

**ДЕРЖАВНИЙ ВИЩИЙ НАВЧАЛЬНИЙ ЗАКЛАД
“УЖГОРОДСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ”
ІНЖЕНЕРНО-ТЕХНІЧНИЙ ФАКУЛЬТЕТ
КАФЕДРА КОМП’ЮТЕРНИХ СИСТЕМ ТА МЕРЕЖ**



ЗАТВЕРДЖУЮ

**Декан інженерно-технічного
факультету**

доц. Йолана ГОЛИК

“ 14 ” червня 2025 р.

РОБОЧА ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

СИСТЕМНЕ ПРОГРАМУВАННЯ

Рівень вищої освіти – перший (бакалаврський)

Галузь знань – 12 Інформаційні технології

Спеціальність – 123 Комп’ютерна інженерія

Освітня програма – Комп’ютерні системи та мережі

Статус дисципліни – обов’язкова

Мова навчання – українська


Ужгород – 2025

Робоча програма навчальної дисципліни «Системне програмування» для здобувачів спеціальності 123 Комп'ютерна інженерія, освітня програма «Комп'ютерні системи та мережі» – 23 с.


Розробник: Горват П.П., завідувач кафедри комп'ютерних систем та мереж, кандидат фізико-математичних наук, доцент.

Робочу програму розглянуто та затверджено на засіданні кафедри комп'ютерних систем та мереж

протокол № 13 від «25» червня 2025 р.

Завідувач кафедри  доц. Петро ГОРВАТ
(підпис) (прізвище та ініціали)

Схвалено науково-методичною комісією інженерно-технічного факультету
протокол № 6 від «27» червня 2025 р.

Голова науково-методичної комісії  доц. Володимир ЦИГИКА
(прізвище та ініціали) (підпис)

1. ОПИС НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

Найменування показників	Розподіл годин за навчальним планом	
	Денна / дистанційна форма здобуття освіти	
Кількість кредитів ЄКТС – 7	Рік підготовки:	
Загальна кількість годин – 210	другий	
Кількість модулів – 4	Семестр:	
	третій	четвертий
Тижневих годин: аудиторних – 3,3 / 2,9 год самостійної роботи студента – 3,3 / 3,1 год	Лекції:	
	30 год	24 год
	Практичні заняття:	
	-	-
Вид підсумкового контролю: екзамен, екзамен	Лабораторні заняття:	
	30 год	20 год
Форма підсумкового контролю: усна або письмова	Самостійна робота:	
	60 год	46 год
	Індивідуальні завдання	
	-	-

2. МЕТА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

Метою дисципліни "Системне програмування" є вивчення студентами фундаментальних основ архітектури сучасних процесорів та методики програмування на мові Асемблера, ознайомлення з проблемами, які стоять перед системним програмуванням і основними напрямками їх вирішення.

Згідно ОП «Комп'ютерні системи та мережі» вивчення дисципліни сприяє формуванню у здобувачів вищої освіти таких компетентностей:

- інтегральна (здатність розв'язувати складні спеціалізовані задачі та практичні проблеми під час професійної діяльності у комп'ютерній галузі або навчання, що передбачає застосування теорій та методів комп'ютерної інженерії і характеризується комплексністю та невизначеністю умов);

- загальні (ЗК1. Здатність до абстрактного мислення, аналізу і синтезу, ЗК2. Здатність вчитися і оволодівати сучасними знаннями, ЗК3. Здатність застосовувати знання у практичних ситуаціях);

- фахові (ФК3. Здатність створювати системне та прикладне програмне забезпечення комп'ютерних систем та мереж, ФК5. Здатність використовувати засоби і системи автоматизації проектування до

розроблення компонентів комп'ютерних систем та мереж, Інтернет додатків, кіберфізичних систем тощо).

3. ПЕРЕДУМОВИ ДЛЯ ВИВЧЕННЯ НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

Перелік дисциплін, засвоєння яких необхідне для вивчення даної дисципліни:

- програмування,
- алгоритми і структури даних,
- організація та функціонування комп'ютерів.

4. ОЧІКУВАНІ РЕЗУЛЬТАТИ НАВЧАННЯ

Відповідно до освітньої програми «Комп'ютерні системи та мережі», вивчення навчальної дисципліни повинно забезпечити досягнення здобувачами вищої освіти таких програмних результатів навчання (ПРН):

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

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

ПРН16. Вміти оцінювати отримані результати та аргументовано захищати прийняті рішення.

ПРН21. Якісно виконувати роботу та досягати поставленої мети з дотриманням вимог професійної етики.

Очікувані результати навчання, які повинні бути досягнуті здобувачами освіти після опанування навчальної дисципліни

Очікувані результати навчання з дисципліни	Шифр ПРН
Володіти знаннями основ архітектури сучасних процесорів на рівні програмно доступних елементів, синтаксису мови Асемблера, семантики директив та машинних інструкцій	ПРН1
Набути уміння та навички програмування на мові Асемблера, здійснювати зв'язок програм на мові Асемблера і мовах високого рівня	ПРН10
Володіти уміннями та навичками, необхідними для створення програм обробки переривань, програм управління зовнішніми пристроями	ПРН10, ПРН16, ПРН21
Знати основні принципи побудови Асемблерів, редакторів зв'язку, завантажувачів, налагоджувачів та уміти їх використовувати при розробці програм	ПРН1, ПРН21
Знати і уміти використовувати системні команди та системні дані різних режимів роботи мікропроцесорів, особливості програмування співпроцесорів	ПРН10, ПРН16, ПРН21

5. ЗАСОБИ ДІАГНОСТИКИ ТА КРИТЕРІЇ ОЦІНЮВАННЯ РЕЗУЛЬТАТІВ НАВЧАННЯ

Дисципліна «Системне програмування» складається з чотирьох модулів. Використовуються традиційні методи навчання: лекції, лабораторні заняття, самостійна та індивідуальна робота студентів, консультації.

Використовуються методи усного контролю та письмового контролю. Поточний контроль передбачає: опитування студентів під час практичних, лабораторних занять та опитування на лекціях; контрольні роботи, індивідуальні, самостійні завдання. Підсумковий контроль передбачає екзамен у кожному семестрі та захист курсового проекту.

Оцінка ECTS, яку студент отримує після вивчення дисципліни, визначається відповідно до рейтингу студента, який є середнім арифметичним рейтингів з окремих модулів. Рейтинг студента з кожного модуля складається з балів (до 100 балів) за такі види робіт:

1. Модульна контрольна робота (МКР) тривалістю 2 академічних години. Максимальна кількість балів за МКР – 60 балів.

2. Виконання практичних та лабораторних робіт, максимальна кількість балів за всі роботи – 50 балів.

3. Бали за самостійну роботи студентів, які нараховуються за виконання самостійних робіт, творчий підхід до виконання завдань, виконання завдань із удосконалення дидактичних матеріалів з дисципліни, виконання індивідуальних завдань – до 10 балів за всі види робіт.

Необхідною умовою допуску до екзамену є відсутність заборгованостей з лабораторних та практичних робіт та виконання модульних контрольних робіт.

Розподіл балів, які отримують студенти за 1 модуль

Лабораторні роботи					Самостійна робота	МКР	Сума
ЛР1	ЛР2	ЛР3					
10	15	15			10	50	100

Розподіл балів, які отримують студенти за 2 модуль

Лабораторні роботи					Самостійна робота / ЛР8	МКР	Сума
ЛР4	ЛР5	ЛР6	ЛР7				
10	10	10	10		10	50	100

Розподіл балів, які отримують студенти за 3 модуль

Лабораторні / практичні роботи					Самостійна робота	МКР	Сума
ЛР1	ЛР2	ЛР3					
10	10	10	10		10	50	100

Розподіл балів, які отримують студенти за 4 модуль

Лабораторні роботи					Самостійна робота	МКР	Сума
ЛР2	ЛР3	ЛР4	ЛР5	ЛР6			
10	10	10	10	10	10	40	100

Оцінювання окремих видів навчальної роботи з дисципліни

Вид діяльності здобувача вищої освіти	Модуль 1-2		Модуль 3-4	
	К-сть	Макс. к-сть балів	К-сть	Макс. к-сть балів
Лабораторні роботи	3 / 4	40 / 40	3 / 3	30 / 30
Практичні роботи			1 / 2	10 / 20
Самостійна робота	1	10	1	10
Модульна контрольна робота	1	50 / 50	1	50 / 40
Разом		100		100

Критерії оцінювання модульної контрольної роботи

Модульна контрольна робота проводиться у вигляді письмової роботи, яка містить чотири-п'ять завдань. Перші два-три завдання включають теоретичний матеріал, наступні одне-два завдання – це практичні завдання. Всі завдання при оцінюванні мають однакову вагу.

Критерії оцінювання підсумкового семестрового контролю

До складання екзамену допускаються лише студенти, які мають рейтинговий бал не менше 35. Екзамен з навчальної дисципліни студент може не складати, якщо він склав усі модулі та його влаштовує рейтингова оцінка. Студенти, які мають рейтинговий бал від 35 до 59 екзамен складають обов'язково. Студент може підвищити на екзамені оцінку, при цьому рейтингова оцінка не може бути зменшена.

За результатами виконання студентом навчальної програми впродовж семестру рекомендується виставляти заліки та екзамени без додаткового опитування за такою шкалою:

Сумарні бали	Оцінка ECTS	Екзамен (диф.залік)	Залік
90 – 100	A	Відмінно	Зараховано
82 – 89	B	Добре	
74 – 81	C		
64 – 73	D	Задовільно	
60 – 63	E		
35 – 59	FX	Незадовільно з можливістю повторного складання	Незараховано з можливістю повторного складання
1 – 34	F	Незадовільно з обов'язковим повторним вивченням дисципліни	Незараховано з обов'язковим повторним вивченням дисципліни

6. ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

6.1. Зміст навчальної дисципліни

1 модуль

Основи архітектури мікропроцесорів x86-64

Склад програмної моделі мікропроцесорів x86. Регістри загального призначення. Сегментні регістри мікропроцесора. Регістр-вказівник команди.

Регістр ознак. Ознаки, їх класифікація та призначення.

Типи даних цілочислового пристрою.

Режими роботи процесора. Реальний та захищений режим. Long mode.

Моделі пам'яті. Сегментована організація пам'яті. Плоска модель. Типи адрес пам'яті. Формування фізичної адреси у реальному режимі.

Способи адресації операндів. Безпосередня та регістрова адресація операндів. Пряма адресація операндів. Непряма регістрова адресація. Базова адресація. Індексна адресація. Базово-індексна адресація. Масштабування адрес.

Архітектурні особливості 64-розрядного процесора. Регістри архітектури x64, використання регістрів. Скалярні типи даних.

Основні директиви та операнди Асемблера

Директиви Асемблера. Керування моделями пам'яті та режимами адресації. Директиви визначення даних та констант. Директиви визначення сегментів та процедур. Директиви керування файлами. Оператори мови Асемблер.

Етапи створення програм мовою Асемблер

Підготовка тексту програми. Отримання об'єктного коду. Отримання виконуваного файлу. Налаштування програми. Редактори тексту. Налаштування середовища.

Операції пересилання даних

Команди пересилання даних загального призначення та їх особливості. Умовне пересилання. Стекова організація пам'яті. Команди для запису у стек та алгоритми їх роботи. Команди для читання із стеку, алгоритми їх роботи. Команди для роботи з адресами і вказівниками. Команди перетворення даних. Команди введення/виведення через порти. Команди пересилання бітів умов.

Основні арифметичні операції

Арифметичні операції. Команди додавання цілочислових даних. Інкремент. Команди розширення цілих чисел. Команди перетворення типів.

Віднімання цілочислових даних. Декремент. Команда порівняння.

Множення цілочислових даних. Знакове та беззнакове множення.

Команди ділення цілочислових даних.

Двійково-десяткові (BCD) та ACSII-числа. Додавання неупакованих BCD-чисел. Віднімання неупакованих BCD-чисел. Множення неупакованих BCD-чисел. Ділення неупакованих BCD-чисел. Додавання упакованих BCD-чисел. Віднімання упакованих BCD-чисел.

2 модуль

Логічні команди та команди зсуву

Логічні команди (команди булевих операцій) та їх використання. Команди перевірки та модифікації бітів. Команди сканування бітів.

Команди логічного зсуву. Команди арифметичного зсуву. Подвійний зсув. Команди циклічного зсуву. Команди установки байта за умовою. Команда test та її застосування.

Команди передачі керування

Мітки, опис та використання. Команда безумовного переходу. Типи переходів. Команда порівняння та її застосування. Умовна передача керування для випадку знакових чисел. Умовна передача керування для беззнакових чисел. Умовна передача керування за значенням ознак. Перехід за значенням регістра-лічильника. Команда організації циклів з лічильником. Команди для організації умовних циклів.

Програмування з використанням процедур

Поняття процедури. Виклик процедури та повернення із процедури. Угоди про виклик функцій. Передача аргументів і угоди іменування. Команди керування стеком при виконанні процедур. Організація процедур. Директиви та оператори для організації процедур. Непрямий виклик процедур. Використання загальних змінних. Виклик обробника переривання та повернення із нього.

Рядкові операції

Загальні принципи обробки рядків. Префікси повторення. Команда пересилання для рядків. Команди завантаження/зберігання для рядків. Команди порівняння та сканування для рядків.

Структури. Організація структур. Складні структури.

Захищений режим

Захищений режим. Рівні привілеїв. Захист за привілеями. Сегменти та сегментні регістри у захищеному режимі. Сегментні дескриптори. Глобальна дескрипторна таблиця. Регістр GDTR. Селектор. Перетворення адреси.

Класифікація переривань у захищеному режимі. Шлюзи. Дескриптори шлюзів. Виняткові (особливі) ситуації. Типи виняткових ситуацій. Обробка переривань у захищеному режимі. Таблиця дескрипторів переривань.

Багатозадачність. Сегмент стану задачі. Регістр задачі. Дескриптори сегмента стану задачі та шлюзу задачі. Керування задачами. Основні дії при переключенні задач.

Механізми захисту на рівні сегментів і сторінок. Перевірки ознак, границь, типів. Механізми захисту за рівнями привілеїв. Перевірки при звертанні до сегментів. Шлюзи виклику. Переключення стеку.

Long mode.

Перехід у long mode. Сегментація, механізм трансляції сторінок. Робота з пам'яттю. Переривання у long mode. Захист і багатозадачність. Режим сумісності.

3 модуль

Обробка дійсних чисел. Математичний співпроцесор

Типи даних співпроцесора. Формати звичайних чисел з рухомою комою (крапкою) Single, Double, Extended. Граничні значення, переведення десяткових чисел у дійсні формати. Формати цілих чисел та VCD-чисел. Спеціальні значення: нуль, нескінченність, денормалізовані числа, нечисла, невизначеність.

Програмна модель співпроцесора. Регістровий стек. Регістр тегів, слово тегів. Регістр керування, слово керування. Регістр стану, слово стану. Показчик команди, показчик даних. Режими адресації співпроцесора: класична стекова адресація; регістрова адресація; регістрова із звільненням вершини стеку; використання операндів у пам'яті.

Система команд співпроцесора. Класифікація та умовні позначення команд. Завантаження / збереження даних. Завантаження констант. Арифметичні команди: додавання; віднімання, множення, ділення, додаткові арифметичні команди. Трансцендентні команди. Порівняння операндів. Тестування операндів. Керування співпроцесором. Особливі (виняткові) ситуації співпроцесора та їх обробка: переповнення, антипереповнення; втрата точності, денормалізація; неіснуюча операція, ділення на нуль.

Макровизначення. Порівняння макровизначень та процедур. Розміщення макровизначень. Типи макросів. Створення макровизначень. Директиви та оператори у макровизначеннях. Додаткові макровизначення і директиви.

Архітектурні розширення

Історія та особливості мультимедійних розширень архітектури IA-32. Модель цілочислового MMX-розширення: регістри, стан співпроцесора. Особливості MMX-команд: арифметика з переносом, арифметика з насиченням. Класифікація MMX-команд. Команди передачі даних. Арифметичні команди. Команди порівняння, логічні команди, команди зсуву. Команди упакування/розпакування. Додаткові команди (Pentium III - IV). Технологія 3DNow!: регістри, типи даних, команди.

Модель XMM(SSE)-розширення: XMM-регістри, типи даних, регістр стану/керування. Система XMM-команд. Арифметичні команди. Скалярні та паралельні операції. Команди пересилання даних. Команди порівняння, перетворення даних, логічні, перерозподілу, керування станом, керування кешуванням.

Технологія SSE2: регістри та типи даних. Команди передавання даних. Використання масок. Арифметичні операції. Обробка 128-бітових дійсних чисел. Обробка 128-бітових цілих чисел. Технологія SSE3. Дублювання даних, завантаження змінних. Особливості команд додавання та віднімання. Використання SSE3 в розробці і оптимізації програм. SSSE3-доповнення

Технологія SSE4. Формати даних. Прискорення відео. Векторні операції. Обробка рядків. Технологія SSE5. Технологія AVX. Особливості використання команд AVX-розширення.

4 модуль

Кодування машинних команд

Байт коду операції у машинній команді. Префікси. Байт способу адресації у машинній команді. Байт масштабу, індексу та бази у машинній команді. Поля зміщення та безпосереднього операнда.

Машинні формати команд співпроцесора.

Загальна структура машинної інструкції у режимі IA-32e. Префікс REX у машинній інструкції. Спеціальні випадки використання префіксу REX. Зміщення та безпосередній операнд в інструкції. Адресація відносно RIP

Асемблери

Призначення асемблера. Призначення мови асемблера. Речення мови асемблера. Команди асемблера. Операнди команд асемблера. Директиви асемблера. Операнди директив асемблера. Структури (бази) даних асемблера. Структури даних першого проходу двопрохідного асемблера. Алгоритм роботи першого проходу двопрохідного асемблера. Структури даних другого проходу двопрохідного асемблера. Алгоритм роботи другого проходу двопрохідного асемблера.

Дизасемблери

Антиналагоджувальні прийоми

Антиналагоджувальні прийоми. Класифікація методів виявлення налагодження. Обробка виняткових ситуацій. Трасування за ознакою трасування. Трасування за шаблоном.

Трасування за часом виконання інструкцій.

Використання функцій мультимедійного таймера

Впровадження коду

PE-формат. Дослідження PE-файла.

Алгоритм впровадження коду

6.2. Структура навчальної дисципліни

№ з/п	ТЕМИ	Всього годин	З них				
			Лекції	Практичні	Лабораторні	Самостійна робота	Інд.робота
1	2	3	4	5	6	7	8
	Модуль № 1						
1.	Основи архітектури мікропроцесорів x86-64	8	2		2	4	
2.	Моделі пам'яті. Способи адресації операндів.	8	2		2	4	
3.	Основні директиви та операнди Асемблера	10	2		4	4	
4.	Етапи створення програм мовою Асемблер	8			4	4	
5.	Операції пересилання даних	8	2		2	4	
6.	Основні арифметичні операції	12	4		2	6	
7.	Двійково-десяткові (BCD) та ASCII-числа	4				4	
	Модульна контрольна робота № 1	2	2				
	Разом за модуль № 1	60	14		16	30	
	Модуль № 2						
1.	Логічні команди та команди зсуву	8	2		2	4	
2.	Команди передачі керування	9	2		3	4	
3.	Команди для організації циклів.	9	2		3	4	
4.	Програмування з використанням процедур	8	2		2	4	
5.	Рядкові операції	8	2		2	4	
6.	Структури. Складні структури	10	2		2	6	
7.	Захищений режим. Long mode	6	2			4	
	Модульна контрольна робота № 2	2	2				
	Разом за модуль № 2	60	16		14	30	
	Модуль № 3						
1.	Типи даних співпроцесора. Модель FPU	8	2		2	4	
2.	Система команд FPU. Макровизначення	8	2		2	4	
3.	Архітектурні розширення. MMX-команди.	8	2		2	4	
4.	Модель XMM(SSE)-розширення	8	2		2	4	
5.	Технології SSE2 – SSE4, AVX	8	2		2	4	
	Модульна контрольна робота № 3	5	2			3	
	Разом за модуль № 3	45	12		10	23	
	Модуль № 4						
1	Кодування машинних команд (protected mode)	8	2		2	4	
2	Кодування машинних команд (long mode)	8	2		2	4	
3	Асемблери. Дизасемблери	8	2		2	4	
4	Антиналагоджувальні прийоми	8	2		2	4	
5	Впровадження коду	8	2		2	4	

Модульна контрольна робота № 4	5	2			3	
Разом за модуль № 4	45	12		10	23	

6.3. Теми лабораторних занять

Модуль № 1

1.	Основи архітектури мікропроцесорів x86-64	2
2.	Моделі пам'яті. Способи адресації операндів.	2
3.	Основні директиви та операнди Асемблера	4
4.	Етапи створення програм мовою Асемблер	4
5.	Операції пересилання даних	2
6.	Основні арифметичні операції	2
	Всього	16

Модуль № 2

1.	Логічні команди та команди зсуву	2
2.	Команди передачі керування	3
3.	Команди для організації циклів.	3
4.	Програмування з використанням процедур	2
5.	Рядкові операції	2
6.	Структури. Складні структури	2
	Всього	14

Модуль № 3

1.	Макровизначення.	2
2.	Типи даних співпроцесора.	2
3.	Система команд співпроцесора.	2
4.	Архітектурні розширення. MMX-команди.	2
5.	Технологія SSE	2
	Всього	10

Модуль № 4

1.	Кодування машинних команд (protected mode)	2
2.	Кодування машинних команд (long mode)	2
3.	Антиналягоджувальні прийоми	2
4.	Антиналягоджувальні прийоми	2
5.	Впровадження коду	2
	Всього	10

6.4 Самостійна робота

№	Назва теми	К-сть годин
	Модуль № 1	
1.	Основи архітектури мікропроцесорів x86-64	4
8.	Моделі пам'яті. Способи адресації операндів.	4
9.	Основні директиви та операнди Асемблера	4
10.	Етапи створення програм мовою Асемблер	4
11.	Операції пересилання даних	4
12.	Основні арифметичні операції	6
13.	Двійково-десяткові (BCD) та ASCII-числа	4
	Модульна контрольна робота № 1	
	Разом за модуль № 1	30
	Модуль № 2	
1.	Логічні команди та команди зсуву	4
8.	Команди передачі керування	4
9.	Команди для організації циклів.	4
10.	Програмування з використанням процедур	4
11.	Рядкові операції	4
12.	Структури. Складні структури	6
13.	Захищений режим. Long mode	4
	Модульна контрольна робота № 2	
	Разом за модуль № 2	30
	Модуль № 3	
1.	Типи даних співпроцесора. Модель FPU	4
6.	Система команд FPU. Макровизначення	4
7.	Архітектурні розширення. MMX-команди.	4
8.	Модель XMM(SSE)-розширення	4
9.	Технології SSE2 – SSE4, AVX	4
	Модульна контрольна робота № 3	3
	Разом за модуль № 3	23
	Модуль № 4	
1	Кодування машинних команд (protected mode)	4
2	Кодування машинних команд (long mode)	4
3	Асемблери. Дизасемблери	4
4	Антиналагоджувальні прийоми	4
5	Впровадження коду	4
	Модульна контрольна робота № 4	3
	Разом за модуль № 4	23

7. ІНСТРУМЕНТИ, ОБЛАДНАННЯ ТА ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ, ВИКОРИСТАННЯ ЯКИХ ПЕРЕДБАЧАЄ НАВЧАЛЬНА ДИСЦИПЛІНА

Практичні та лабораторні роботи виконуються на персональних комп'ютерах із встановленою операційною системою Windows, Linux або іншою.

Програмне забезпечення:

- транслятор MASM64 або аналогічний, x64Dbg, LordPE або PETools

8. РЕКОМЕНДОВАНІ ЛІТЕРАТУРНІ ДЖЕРЕЛА

1. Intel® 64 and IA-32 Architectures Software Developer's Manual. Volume 1, 2a, 2b, 2c, 2d, 3a.

2. Рисований О.М. Системне програмування [Текст]: підручник для студентів напрямку "Комп'ютерна інженерія" вищих навчальних закладів в 2-х томах. Том 1. – Видання четверте: виправлено та доповнено – Харків: "Слово", 2015. – 576 с.

3. Рисований О.М. Системне програмування [Текст]: підручник для студентів напрямку "Комп'ютерна інженерія" вищих навчальних закладів в 2-х томах. Том 2. – Видання четверте: виправлено та доповнено – Харків: "Слово", 2015. – 378 с.

4. Архітектура комп'ютерів. Машинні команди та програмування на асемблері: навчальний посібник / О.С. Тонкошкур, О.Б. Гниленко, Н.О. Матвеева, О.С. Морозов. – □ Д.: Вид-во «Нова Ідеологія», 2018. – 179 с.

5. X86 Opcode and Instruction Reference - <http://ref.x86asm.net/index.html>

**Перелік питань для підготовки до модульного контролю
та самостійної роботи
Модуль № 1**

Перше завдання: дати чітку відповідь щодо реєстрів, роботи з пам'яттю, режимів роботи, подання чисел

Друге завдання: навести загальний формат команди, пояснити особливі випадки, операнди, вплив на ознаки, навести приклади

Третє завдання: вказати та пояснити результат подання чисел, виконання команд, обчислення адреси тощо, за можливості підтвердити результатами виконання програми і т.д.

Перше завдання:

1. Назвати основні групи реєстрів архітектури x86-64 та їх склад
2. Реєстри загального призначення: назви, розрядність, використання.
3. Сегментні реєстри: назви, розрядність, використання.
4. Реєстр - вказівник інструкцій (команд).
5. Реєстр ознак. Ознаки, їх класифікація та використання.
6. Моделі пам'яті: плоска, сегментована, реального режиму адресації.
7. Адреси у форматі сегмент : зсув (зміщення).
8. Режими адресації x86-64: безпосередня, реєстрова, пряма
9. Режими адресації x86-64: непряма реєстрова, базова, індексна, базово-індексна
10. Режими адресації x86-64: масштабована індексна, масштабована базово-індексна, базово-індексна зі зміщенням, масштабована базово-індексна зі зміщенням.
11. Цілі беззнакові числа розміром 1, 2, 4, 8 Байт: подання в пам'яті, діапазони представлення.
12. Цілі знакові числа розміром 1, 2, 4, 8 Байт: подання в пам'яті, діапазони представлення.
13. Особливості 64-бітних реєстрів
14. Директиви оголошення (визначення) даних.

Друге завдання:

1. Команда пересилання даних та її особливості.
2. Команди обміну даними та їх особливості.
3. Команди для запису у стек та алгоритми їх роботи
4. Команди для читання із стеку, алгоритми їх роботи
5. Додавання цілочислових даних (ADD, INC)
6. Додавання цілочислових даних (ADC, XADD)
7. Віднімання цілочислових даних (SUB, SBB)
8. Віднімання цілочислових даних (DEC, NEG)
9. Множення цілочислових даних (MUL)
10. Команди ділення цілочислових даних (IDIV)
11. Множення цілочислових даних (IMUL)
12. Команди ділення цілочислових даних (DIV)

13. Двійково-десяткові (BCD) та ASCII-числа
14. Арифметичні операції над упакованими BCD-числами
- Третє завдання:
1. Знайти лінійну адресу у реальному режимі роботи процесора за значеннями покажчика: 09FA:0EE0; 0E0A:09DD
 2. Знайти лінійну адресу у реальному режимі роботи процесора за значеннями покажчика: 013D:1AA2; 2D13:111F
 3. Одержати машинне представлення числа -987 у всіх можливих цілочислових форматах.
 4. Одержати машинне представлення числа 109 у всіх можливих цілочислових форматах.
 5. Одержати машинне представлення чисел 16385 та -4095 у всіх можливих цілочислових форматах.
 6. За машинним представленням чисел 0345Ah знайти значення у цілочислових форматах.
 7. За машинним представленням числа 078h знайти значення у цілочислових форматах.
 8. Для опису даних:


```
x db -4, -1, 0, 5
y dw 1000h, 3000h, 2000h, 4000h
z dd 8, 5, 9, 85
```

 вказати значення приймача після кожної команди


```
mov AL, x
mov AH, [x+3]
add AX, [y+4]
```
 9. Вказати значення приймача та ознак (sf, cf, zf, of) після кожної команди


```
mov CL, 0FFh
mov CH, 12h
inc CX
sub AX, CX
adc AL, CL
```
 10. Для опису даних:


```
x db -4, -1, 0, 5
y dw 1000h, 3000h, 2000h, 4000h
z dd 8, 5, 9, 85
```

 вказати значення приймача після кожної команди


```
mov AL, x
mov AH, [x+3]
add AX, [y+4]
mov BX, [z-4]
sub ECX, EDX
```
 11. Вказати значення приймача та ознак (sf, cf, zf, of) після кожної команди


```
mov AL, 80h
add AL, 0A0h
```

```

mov BX, 0FFE2h
sub BX, 1010h
mov CL, 0FFh
mov CH, 12h

```

12. Вказати значення приймача та ознак (sf, cf, zf) після кожної команди

```

mov AX, 353h
mov DX, 0
mov CX, 100h
mul CX

```

13. Вказати значення приймача після кожної команди

```

mov AX, 81h
mov BL, 10h
div BL
mov AX, 0F2
idiv BL

```

14. Вказати значення приймача після кожної команди

```

mov AL, E1h
mov BL, 10h
add AL, 2Ah
adc BL, 0

```

Модуль № 2

Перше завдання — теоретичне питання: описати особливості режимів роботи процесора

1. Режим реальних адрес: особливості та обмеження режиму
2. Режим реальних адрес: розрядність адрес та даних; обчислення фізичної адреси
3. Особливості та обмеження у захищеному режимі, розрядність адрес та даних
4. Рівні привілей у захищеному режимі
5. Сегменти та сегментні дескриптори у захищеному режимі
6. Дескрипторні таблиці, регістри дескрипторних таблиць у захищеному режимі
7. Логічні адреси. Обчислення логічної та лінійної адреси у захищеному режимі
8. Переривання та обробка переривань у реальному режимі
9. Переривання та обробка переривань у захищеному режимі
10. Багатозадачність, поняття задачі, сегмент стану задачі
11. Реалізація багатозадачності, регістр задачі, переключення задач.
12. Регістри та система команд у 64-бітному режимі
13. Сегменти та дескриптори сегментів
14. Перехід у 64-бітний режим
15. Віртуальні та фізичні адреси у 64-бітному режимі
16. Сторінки та дескриптори сторінок

17. Трансляція сторінок (розмір сторінки - 4 КБайт)
18. Трансляція сторінок (розмір сторінки - 2 МБайт)
19. Трансляція сторінок (розмір сторінки - 1 ГБайт)
20. Виняткові ситуації (переривання) та їх обробка
21. Захист та багатозадачність

Друге завдання — описати асемблерну(і) команду(и), вказати можливі операнди, вплив на ознаки, особливі ситуації

1. Логічна команда AND та її використання
2. Логічна команда OR та її використання
3. Логічна команда NOT та її використання
4. Логічна команда XOR та її використання
5. Логічний зсув, команди та їх застосування
6. Подвійний зсув, команди та їх застосування
7. Арифметичний зсув, команди та їх застосування
8. Циклічний зсув, команди та їх застосування
9. Команди перевірки і модифікації бітів
10. Команди сканування бітів
11. Команда порівняння CMP
12. Команда перевірки TEST
13. Команди умовного пересилання CMOVcc
14. Команди установки байта за умовою SETcc
15. Команди для роботи з ознаками (STx, SLx, SMx)
16. Команди для роботи з ознаками (LAHF, SAHF)

Третє завдання — описати реалізацію розгалужень, циклів, обробки рядків

1. Дані, опис та використання
2. Мітки, опис та використання
3. Команда безумовного переходу
4. Типи переходів
5. Умовна передача керування для випадку знакових чисел
6. Умовна передача керування для беззнакових чисел
7. Умовна передача керування за значенням ознак
8. Перехід за значенням регістра-лічильника
9. Виклик процедур та повернення із процедур
10. Організація циклів
11. Команда організації циклів з лічильником
12. Команди для організації умовних циклів
13. Рядкові команди: завантаження/зберігання
14. Рядкові команди: порівняння/сканування
15. Рядкові команди: пересилання
16. Робота з портами. Рядкові команди для роботи з портами

Четверте завдання — вказати і пояснити результати виконання команд асемблера

1. Вказати значення приймача та ознак (sf, cf, zf, of) після кожної команди


```
mov AL, 52
mov AH, AL
mov CL, 0FFh
mov CH, 12h
inc CX
cmp AX, CX
```
2. Вказати значення приймача та ознак (sf, cf, zf, of) після кожної команди


```
mov AL, 80h
add AH, 0A0h
mov CL, 0FFh
mov CH, 12h
inc CX
cmp AX, CX
```
3. У вказаних місцях (а, б, в, г) послідовності команд вкажіть значення регістра AL у двійковій формі:


```
mov al, 00001111b
and al, 00111011b ;а) AL = ???
mov al, 6Dh
and al, 4Ah ;б) AL = ???
mov al, 00001111b
or al, 61h ;в) AL = ???
mov al, 94h
xor al, 37h ;г) AL = ???
```
4. У вказаних місцях (а, б, в, г) послідовності команд вкажіть значення регістра AL у шістнадцятковій формі:


```
mov al, 7Ah
not al ;а) AL = ???
mov al, 3Dh
and al, 74h ;б) AL = ???
mov al, 9Bh
or al, 35h ;в) AL = ???
mov al, 72h
xor al, 0DCh ;г) AL = ???
```
5. У вказаних місцях (а, б, в, г) послідовності команд вкажіть значення регістра AL у шістнадцятковій формі:


```
mov al, 7Ah
not al ;а) AL = ???
mov al, 3Dh
and al, 74h ;б) AL = ???
mov al, 9Bh
or al, 35h ;в) AL = ???
mov al, 72h
xor al, 0DCh ;г) AL = ???
```

6. У вказаних місцях (а, б, в) наведеної нижче послідовності команд вкажіть значення регістра AL :
- ```
mov al, 0D4h
shr al, 1 ; а) AL = ?
mov al, 0D4h
sar al, 1 ; б) AL = ?
mov al, 0D4h
shl al, 1 ; в) AL = ?
```
7. У вказаних місцях (а, б) наведеної нижче послідовності команд вкажіть значення регістра AL :
- ```
mov al, 0D4h
rcl al, 1      ; а) AL = ?
mov al, 0D4h
rcr al, 3      ; б) AL = ?
```
8. У вказаних місцях (а, б) наведеної нижче послідовності команд вкажіть значення регістра AL :
- ```
mov al, 0D4h
shr al, 1 ; а) AL = ?
mov al, 0D4h
shl al, 1 ; б) AL = ?
```
9. У вказаних місцях (а, б) наведеної нижче послідовності команд вкажіть значення регістра AL :
- ```
mov al, 0D4h
ror al, 3      ; а) AL = ?
mov al, 0D4h
rcl al, 1      ; б) AL = ?
```
10. За допомогою однієї команди занулити старші 8 біт регістра AX, не змінюючи при цьому значення молодших 8 бітів.
11. За допомогою однієї команди встановити в одиницю старші 8 біт регістра AX, не змінюючи при цьому значення молодших 8 бітів.
12. Як, не користуючись командою NOT, інвертувати всі біти регістра EAX?
13. За допомогою якої команди можна встановити ознаку ZF, якщо в регістрі EAX знаходиться парне значення, і скинути ознаку, якщо значення непарне?
14. У вказаних місцях (а, б, в) послідовності команд вкажіть стан ознак cf, zf, of і sf:
- ```
mov AL, 00001111b
cmp AL, 2 ; а)
mov AL, 6
cmp AL, 5 ; б)
mov AL, 5
cmp AL, 7 ; в)
```

15. Що знаходиться у регістрі EDX після виконання команд:

```
mov edx, eax
cmp eax, ecx
jae L1
mov edx, ecx
```

L1:

16. Чи буде виконано перехід на мітку Target

```
mov ax, 8109h
cmp ax, 26h
jg Target
```

### Модуль 3

1. Звичайні типи даних співпроцесора: формати чисел з рухомою комою (крапкою).
2. Звичайні типи даних співпроцесора: формати цілих чисел та BCD-чисел.
3. Особливі типи даних співпроцесора. Спеціальні числові значення, їх застосування.
4. Програмна модель співпроцесора: регістровий стек, регістр тегів, слово тегів.
5. Програмна модель співпроцесора: регістр керування, слово керування.
6. Програмна модель співпроцесора: регістр стану, слово стану.
7. Програмна модель співпроцесора: покажчик команди, покажчик даних.
8. Режими адресації співпроцесора: класична стекова адресація. Приклади.
9. Режими адресації співпроцесора: регістрова адресація. Приклади.
10. Режими адресації співпроцесора: регістрова із звільненням вершини стеку.
11. Режими адресації співпроцесора: використання операндів у пам'яті. Приклади.
12. Система команд співпроцесора: класифікація та умовні позначення команд.
13. Команди передачі даних для співпроцесора. Завантаження констант.
14. Арифметичні команди співпроцесора: формати команди додавання.
15. Арифметичні команди співпроцесора: віднімання, множення, ділення та інші.
16. Трансцендентні команди співпроцесора: формат та особливості застосування.
17. Команди порівняння та тестування операндів. Організація умовних переходів.
18. Команди керування співпроцесором. Завантаження та зберігання середовища.
19. Особливі (виняткові) ситуації співпроцесора та їх обробка.
20. Машинні формати команд співпроцесора.
21. Історія та особливості мультимедійних розширень архітектури IA-32.
22. Модель цілочислового MMX-розширення: регістри, стан співпроцесора.

23. Особливості MMX-команд: арифметика з переносом, арифметика з насиченням.
24. Система MMX-команд: класифікація команд, команди передачі даних.
25. Система MMX-команд: арифметичні команди.
26. Система MMX-команд: команди порівняння, логічні команди, команди зсуву.
27. Система MMX-команд: команди упакування/розпакування.
28. Система MMX-команд: додаткові команди (Pentium III).
29. Технологія 3DNow!: регістри, типи даних, команди.
30. Модель XMM(SSE)-розширення: XMM-регістри, типи даних
31. Модель XMM-розширення: регістр стану/керування
32. Система XMM-команд: арифметичні команди. Скалярні та паралельні операції.
33. Система XMM-команд: класифікація, команди пересилання даних.
34. Система XMM-команд: порівняння, перетворення даних, логічні, перерозподілу.
35. Система XMM-команд: керування станом, керування кешуванням.
36. Технологія SSE2: регістри та типи даних.
37. Технологія SSE2: обробка 128-бітових дійсних чисел.
38. Технологія SSE2: обробка 128-бітових цілих чисел.
39. Технологія SSE3. Дублювання даних, завантаження змінних.
40. Особливості команд додавання та віднімання.
41. Використання SSE3 в розробці і оптимізації програм.
42. SSSE3-доповнення
43. Технологія SSE4. Формати даних. Прискорення відео.
44. Векторні операції. Обробка рядків.
45. Технологія SSE5.
46. Технологія AVX.
47. Особливості використання команд AVX-розширення.

#### **Модуль 4**

1. Загальна структура машинної інструкції, довжина машинної інструкції
2. Префікси та їх роль префіксів у машинній інструкції
3. Можливі формати коду операції
4. Байт ModR/M, приклади кодування
5. Байт SIB, приклади кодування. Зміщення та безпосередній операнд в інструкції
6. Загальна структура машинної інструкції у режимі IA-32e
7. Префікс REX у машинній інструкції
8. Спеціальні випадки використання префіксу REX
9. Зміщення та безпосередній операнд в інструкції
10. Адресація відносно RIP
11. Призначення асемблера.
12. Призначення мови асемблера.

13. Речення мови асемблера.
14. Команди асемблера.
15. Операнди команд асемблера.
16. Директиви асемблера.
17. Операнди директив асемблера.
18. Структури (бази) даних асемблера.
19. Структури даних першого проходу двопрхідного асемблера.
20. Алгоритм роботи першого проходу двопрхідного асемблера.
21. Структури даних другого проходу двопрхідного асемблера.
22. Алгоритм роботи другого проходу двопрхідного асемблера.
23. Однопрхідний асемблер.
24. Багатопрхідний асемблер.
25. Антиналагоджувальні прийоми.
26. Класифікація методів виявлення налагодження.
27. Обробка виняткових ситуацій.
28. Трасування за ознакою трасування.
29. Трасування за шаблоном.
30. Трасування за часом виконання інструкцій.
31. Використання функцій мультимедійного таймера
32. PE-формат.
33. Дослідження PE-файла.
34. Алгоритм впровадження коду