. Intel® 64 and IA-32 Architectures Software Developer Manuals. [Електронний ресурс]. – Режим доступу: https://www.intel.com/content/www/us/en/content-details/671488/intel-64-and-ia-32-architectures-optimization-reference-manual-volume-1.html12. Intel® 64 and IA-32 Architectures Software Developer Manuals. Intel Corporation. [Електронний ресурс]. – Режим доступу: https://www.intel.com/content/www/us/en/content-details/671488/intel-sdm.html.13. Microsoft Docs. Підтримка SIMD у Visual C++. [Електронний ресурс]. – Режим доступу: https://learn.microsoft.com/cpp.	

14. Intel Intrinsic Guide. Документація по intrinsic-функціях SIMD. [Електронний ресурс]. – Режим доступу:
<https://www.intel.com/content/www/us/en/docs/intrinsics-guide>.

Основна (бсем)

14. Основи проектування трансляторів: Інструкції та завдання до виконання лабораторних робіт та розрахунково-графічної роботи з дисципліни «Основи проектування трансляторів» : [Електронний ресурс] : навч. посіб. для студ. спеціальності 123 – «Комп'ютерна інженерія» / О. І. Марченко, О. О. Марченко ; КПІ ім. Ігоря Сікорського. – Електронні текстові дані (1 файл: 1,08 Мбайт). – Київ : КПІ ім. Ігоря Сікорського, 2021. – 118 с.
15. Формальні мови, граматики та автомати: Навчальний посібник/ Гавриленко С.Ю. – Харків: НТУ «ХПІ», 2021. – 133 с.
16. Clinton L. Jeffery. Build Your Own Programming Language: A programmer's guide to designing compilers, interpreters, and DSLs for solving modern computing problems. Packt Publishing, 2021. – 494 p.
17. Douglas Thain. Introduction to Compilers and Language Design: Second Edition. Independently published, 2020. – 247 p.
18. Anthony J. Dos Reis. Writing Interpreters and Compilers for the Raspberry Pi Using Python: Second Edition. Independently published, 2020. – 247 p.
19. Watt, David A. Programming language processor in Java: compilers and interpreters / Davit A. Watt and Deryck F. Brown. GB: Pearson EL, 2000. – 424 p.
20. Alfred Aho, Jeffrey Ullman, Ravi Sethi, Monica Lam. Compilers: Principles, Techniques, and Tools 2nd Edition. Addison-Wesley, 2007. – 1040 p.

Додаткова (6 сем)

21. Keith D. Cooper and Linda Torczon. Engineering a Compiler. 2012 Elsevier, Inc., 2012. – 800 p.
22. Andrew W. Appel, Maia Ginsburg. Modern Compiler implementation in C. Cambridge University press, 1998. – 544 с.
23. Allen I. Holub. Compiler design in C. Prentice-Hall, Inc. , 1990. – 984 p.
24. Jean-Paul Tremblay, Paul G. Sorenson. The theory and practice of compiler writing. McGraw-Hill Book Company, 1985. – 796 p.
25. Flex [Інтернет ресурс]: Генератор лексичних аналізаторів. Режим доступу:
<https://sourceforge.net/p/re-flex/wiki/Home/>
26. GNU Bison [Інтернет ресурс]: Генератор парсерів загального призначення. Режим доступу: <https://www.gnu.org/software/bison/>
27. PLY (Python Lex-Yacc) [Інтернет ресурс]: Лексичний і синтаксичний аналізатор на мові Python. Режим доступу: <https://www.dabeaz.com/py>
28. ANTLR [Інтернет ресурс]: Потужний генератор парсерів. Режим доступу:
<https://wwwantlr.org/>
27. LLVM [Інтернет ресурс]: Оптимізатор і генератор коду. Режим доступу:
<https://llvm.org/>

7. Контактна інформація

Кафедра	комп'ютерної інженерії та електроніки, вул. Шевченка, 57, ауд. 210 а, (0342)59-60-07, https://kkite.pnu.edu.ua/kkie@pnu.edu.ua
Викладач	к.т.н., доц. Голота В.І.
Контактна інформація викладача	viktor.holota@cnu.edu.ua

8. Політика навчальної дисципліни

Академічна доброчесність	Дотримання академічної доброчесності засновується на ряді положень та принципів академічної доброчесності, що регламентують діяльність здобувачів вищої освіти та викладачів університету: https://pnu.edu.ua/положення-про-запобігання-плагиату/ Не допускається підказування та списування під час здачі будь-яких робіт поточного, рубіжного, підсумкового контролю. Не допускається користування телефонами та будь-якими іншими електронними засобами під час здачі будь-яких робіт поточного, рубіжного, чи підсумкового контролю.
Пропуски занять (відпрацювання)	Можливість і порядок відпрацювання пропущених здобувачем освіти занять регламентується Положення про порядок організації та проведення оцінювання успішності здобувачів освіти ДВНЗ «Прикарпатського національного університету ім. Василя Стефаника» (введено в дію наказом ректора № 799 від 26.11.2019 р.; із внесеними змінами наказом № 212 від 06.04.2021 р.) .
Виконання завдання пізніше встановленого терміну	У разі виконання завдання здобувачем освіти пізніше встановленого терміну, без попереднього узгодження ситуації з викладачем, оцінка виставляється», відповідно до Положення про порядок організації та проведення оцінювання успішності студентів ДВНЗ «Прикарпатського національного університету ім. Василя Стефаника» (введено в дію наказом ректора № 799 від 26.11.2019 р.; із внесеними змінами наказом № 212 від 06.04.2021 р.) .
Невідповідна поведінка під час заняття	Невідповідна поведінка під час заняття регламентується рядом положень про академічну доброчесність та може призвести до відрахування здобувача вищої освіти «за порушення навчальної дисципліни і правил внутрішнього розпорядку вищого закладу освіти», відповідно до Положення про порядок переведення, відрахування та поновлення студентів вищих закладів освіти» (затверджене наказом Міністерства України № 245 від 15.07.1996 р.) .
Додаткові бали	Студент має змогу також отримати додаткові бали, пройшовши навчальний курс у вигляді неформальної освіти з отриманням сертифікату в межах тематики дисципліни

	<p>впродовж навчального семестру; взявши участь у науковому, освітньому чи прикладному проекті, конференції, круглому столі, інших видах наукової активності, які відповідають профілю дисципліни; опублікувавши наукову працю, яка відповідає профілю дисципліни. Відповідно до Положення про порядок організації та проведення оцінювання успішності студентів ДВНЗ «Прикарпатського національного університету ім. Василя Стефаника» (введено в дію наказом ректора № 799 від 26.11.2019 р.; із внесеними змінами наказом № 212 від 06.04.2021 р.). відповідні студенти можуть отримати додаткові бали на підставі рішенням кафедри.</p>
Неформальна освіта	<p>У випадку, коли студент приймав участь у програмі мобільності, можливе врахування отриманих оцінок в іншому навчальному закладі за умови відповідності навчальних планів.</p> <p>Можливість зарахування результатів неформальної освіти регламентується Положенням про порядок зарахування результатів неформальної освіти у ДВНЗ «Прикарпатський національний університет імені Василя Стефаника» (введено в дію наказом ректора № 819 від 29.11.2019; із внесеними змінами наказом № 80 від 12.02.2021 р.).</p> <p>Рекомендовані платформи: https://ua.udemy.com/; https://www.coursera.org/ https://prometheus.org.ua/</p>

Викладач



Голота В.І.