МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ЗАХІДНОУКРАЇНСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
ФАКУЛЬТЕТ КОМП'ЮТЕРНИХ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ

ЗАТВЕРДЖУЮ:
Декан факультету комп'ютерних інформаційних технологій
Ігор ЯКИМЕНКО
_____
"29" _____2025 р.

ЗАТВЕРДЖУЮ:
Проректор з науково-педагогічної роботи
Віктор ОСТРОВЕРХОВ
_____
"29" _____2025 р.

РОБОЧА ПРОГРАМА

з дисципліни «**Системне програмування та архітектура комп'ютерів**»

ступінь вищої освіти – **бакалавр**

галузь знань – **F "Інформаційні технології"**

спеціальність – **F3 „Комп'ютерні науки"**

освітньо-професійна програма – **„ Штучний інтелект"**

Кафедра інформаційно-обчислювальних систем і управління

Форма навчання	Курс	Семестр	Лекції (год.)	Лабораторні заняття (год.)	ІРС (год.)	Тренінг (год.)	Самост. робота студ. (год.)	Разом (год.)	Екз. (сем.)
Денна	1	2	30	30	4	8	78	150	2

Тернопіль – ЗУНУ
2025

Робоча програма складена на основі освітньо–професійної програми «Штучний інтелект» підготовки бакалавра галузі знань F “Інформаційні технології” спеціальності F3 «Комп’ютерні науки», затвердженої Вченою радою ЗУНУ (26 червня 2025 р., протокол N 8)

Робочу програму склав: доцент кафедри ІОСУ, к.т.н. Олександр ОСОЛІНСЬКИЙ

Робоча програма затверджена на засіданні кафедри інформаційно-обчислювальних систем і управління, протокол № 1 від 26 серпня 2025 р.

В.о. завідувача кафедри Надія ВАСИЛЬКІВ

Гарант освітньо-професійної програми " Штучний інтелект ", к.т.н, доцент Василь КОВАЛЬ

СТРУКТУРА РОБОЧОЇ ПРОГРАМИ НАВЧАЛЬНОЇ ДИСЦИПЛІНИ "СИСТЕМНЕ ПРОГРАМУВАННЯ ТА АРХІТЕКТУРА КОМП'ЮТЕРІВ"

1. Опис дисципліни "Системне програмування та архітектура комп'ютерів"

Дисципліна «Системне програмування та архітектура комп'ютерів»	**Галузь знань, спеціальність, СВО**	**Характеристика навчальної дисципліни**
Кількість кредитів – 5	Галузь знань – F "Інформаційні технології"	**Статус дисципліни:** обов'язкова дисципліна циклу професійної підготовки **Мова навчання:** Українська
Кількість залікових модулів – 5	Спеціальність – F3 «Комп'ютерні науки»	Рік підготовки: *1* Семестр: *Денна* – 2
Кількість змістових модулів – 2	Освітньо-професійна програма «Штучний інтелект»	Лекції: *Денна* –30 год. Лабораторні заняття: *Денна* – 30 год.
Загальна кількість годин – 150	Ступінь вищої освіти – бакалавр	Самостійна робота: *Денна* – 78 год., Тренінг *Денна* – 8 год. Індивідуальна робота: *Денна* – 4 год.
Тижневих годин – 10, з них аудиторних – 4 год.		Вид підсумкового контролю – екзамен

2. Мета і завдання дисципліни "Системне програмування та архітектура комп’ютерів "

2.1. Мета вивчення дисципліни

Метою дисципліни “Системне програмування та архітектура комп’ютерів” є формування і засвоєння у здобувачів вищої освіти необхідних теоретичних знань з основ побудови та архітектури сучасних комп’ютерів і практичних навичок з основ системного програмування.

2.2. Завдання вивчення дисципліни

Завдання дисципліни “Системне програмування та архітектура комп’ютерів” полягає у формуванні у студентів принципів побудови і функціонування сучасних комп’ютерів і комп’ютерних систем, їх структуру, порядок виконання команд і програм у комп’ютері, принципів побудови та функціонування пристроїв сучасних комп’ютерних систем, основні принципи створення програм мовою Асемблера. Здобути практичні навички використання основних директив, операторів, процедур та бібліотек мови Асемблер.

2.3. Найменування та опис компетентностей, формування яких забезпечує вивчення дисципліни:

СК12. Здатність забезпечити організацію обчислювальних процесів в інформаційних системах різного призначення з урахуванням архітектури, конфігурування, показників результативності функціонування операційних систем і системного програмного забезпечення.

2.4. Передумови для вивчення дисципліни

Вивчення курсу „Системне програмування та архітектура комп’ютерів” передбачає наявність систематичних та ґрунтовних знань із дисциплін, «Основи комп’ютерних наук» та «Основи програмування»

2.5. Результати навчання

ПР13. Володіти мовами системного програмування та методами розробки програм, що взаємодіють з компонентами комп’ютерних систем, знати мережні технології, архітектури комп’ютерних мереж, мати практичні навички технології адміністрування комп’ютерних мереж та їх програмного забезпечення.

3. Програма навчальної дисципліни «Системне програмування та архітектура комп'ютерів»

Змістовий модуль 1. Системне програмування

Тема 1. **Програмування мовою Асемблер**

Мова Асемблер, як основа системного програмування. Екскурс в історію мови Асемблер. Етапи створення програми мовою Асемблер.

Тема 2. **Відлагодження програми**

Стратегії відлагодження програм мови асемблера. Відлагоджувач GDB. Відлагоджувач DDD

Тема 3. **Основні директиви та оператори мови асемблера. Операції пересилання даних**

Директиви асемблера. Оператори мови Асемблер. Подання даних. Команди пересилання даних загального призначення. Команди роботи зі стеком, з адресами та вказівками. Команди перетворення даних, введення та виведення в порт, пересилання бітів умов.

Тема 4. **Основні арифметичні операції**

Команди додавання чисел. Команди перетворення типів. Команди віднімання та множення. Команди ділення. Команди двійково-десяткової арифметики. Точки зупину.

Тема 5. **Логічні команди, команди зсуву та передання керування**

Логічні команди булевих операцій та їх використання. Команди зсування. Команди безумовного та умовного передання керування. Команди керування циклами.

Тема 6. **Процедури**

Процедури. Базові поняття. Визначення процедури. Виклик процедури. Повернення до точки виклику. Використання пам'яті та регістрів у процедурах.

Тема 7. **Обробка стрічок та організація вводу-виводу**

Процедури. Базові поняття. Визначення процедури. Виклик процедури. Повернення до точки виклику. Використання пам'яті та регістрів у процедурах. Файловий ввід-вивід. Виведення повідомлень і чисел. Введення з консолі.

Змістовий модуль 2. Архітектура комп'ютерів

Тема 8. **Основні поняття про сучасний комп'ютер**

Класична архітектура комп'ютера. Організація зв'язків між функціональними вузлами комп'ютера. Основні характеристики комп'ютера.

Тема 9. **Архітектури фон Неймана та Гарвадська архітектура. Ієрархічний принцип побудови апаратних та програмних засобів комп'ютерів**

Поняття архітектури комп'ютера. Завдання етапу розробки архітектури. Історичний огляд архітектур комп'ютерів. Архітектурні принципи Джона фон Неймана. Гарвардська архітектура. Типи сучасних комп'ютерів.

***Тема 10.* Цифрові логічні схеми**

Прості логічні елементи. Логічні функції. Комбінаційні схеми. Програмовані логічні інтегральні схеми. Арифметико-логічний пристрій. Послідовні схеми. Тригери. Суматори.

***Тема 11.* Організація пам'яті**

Основні операції з пам'яттю. Типи пам'яті. Блоки пам'яті. Розширення пам'яті. Адресація. Принципи зберігання багатобайтних даних. Вирівнювання даних.

***Тема 12.* Системи команд мікропроцесорів. Структура і формати команд. Способи адресації операндів**

Системи команд процесора. Принципи формування системи команд комп'ютера. Формати команд. Загальна характеристика способів адресації операндів. Безпосередня, пряма і непряма адресація. Відносна адресація. Базова адресація

***Тема 13.* Алгоритми організації обчислювальних процесів та їх архітектурні рішення**

Загальні відомості про операції та їхній розподіл на групи. Алгоритми виконання арифметичних операцій над двійковими числами та їхні архітектурні рішення. Алгоритми і пристрої для виконання логічних операцій. Особливості виконання операцій десяткової арифметики. Алгоритм додавання і віднімання чисел з плаваючою крапкою.

***Тема 14.* Універсальні та спеціалізовані мікропроцесори**

Характеристика і класифікація мікропроцесорів. Особливості архітектури мікропроцесорів фірми Intel. Структура мікропроцесора. Новітні архітектури спеціалізованих процесорів GPU та TPU та їх особливості використання.

4. Структура залікового кредиту з дисципліни «Системне програмування та архітектура комп'ютерів»

Денна форма навчання

Тема	*Кількість годин*					
	Лекції	Лаборат. заняття	Індивід. робота	Тренінг	Самостійна робота	Контрольні заходи
Змістовий модуль 1 Системне програмування						
Тема 1. Програмування мовою Асемблер	2	2	2	4	6	Опитування під час заняття
Тема 2. Відлагодження програми	2	2			6	
Тема 3. Основні директиви та оператори мови асемблера. Операції пересилання даних	2	2			6	
Тема 4. Основні арифметичні операції	2	2			6	
Тема 5. Логічні команди, команди зсуву та передання керування	2	2			6	
Тема 6. Процедури	2	2			6	
Тема 7. Обробка стрічок та організація вводувиводу	2	-			6	
Змістовий модуль 2 Архітектура комп'ютерів						
Тема 8. Основні поняття про сучасний комп'ютер	2	-	2	4	6	Опитування під час заняття
Тема 9 Архітектури фон Неймана та Гарвадська архітектура. Ієрархічний принцип побудови апаратних та програмних засобів комп'ютерів	2	2			6	
Тема 10. Цифрові логічні схеми	2	4			6	
Тема 11 Організація пам'яті	2	2			6	
Тема 12. Системи команд мікропроцесорів. Структура і формати команд. Способи адресації операндів	2	4			4	
Тема 13. Алгоритми організації обчислювальних процесів та їх архітектурні рішення	2	4			4	
Тема 14. Універсальні та спеціалізовані мікропроцесори	4	2			4	
Разом	30	30	4	8	78	

5. Тематика лабораторних занять

Лабораторна робота №1.

Тема: Ознайомлення з програмуванням на мові Асемблер для IBM-сумісних комп'ютерів

Мета. Ознайомлення з структурою Асемблерних програм. Особливості створення програми на мові Асемблера. Процес обробки програми на мові асемблера. Правила оформлення програм на мові Асемблера. Робота з транслятором NASM. Робота з компонувальником LD.

Лабораторна робота №2.

Тема: Робота з файлом лістингу

Мета. Ознайомлення з правами доступу до файлів. Призначення файлу лістингу. Структура файлу лістингу.

Лабораторна робота №3.

Тема: Відлагоджувальник GDB

Мета: Вивчення методів пошуку помилок. Пошук розміщення помилки. Визначення причини помилки. Виправлення помилок.

Лабораторна робота №4.

Тема: Візуальне відлагодження

Мета: Вивчення основних можливостей відлагоджувальника DDD. Робота з вікном регістрів. Робота з вікном дампу пам'яті. Робота з вікном стану стеку. Дослідження програми на мові Асемблера в DDD.

Лабораторна робота №5.

Тема: Процедури

Мета: Робота з стеком. Реалізація стеку. Стекові операції, базові інструкції. Додаткові інструкції. Операції стеку над прапорами стану. Операції стеку для регістрів загального призначення.

Лабораторна робота №6.

Тема: Режими адресації

Мета: Вивчення основних режимів адресації. Робота з масивами.

Лабораторна робота №7.

Тема: Арифметичні операції

Мета: Вивчення основних арифметичних операцій на мові Асемблера. Прапори стану. Арифметичні інструкції.

Лабораторна робота №8.

Тема: Бітові операції

Мета: Вивчення команд зсуву над окремими бітами операндів. Зсув вліво або вправо. Вивчення команд циклічного зсуву вправо, вліво, вправо з переносом, вліво з переносом.

Лабораторна робота №9.

Тема: Вивчення стрічкових інструкцій.

Мета: Робота з стрічковими примітивами. Префікс повторення стрічкового примітиву. Примітив порівняння стрічок.

Лабораторна робота №10.

Тема: Робота з файлами

Мета: Робота з стрічковим примітивом пошуку. Робота з файлами.

Лабораторна робота №11.
Тема: Операції з плаваючою крапкою.
Мета: Робота з плаваючою крапкою на мові Асемблера. Інструкції з плаваючою крапкою.

Лабораторна робота №12.
Тема: Синтез цифрових логічних схем.
Мета: Побудова комбінаційних схем. Послідовні схеми. Тригери. Суматори. Регістри.

6. Самостійна робота

Самостійна робота студентів з дисципліни «Системне програмування та архітектура комп'ютерів» передбачає виконання індивідуальних проектів, які спрямовані на оволодіння навичками системного програмування мовою Асемблер, стратегіями відлагодження програм, застосування основних директив та операторів, арифметичних, логічних операцій та команд пересилання даних, обробку та організацію вводу-виводу в комп'ютерних системах. Також самостійна робота включає опанування такими навичками, як сучасні архітектури комп'ютерних систем, побудову цифрових логічних схем, організацію пам'яті, основні операції з пам'яттю та її типи, основні алгоритми обробки інформації та їх архітектурні рішення для цих алгоритмів, особливості архітектур сучасних мікропроцесорів.

Завдання на проект обираються студентом з переліку тем, наведеного в таблиці або тема дослідження пропонується самостійно студентом (узгоджується з викладачем). Студенти мають представити та захистити свій проект у встановлені терміни.

Кожен проект оцінюється на основі наступних критеріїв:

- Завершеність проекту(часткове або повне завершення).
- Опис або реалізація методів для вирішення завдання.
- Результати тестування.
- Чіткість та логічність представлення результатів.
- Відповідність оформлення проекту встановленим вимогам.

Перелік тем:

№ варіанту	Тема
1.	Реалізація програми на асемблері, що виконує базові арифметичні операції.
2.	Дослідження механізму системних викликів через асемблер.
3.	Реалізація програми, що читаютає та пише в файли за допомогою системних викликів.
4.	Дослідження особливостей програмування під 64-бітну архітектуру.
5.	Реалізація програми, що використовує розширену пам'ять та регістри
6.	Розробка змішаних програм, що включають модулі на С та асемблері.
7.	Аналіз викликів функцій та передачі даних між модулями
8.	Розробка механізмів обробки помилок та виняткових ситуацій
9.	Імплементація стекових фреймів для відновлення після помилок
10.	Аналіз та оптимізація швидкодії та використання пам'яті програм на асемблері.
11.	Розробка програми, що взаємодіє з апаратними компонентами системи через порти вводу/виводу
12.	Реалізація простої багатопотокової програми із використанням асемблерних інструкцій
13.	Імплементація простої програми шифрування на асемблері
14.	Реалізація основних стекових операцій для ручного управління пам'яттю.
15.	Створення та використання функцій на асемблері, з використанням конвенції викликів.
16.	Розробка обгорток для системних викликів
17.	Інтеграція С-бібліотеки у програмі на асемблері.
18.	Реалізація обробників сигналів та аналіз їх впливу на стабільність програм

№ варіанту	Тема
19.	Дослідження оптимізації алгоритмів для підвищення їх продуктивності на асемблері
20.	Моделювання роботи процесора за допомогою асемблера
21.	Реалізація програми яка використовує SIMD інструкції для оптимізації обчислень
22.	Реалізація програми для роботи з плаваючою крапкою на асемблері
23.	Аналіз та модифікація бінарних файлів для розуміння їхньої поведінки та структури
24.	Розробка makefiles для автоматизації збірки складних асемблерних проектів
25.	Розробка програми для мікроконтролерів або інших вбудованих систем на асемблері
26.	Дослідження особливості архітектур та застосування сигнальних процесорів
27.	FPGA матриці та особливості роботи з ними
28.	Синтез комбінаційних схем

7. Тренінг з дисципліни

Завданням тренінгу є вивчення концепцій процесів в ОС Linux та механізмів їх створення і керування, організації доступу до апаратних ресурсів через порти вводу/виводу та меморійні мапи, методи створення та інтеграції бібліотек на асемблері з програмами на мовах вищого рівня. Основна мета тренінгу — розробка програм для створення, завершення та синхронізації процесів з використанням системних викликів fork, exit та wait, та створення набору асемблерних функцій, компіляція їх у бібліотеку і використання цієї бібліотеки в програмах на C/C++.

Етап тренінгу	**Завдання**
Апаратні ресурси та організація доступу	-Створення програми для організації доступу до апаратних ресурсів через порти вводу/виводу.
Створення бібліотек і синхронізація процесів	-Створення асемблерних функцій, компіляція їх у бібліотеку, для використання в інших мовах програмування - Розробка програми синхронізації процесів з використанням системних викликів fork, exit та wait.
Презентація та захист проекту	Підготовка та презентація результатів.

8. Критерії, форми поточного та підсумкового контролю

Підсумковий бал (за 100-бальною шкалою) з дисципліни «Системне програмування та архітектура комп'ютерів» визначається як середньозважена величина, залежно від питомої ваги кожної складової залікового кредиту:

Модуль 1		Модуль 2		Модуль 3	Модуль 4	Модуль 5
10 %	10 %	10 %	10 %	5%	15%	40%
Поточне оцінювання	Модульний контроль 1	Поточне оцінювання	Модульний контроль 2	Тренінг	Самостійна робота	Екзамен
Оцінювання виконання лабораторних робіт (6 робіт)	25 тестових питань	Оцінювання виконання лабораторних робіт (6 робіт)	- Одне теоретичне питання - Одне практичне завдання	Виконання завдань тренінгу (3 завдання)	Виконання завдання для самостійної роботи	- 20 тестових питань - Одне теоретичне питання - Одне практичне завдання

Критерії поточного оцінювання

90–100 — впевнене володіння матеріалом; коректні програми на NASM і збірка з LD; відлагодження в GDB/DDD; безпомилкове застосування стеку/процедур/адресації; коректні схеми, повний звіт з лог- файлами.

75–89 — Повна реалізація програм, присутні неточності, наприклад маскування прапорів, помилки у викликах/адресації, демонстрація успішна; звіт містить незначні помилки.

65–74 — В реалізації присутні помилки, які викликають нестабільність роботи програми, помилки у вводі-виводі, прогалини в теорії; звіт неповний.

60–64 - Фрагментарні знання; програмна реалізація нестабільна, звіт поверхневий.

1–59 — відсутні ключові результати або відмова від захисту.

Критерії оцінювання модульного контролю 1

25 тестових питань:

4 бали — правильна відповідь;

0 балів — неправильна/відсутня відповідь.

Якщо завдання з багатьма правильними варіантами бал зараховується лише за повну відповідність ключу.

23–25 правильних -90–100 балів

19–22 правильних -75–89 балів

16–18 правильних - 65–74 бали

15 правильних - 60–64 бали

9–14 правильних -35–59 балів

0–8 правильних - 1–34 бали

Критерії оцінювання модульного контролю 2

А) Теоретичне питання — 40 балів.

36–40 — Повне і точне розкриття, наприклад модель фон Неймана/Гарвардська архітектура, пам’ять, адресація, формати команд, АЛП, мікропроцесори/GPU/TPU); коректні визначення, доречні схеми/приклади.

30–35 — Питання правильно та аргументовано висвітлено, але присутні незначні неточності.

24–29 — Питання висвітлено в недостатній мірі.

0–23 — Питання висвітлено фрагментарно та присутні посилки в термінології та поняттях.

Б) Практичне завдання — 60 б.

54–60 — Коректна реалізація, стабільна робота, обґрунтоване використання інструкцій/режимів, відлагодження в GDB або DDD, демонстрація та повний звіт 45–53 — Рішення працездатне з дрібними похибками.

36–44 — Базова функціональність реалізована частково, відслідковується нестабільність, помилки.

0–35 — Некоректна або неповна реалізація, відсутня демонстрація.

Критерії оцінювання виконання завдань тренінгу

90–100 — Повний функціонал, валідні приклади доступу до I/O, бібліотека з коректним інтерфейсом, процеси синхронізовані, чітка демонстрація та звіт.

75–89 — Функціонал реалізовано, але присутні дрібні неточності, демонстрація успішна, звіт з незначними прогалинами.

65–74 — Працездатність, але нестабільність та частковість функціонування, аналіз поверхневий, звіт неповний.

60–64 — мінімум: неповне демо/слабка інтеграція; звіт формальний.

1–59 — часткова або неправильна реалізація, демонстрації немає.

Критерії оцінювання завдань самостійної роботи

90–100 — Проєкт завершений, валідна методика, результати тестування повний та звіт з висновками представлено.

75–89 — Проєкт завершений з незначними спрощеннями, аналіз і звіт представлено.

60–74 —Функціонал проєкту мінімальний, представлено поверхневий аналіз та звіт.

1–59 — Малий обсяг, присутні критичні помилки, відсутня аналітика та демонстрація.

Критерії екзаменаційного оцінювання

Бал екзамену = Тести (0–20) + Теорія (0–40) + Практика (0–40) = 0–100 балів

А) Тести — 20 питань з однією/кількома правильними відповідями. Нарахування 1 бал — відповідь повністю правильна, Якщо завдання з багатьма правильними варіантами бал зараховується лише за повну відповідність ключу, 0 балів — неправильна/відсутня відповідь.

Тести: 18–20 правильних відповідей. Відмінно.

Тести: 14–17 правильних відповідей. Добре.

Тести: 11–13 правильних відповідей. Задовільно.

Тести: 1–10 правильних відповідей. Незадовільно

Б) Теоретичне питання — 40 балів.

36–40 балів — відмінно. Повне, точне розкриття питання, коректна термінологія, приклади.

30–35 балів — добре. В основному правильно аргументовані відповіді, незначні неточності.

24–29 балів — задовільно. Основні елементи розкрито, але є прогалини, відповіді поверхневі.

0–23 балів — незадовільно. Відповідь фрагментарна або з критичними помилками.

В) Практичне питання — 40 балів.

36–40 —Повна коректність, чиста архітектура, впевнена демонстрація з

аргументованими висновками.

30–35 — Рішення працездатне, дрібні неточності або неповний аналіз/оформлення.

24–29 — Базова функціональність реалізована, але помітні слабкі тести, недоліки у стек-рамці/FSM, часткова специфікація.

0–23 — Некоректна або неповна реалізація; відсутні ключові тести та демонстрація.

Шкала оцінювання:

За шкалою ЗУНУ	За національною шкалою	За шкалою ECTS
90-100	Відмінно	A (відмінно)
85-89	Добре	B (дуже добре)
75-84		C (добре)
65-74	Задовільно	D (задовільно)
60-64		E (достатньо)
35-59	Незадовільно	FX (незадовільно, з можливістю повторного складання)
1-34		F (незадовільно, з обов’язковим повторним курсом)

9. Інструменти, обладнання та програмне забезпечення, використання яких передбачає навчальна дисципліна

№	Найменування	Номер теми
1.	Мультимедійне обладнання	1-7
2.	Netwide Assembler (NASM) — асемблер і дизасемблер для архітектури x86/64	1-14
3.	Програма відлагоджувальник GDB	1-7
4.	Програма відлагоджувальник DDD	1-7
5.	Веб-ресурс для побудови комбінаційних схем www.tinkercad.com	8-14
6.	Веб-ресурс для моделювання логічних схем https://logic.ly/	8-14

РЕКОМЕНДОВАНІ ДЖЕРЕЛА ІНФОРМАЦІЇ

1. Ed, Jorgensen. "x86-64 assembly language programming with Ubuntu." (2019). URL: https://dlib.hust.edu.vn/bitstream/HUST/25064/1/OER000003243.pdf
2. Duntemann, Jeff. x64 Assembly Language Step-by-Step: Programming with Linux, 4th Edition. John Wiley & Sons, 2024.
3. Галісєєв Г. Системне програмування. К.: Університет "Україна". 2019. - 111с.
4. Noam Nisan, Shimon Schocken, The Elements of Computing Systems, second edition: Building a Modern Computer from First Principles 2nd Edition. 2021.

344P.
5. Jim Ledin, Modern Computer Architecture and Organization: Learn x86, ARM, and RISC-V architectures and the design of smartphones, PCs, and cloud servers, 2nd Edition, 2022.
6. Intel Corporation. Intel® 64 and IA-32 Architectures Software Developer's Manual Combined Volumes: 1, 2A, 2B, 2C, 2D, 3A, 3B, 3C, 3D, and 4. Submitted: May 01, 2018 Last updated: May 27, 2020. URL: https://software.intel.com/content/www/us/en/develop/download/intel-64-and-ia32-architectures-sdm-combined-volumes-1-2a-2b-2c-2d-3a-3b-3c-3d-and4.html.
7. Ed Jorgensen, Ph.D. x86-64 Assembly Language Programming with Ubuntu, Version 1.1.44, May 2022
8. TI-RSLK, Texas Instruments Robotics System Learning Kit. URL: https://www.ti.com/lit/ml/swrp141/swrp141.pdf?ts=1695418985444
9. Осолінський О.Р. Опорний конспект лекцій з дисципліни «Системне програмування та архітектура комп'ютерів» для студентів спеціальності 122 «Комп'ютерні науки». - Тернопіль: ЗУНУ, 2021. - 80 с.
10. Осолінський О.Р. Методичні вказівки до виконання лабораторних робіт з дисципліни «Системне програмування та архітектура комп'ютерів» для студентів спеціальності 122 «Комп'ютерні науки». - Тернопіль: ЗУНУ, 2021. - 40 с.
11. Orest Kochan, Oleksandr Osolinskyi, Anatoliy Sachenko, Volodymyr Kochan, Ihor Romanets, "Simulator of Microcontrollers Power Consumption," 2023 IEEE 12th International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS), Dortmund, Germany, 2023, pp. 787-792.
12. Osolinskyi, O., Lipianina-Honcharenko, K., Kochan, V., Sachenko, A., & Zahorodnia, D. (2023). Energy Consumption of Methods for Pattern Recognition using Microcontrollers. International Journal of Computing, 22(4), 502-508. https://doi.org/10.47839/ijc.22.4.3358
13. A. Kanovskyi, O. Osolinskyi, H. Lipyanina-Goncharenko and A. Sachenko, "A Concept of Microprocessor-Based Electrical Load Control System with Extended Functionality," 2021 IEEE 4th International Conference on Advanced Information and Communication Technologies (AICT), 2021, pp. 112-117, doi: 10.1109/AICT52120.2021.9628945.