

Міністерство освіти і науки України
Харківський національний університет імені В.Н. Каразіна
Кафедра електроніки та управляючих систем

“ЗАТВЕРДЖУЮ”

Проректор
з науково-педагогічної роботи



Антон ПАНТЕЛЕЙМОНОВ

« 2021 р.

РОБОЧА ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

Системне програмне забезпечення

рівень вищої освіти	перший (бакалаврський) рівень
галузь знань	12 Інформаційні технології
спеціальність	123 Комп'ютерна інженерія
освітня програма	Комп'ютерна інженерія
вид дисципліни	обов'язкова
факультет	комп'ютерних наук

Програму рекомендовано до затвердження Вченою радою факультету комп'ютерних наук

«30» червня 2021 року, протокол № 15

РОЗРОБНИКИ ПРОГРАМИ:

кандидат технічних наук, доцент кафедри електроніки та управляючих систем
Стервоєдов Микола Григорович

Програму схвалено на засіданні кафедри електроніки та управляючих систем

Протокол від «07» червня 2021 року № 10

Завідувач кафедри електроніки та управляючих систем


_____ Микола СТЕРВОЄДОВ

Програму погоджено з гарантом освітньої програми «Комп'ютерна інженерія»

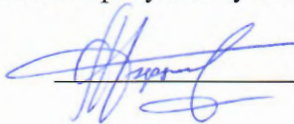
Гарант освітньої програми «Комп'ютерна інженерія»


_____ Вікторія СТРИЛЕЦЬ

Програму погоджено науково-методичною комісією факультету комп'ютерних наук

Протокол від «25» червня 2021 року № 9

Голова науково-методичної комісії факультету комп'ютерних наук


_____ Анатолій БЕРДНІКОВ

ВСТУП

Програма навчальної дисципліни «Системне програмне забезпечення» складена відповідно до освітньо-професійної програми підготовки першого рівня вищої освіти бакалавр напрямку 123 Комп'ютерна інженерія.

1. Опис навчальної дисципліни

1.1. Метою викладання курсу є вивчення студентами взаємозв'язку між архітектурою ЕОМ та системними програмами, надання студентам базових знань з системного адміністрування, проектування та розробки системного програмного забезпечення. Основною метою викладання дисципліни є розкриття сучасних наукових концепцій, понять, методів та технологій проектування та реалізації системного програмного забезпечення на підставі засвоєння алгоритмів, що покладені в основу операційних систем, алгоритмів системних служб, алгоритмів обробки інформації різних типів даних, вивчення принципів реалізації системного програмного забезпечення операційних середовищ та систем з використанням сучасних технологій програмування.

1.2. Основними завданнями вивчення дисципліни є надати базові знання у галузі проектування та реалізації системних програм та компонентів системного програмного забезпечення. Ця дисципліна розкриває не тільки методологію створення програмного забезпечення для комп'ютерних систем, а також розкриває основні труднощі, які виникають під час створення подібних програм, та шляхи їх вирішення. Коректне збудоване та правильно налагоджене системне програмне забезпечення дозволяє підвищити ефективність роботи як локальних так і мережевих комп'ютерних систем. Об'єктами діяльності фахівців з комп'ютерної інженерії, які вивчають цей курс, є автоматизація та комп'ютерно-інтегровані технології, технічне, програмне, математичне, інформаційне та організаційне забезпечення систем автоматизації для збору, передачі та обробки інформації, а також управління процесами і виробництвами в різних галузях науки, промисловості, теплоенергетики, електроенергетики та інших об'єктах автоматизації на різних рівнях управління ними та їх інтеграції в організаційно-технічні системи з використанням сучасної мікропроцесорної техніки, спеціалізованого прикладного програмного забезпечення та інформаційних технологій.

В ході вивчення дисципліни у студента повинні формуватися наступні компетентності.

Інтегральна компетентність.

Здатність розв'язувати складні задачі та вирішувати практичні завдання під час професійної діяльності в комп'ютерній галузі, що передбачає застосування теорій та методів інформаційних технологій і характеризуються комплексністю та невизначеністю умові вимог.

Спеціальні (фахові, предметні) компетентності (ФК)

- Здатність забезпечувати захист інформації, що обробляється в комп'ютерних та кіберфізичних системах та мережах з метою реалізації встановленої політики інформаційної безпеки (ФК 4);
- Здатність використовувати засоби і системи автоматизації проектування до розроблення компонентів комп'ютерних систем та мереж, Інтернет додатків, кіберфізичних систем тощо (ФК 5);
- Готовність брати участь у роботах з впровадження комп'ютерних систем та мереж, введення їх до експлуатації на об'єктах різного призначення (ФК 8);
- Здатність системно адмініструвати, використовувати, адаптувати та експлуатувати наявні інформаційні технології та системи (ФК 9).

1.3. Кількість кредитів – 4

1.4. Загальна кількість годин – 120

1.5. Характеристика навчальної дисципліни	
Нормативна / за вибором	
Денна форма навчання	Заочна (дистанційна) форма навчання
Рік підготовки	
3-й	3-й
Семестр	
5-й	
Лекції	
32 год.	
Практичні, семінарські заняття	
Лабораторні заняття	
32 год.	
Самостійна робота	
56 год.	
у тому числі індивідуальні завдання	
5 год.	

1.6. Заплановані результати навчання

Згідно з вимогами освітньо-професійної програми студенти мають досягти таких результатів навчання:

знати:

- методи та засоби проектування програмних засобів;
- компоненти системного програмного забезпечення (СПЗ);
- технологію розробки компонентів СПЗ;
- механізми керування системою;
- внутрішній устрій процесів;
- внутрішній устрій та планування потоків;
- засоби синхронізації й тупики;
- керування віртуальною пам'яттю;
- основні інструменти автоматизації роботи;
- роботу з файловою системою;
- архітектуру, характеристики, можливості і області застосування комп'ютерів, комп'ютерних систем, мікроконтролерів і програмованих логічних контролерів основних класів і типів;
- склад, принципи організації і функціонування окремих підсистем, комп'ютерів, мікроконтролерів і комп'ютерних систем в цілому;
- архітектуру інформаційно-управляючих, інформаційно-обчислювальних і проблемно-орієнтованих систем

вміти:

- ставити та вирішувати задачі, які виникають в процесі проектування та налагоджування системних програмних засобів;
- розрізняти основні функції ОС;
- керувати задачами;
- здійснювати планування та диспетчеризацію задач;
- керувати пам'яттю;
- керувати файлами;
- обробляти переривання; керувати процесами;
- керувати пристроями вводу – виведення;
- розрізняти і вміти використати сучасні ОС;
- розробляти математичні моделі елементів та вузлів комп'ютерних систем;

- тестувати й налагоджувати апаратно-програмні засоби і комплекси систем автоматизації та управління.

Самостійна робота передбачає вивчення окремих теоретичних питань, орієнтованих на обов'язкове використання обчислювальної техніки і максимально наближених до реальних інженерних задач майбутньої спеціальності.

Вивчений теоретичний матеріал з дисципліни повинен використовуватися і закріплюватися під час проведення лабораторних занять.

В результаті вивчення дисципліни у студента повинні формуватися наступні програмні результати навчання (ПРН).

- Знати і розуміти наукові і математичні положення, що лежать в основі функціонування комп'ютерних засобів, систем та мереж (ПРН 1);
- Мати навички проведення експериментів, збирання даних та моделювання в комп'ютерних системах (ПРН 2);
- Вміти застосовувати знання для ідентифікації, формулювання і розв'язування технічних задач спеціальності, використовуючи методи, що є найбільш придатними для досягнення поставлених цілей (ПРН 6);
- Вміти розв'язувати задачі аналізу та синтезу засобів, характерних для спеціальності (ПРН 7);
- Вміти системно мислити та застосовувати творчі здібності до формування нових ідей (ПРН 8);
- Вміти застосовувати знання технічних характеристик, конструктивних особливостей, призначення і правил експлуатації програмно-технічних засобів комп'ютерних систем та мереж для вирішення технічних задач спеціальності (ПРН 9);
- Вміти розробляти програмне забезпечення для вбудованих і розподілених застосувань, мобільних і гібридних (ПРН 10);
- Вміти ефективно працювати як індивідуально, так і у складі команди (ПРН 12);
- Вміти ідентифікувати, класифікувати та описувати роботу комп'ютерних систем та їх компонентів (ПРН 13)
- Вміти поєднувати теорію і практику, а також приймати рішення та виробляти стратегію діяльності для вирішення завдань спеціальності з урахуванням загальнолюдських цінностей, суспільних, державних та виробничих інтересів (ПРН 14);
- Спілкуватись усно та письмово з професійних питань українською мовою та однією з іноземних мов (англійською, німецькою, італійською, французькою, іспанською) (ПРН 17);
- Використовувати інформаційні технології та для ефективного спілкування на професійному та соціальному рівнях (ПРН 18);
- Усвідомлювати необхідність навчання впродовж усього життя з метою поглиблення набутих та здобуття нових фахових знань, удосконалення креативного мислення (ПРН 20);
- Якісно виконувати роботу та досягати поставленої мети з дотриманням вимог професійної етики (ПРН 21).

2. Тематичний план навчальної дисципліни

Розділ 1. Состав, структура й основні функції системного й прикладного програмного забезпечення. Інженерія СПЗ і ППЗ.

Тема 1. Мета та задачі курсу. Основні поняття та структура системного програмного забезпечення. Состав, структура й основні функції системного й прикладного програмного забезпечення. Поняття обчислювальної машини та інформаційної системи.

Тема 2. Програмні засоби та середовища. Види та класифікація програмного забезпечення. Системне ПЗ. Прикладне ПЗ. Інструментальне ПЗ. Мережеве ПЗ. Інтегровані програмні пакети. Бібліотеки стандартних програм. Операційні системи, файлові системи. Види та класифікація ОС. Файлові менеджери. Програмне забезпечення мобільних пристроїв.

Тема 3. Інженерія ПЗ. Процес розробки програмного забезпечення Поняття інженерії програмного забезпечення. Інструментальне програмне забезпечення. Інтегровані середовища розробки ПЗ. Проектування, моделювання та розробка ПЗ. Поняття процесу розробки ПЗ. Універсальний процес. Поточний процес. Конкретний процес. Стандартний процес. Удосконалення процесу. Класичні моделі процесу: Водопадна модель, спіральна модель.

Тема 4. Фази і види діяльності. Поняття архітектури ПЗ. Види вимог: функціональні вимоги, не функціональні вимоги. Властивості вимог: ясність і недвозначність, повнота і несуперечність, необхідний рівень деталізації. Формалізація вимог. Управління версіями. Стандартизація якості. Методи забезпечення якості ПЗ. Поняття тестування. Тестування чорного ящика. Тестування білого ящика. Інструменти тестування.

Розділ 2. Концепція dataflow програмування. Середовище розробки и платформа для виконання програм LabVIEW, віртуальні прилади.

Тема 5. Знайомство з елементами інтерфейсу середовища LabVIEW. Склад віртуальних приладів LabVIEW. Створення віртуальних приладів і проектів LabVIEW. Робота з передньою панеллю і блок-схемою LabVIEW. Виконання пошуку елементів управління, ВП і функцій. Основні прийоми побудови програми в LabVIEW. Побудова простого додатка LabVIEW для збору, аналізу і представлення даних. Формування розуміння моделі програмування на основі потоку даних. Визначення основних типів даних.

Тема 6. Модульне програмування та віртуальні підприбори (SubVI). Освоєння модульного програмування в LabVIEW і прийомів створення і налаштування іконки і сполучної панелі віртуального приладу для його використання в якості ВПП. Основи модульного програмування і документування коду. Пошук помилок та налагодження ВП. Навчання прийомам налагодження і перевірки на наявність помилок в LabVIEW. Ідентифікація загальних проблем організації блок-схеми і потоку даних в блок-схемі.

Тема 7. Цикли і алгоритмічні структури віртуальних приладів LabVIEW Використання циклів. Знайомство з різними шляхами організації багаторазового запуску коду LabVIEW і прийомами для управління виконанням циклів. Налаштування програмних тимчасових режимів коду. Використання даних в різних ітераціях циклів. Використання структур прийняття рішень. Вивчення структур LabVIEW, які використовуються для реалізації алгоритмів прийняття рішень в додатках. Створення і застосування структури варіанту і структури обробника подій. Типи і структури даних віртуальних приладів LabVIEW типи даних LabVIEW.

Тема 8. Різні способи представлення числових даних. Приведення даних. Робота з текстовими даними. Операції з рядками. Перетворення типів даних. Структури даних.

Знайомство з типами даних, які об'єднують дані в єдину структуру для спрощення доступу до даних і їх аналізу. Створення і використання елементів для роботи з масивами. Кластери. Інструменти для роботи з кластерами. Визначення типу. Використання визначень типів для підвищення ефективності повторного використання структур даних в додатках.

Тема 9. Робота з файлами 4.1 Доступ до файлів з LabVIEW. Вивчення базових концепцій файлового введення / виводу і основних шляхів доступу і модифікації ресурсів в LabVIEW. Високорівневі і низькорівневі функції файлового введення / виводу. Технології введення / виведення файлів. Вивчення різних форматів файлів для збору і зберігання даних і підходи до вибору підходящого формату для конкретних додатків. Практичне застосування модульного коду, який зчитує або записує виміряні дані. Порівняння файлових форматів.

Тема 10. Управління потоком даних і графічним інтерфейсом користувача. Використання змінних. Вивчення локальних змінних і можливостей їх використання для модифікації значень елементів управління передньої панелі, зупинки паралельних циклів і обходу обмежень потоку даних. Комунікація між паралельними циклами, запис в елементи управління і зчитування даних з індикаторів. Передача даних між паралельними циклами. Освоєння прийомів організації передачі даних між паралельними циклами. Створення коду програми, керованого через призначений для користувача інтерфейсом з синхронізації даних через паралельні цикли. Використання черг (queues) для передачі накопичених даних між циклами і повідомлень для одночасної передачі даних декількома циклами.

Тема 11. Управління призначеним для користувача інтерфейсом. Вивчення методів програмного керування атрибутами об'єктів передньої панелі. Тимчасове відключення елементів управління. Освоєння можливостей використання VI-сервера для доступу до властивостей і методів об'єктів передньої панелі.

Тема 12 Використання високорівневих структур і шаблонів для розробки ефективних програм. Використання програмування на основі послідовності і кінцевого автомата. Прийоми послідовного програмування. Застосування машини станів (кінцевого автомата). Застосування шаблонів проектування. Знайомство з основами шаблонів проектування і вивчення їх переваг і функціональних можливостей. Скорочення часу розробки за рахунок використання шаблонів. Дві основні категорії шаблонів: з одним циклом і з декількома циклами.

3. Структура навчальної дисципліни

Назви розділів і тем	Кількість годин					
	Денна форма					
	Усього	у тому числі				
л		п	лаб	інд	ср	
1	2	3	4	5	6	7
Розділ 1. Состав, структура й основні функції системного й прикладного програмного забезпечення. Інженерія СПЗ і ППЗ.						
Тема 1. Мета та задачі курсу.	15	3		6		6
Тема 2. Програмні засоби та середовища.	11	3		2		6
Тема 3. Інженерія ПЗ.	10	2		2		6
Тема 4. Фази і види діяльності.	10	2		2		6
Разом за розділом 1	46	10		12		24
Розділ 2. Концепція dataflow програмування. Середовище розробки и платформа для виконання програм LabVIEW, віртуальні прилади.						
Тема 5. Знайомство з елементами інтерфейсу середовища LabVIEW.	8	2		2		4
Тема 6. Модульне програмування та віртуальні підприбори (SubVI).	10	4		2		4
Тема 7. Цикли і алгоритмічні структури віртуальних приладів LabVIEW.	10	4		2		4
Тема 8. Різні способи представлення числових даних.	10	4		2		4
Тема 9. Робота з файлами 4.1	10	2		4		4
Тема 10. Управління потоком даних і графічним інтерфейсом користувача.	10	2		4		4
Тема 11. Управління призначеним для користувача інтерфейсом.	8	2		2		4
Тема 12 Використання високорівневих структур і шаблонів для розробки ефективних програм.	8	2		2		4
Разом за розділом 2	74	22		20		32
Усього годин	120	32		32		56

4. Теми практичних (семінарських, лабораторних) занять

№ з/п	Назва теми (комплексної лабораторної роботи)	Кількість годин
Л1	Состав, структура й основні функції системного й прикладного програмного забезпечення	4
Л2	Інтегровані середовища розробки ПЗ	4
Л3	Знайомство з середовищем графічного програмування автоматизації наукових досліджень LabVIEW	4
Л4	Створення підпрограм віртуальних приладів и графічне відображення даних	4
Л5	Вивчення устрою збору даних на прикладі DAQ плати AECS-Lab 810	4
Л6	Дослідження системи управління кліматичними умовами з дистанційним моніторингом і застосуванням хмарних сервісів. (На базі ESP32 і Labview)	4
Л7	Дослідження системи управління	4
Л8	Аналіз продуктивності і створення автономних додатків на базі віртуальних приладів і ПЛК.	4
	Разом	32

Виконуються реальні і віртуальні частини лабораторної роботи.

5. Завдання для самостійної роботи

№ з/п	Назва теми	Кількість годин
1	Підготовка до лабораторних робіт та їх оформлення	5
2	Повторення лекційного матеріалу	8
3	Програмування мікроконтролерів і ПЛК	8
4	Програмування в середовищі графічного програмування LabVIEW	8
5	Вивчення комп'ютерних систем моделювання електронних схем Multisim, PROTEUSVSM та моделювання схем елементів КІСУ.	8
6	Підготовка до контрольної роботи	5
7	Виконання домашніх завдань	14
	Разом	56

6. Індивідуальні завдання

За бажанням студенту надається індивідуальне завдання

7. Методи навчання

Як правило лекційні та практичні заняття проводяться аудиторне. В умовах дії карантину заняття проводяться відповідно до Наказу ректора Харківського національного університету імені В.Н. Каразіна (аудиторне або дистанційно за допомогою платформ Google Meet або Zoom).

На досягнення освітніх цілей спрямовані такі методи навчання студентів:

– *практичні* (використовують для пізнання дійсності, формування навичок і вмінь, поглиблення знань. Під час їх застосування використовуються такі прийоми: планування виконання завдання, постановка завдання, оперативне стимулювання, контроль і регулювання, аналіз результатів, визначення причин недоліків);

– *пояснювально-ілюстративний* (використовують для викладання й засвоєння нового навчального матеріалу, фактів, підходів, оцінок, висновків тощо);

– *репродуктивний* (для застосування студентами вивченого на основі зразка або правила, алгоритму, що відповідає інструкціям, правилам, в аналогічних до представленого зразка ситуаціях);

8. Методи контролю

Поточний контроль – *контрольна робота*, звіти з виконання лабораторних робіт. Підсумковий семестровий контроль - *Залік – письмово*.

9. Схема нарахування балів

Розподіл балів для підсумкового семестрового контролю при проведенні залікової роботи

Поточний контроль, самостійна робота, індивідуальні завдання			Залік	Сума
Оцінка за виконання та захист лабораторних робіт Л1 – Л8	Контрольна робота, передбачена навчальним планом	Разом		
40	20	60	40	100

КРИТЕРІЇ ОЦІНКИ УСПІШНОСТІ ТА РЕЗУЛЬТАТІВ НАВЧАННЯ

Умовою допуску до заліку є виконання і захист лабораторних робіт (5 балів за кожен зі 8 робіт, максимальна оцінка - 40 балів) і обов'язкове виконання контрольної роботи (до 20 балів за теми Т1 — Т6).

Кожна лабораторна робота оцінюється в 5 балів – до 3 балів оцінюються оформлення згідно вимогам, які затверджено на кафедрі, та до 2 балів за захист роботи. При захисті роботи задається 2 питання. За кожен правильну відповідь нараховується 1 бал. За неналежне оформлення результатів роботи віднімається 1 бал. Для допуску до екзамену студенту необхідно набрати не менше 20 балів.

Контрольна робота оцінюється наступним чином. Студент повинен відповісти на 4 питання одного з 3 варіантів контрольної роботи. За кожне питання нараховується до 5 балів.

А саме:

Оцінка «5 балів» виставляється за:

- знання і розуміння програмного матеріалу в повному обсязі;
- послідовний, логічний, обґрунтований, безпомилковий виклад матеріалу;
- вільне формування висновків та узагальнень;
- самостійне застосування знань в конкретних ситуаціях;
- правильне, охайне оформлення контрольної роботи.

Оцінка «4 бали» виставляється за:

- знання і розуміння програмного матеріалу в повному обсязі;
- послідовний, логічний, безпомилковий виклад матеріалу;
- формування висновків та узагальнень;
- допущення окремих несуттєвих помилок;
- коли відповідь в основному відповідає вимогам, що і відповідь на оцінку «відмінно», але студент допускає незначні помилки, які не впливають у цілому на загальне рішення задачі.

Оцінка «3 бали» виставляється за:

- знання і розуміння тільки основного матеріалу;
- спрощений і неповний виклад матеріалу;
- допущення окремих несуттєвих помилок;
- коли студент в основному виконав завдання, але не глибоко володіє матеріалом, його знання мають розрізнений характер, допускаються помилки, які можна легко виправити і не викликають поважних ускладнень.

Оцінка «2 бали» виставляється за:

- спрощений і непослідовний виклад матеріалу;
- допущення окремих суттєвих помилок;
- відсутність узагальнень і висновків;
- коли студент орієнтується, дає невірну відповідь, має слабкі теоретичні знання.

Оцінка «1 бал» виставляється за:

- поверхове знання і розуміння основного матеріалу;
- допущенням істотних помилок.

Оцінка «0 балів» виставляється за:

- присутність на контрольній роботі та деяке орієнтування в матеріалі.

Залікова робота складається з 3 питань, вичерпна відповідь на кожне з них зараховується як 12 балів – перше питання, 12 балів – друге питання і 16 - балів – третє питання, що дає в сумі максимальні 40 балів за іспит. Часткова відповідь на кожне питання знижує максимальну оцінку до меншої кількості балів пропорційно тому, яку частину від повної відповіді на це питання містить письмова робота студента.

Як альтернатива може бути що залік проводиться в письмовому форматі, в формі тестування. Тест включає 40 питань з вибором правильного варіанта. Під час виконання тесту не допускається використання будь-яких допоміжних матеріалів і електронних пристроїв. Залік проводиться в присутності викладача. На виконання тесту дається 60 хвилин. Оцінка за письмовий іспит (тест) виставляється пропорційне числу правильних відповідей в тесті. Підсумкова оцінка за тест виставляється за 40-ка бальною шкалою.

Шкала оцінювання

Сума балів за всі види навчальної діяльності протягом семестру	Оцінка	
	для чотирирівневої шкали оцінювання	для дворівневої шкали оцінювання
90 – 100	відмінно	зараховано
70-89	добре	
50-69	задовільно	
1-49	незадовільно	не зараховано

10. Рекомендована література

Основна література

1. Д.В.Кознов Введение в программную инженерию. Часть I. Изд-во Санкт-Петербургского ун-та, 2005 г., 43 с
2. Трэвис Д., Кринг Д. LabVIEW для всех. 4–издание, переработанное и дополненное. – М.: ДМК Пресс, 2011. – 904 с. 2. Блюм П. LabVIEW: стиль программирования / П. Блюм; Пер. с англ. под ред. П. Михеева. – М.: ДМК Пресс, 2008. – 399 с.
3. Таненбаум Эндрю Современные операционные системы / Эндрю Таненбаум. – Издательский дом «Питер», 2007. – 1040 с
4. Олифер В. Г. Сетевые операционные системы / В. Г. Олифер, Н. А. Олифер. – СПб.: Питер, 2004. – 554 с

Допоміжна література

1. Essick, J. Hands-on introduction to LabVIEW for scientists and engineers / J. Essick. – 3rd ed. – New York [etc.]: Oxford University Press, 2016. – 668 с.
2. В.Кознов Языки визуального моделирования: проектирование и визуализация программного обеспечения. Изд. СПбГУ, 2004. 170 с
3. Буч Г., Якобсон А., Рамбо Дж UML. Изд. 2-е, СПб.: Питер, 2006, 735 с
4. H.Mills The management of software engineering Part I: Principles of software engineering IBM System Journal, Vol. 38, No 2&3, 2009, pp. 289-295

11. Посилання на інформаційні ресурси в Інтернеті, відео-лекції, інше методичне забезпечення

1. Програмне забезпечення: National Instruments LabVIEW 2013
2. База знань National Instruments: [Електронний ресурс] // National Instruments Corporation, 2019. URL: <http://www.ni.com/kb/>
3. Учебные материалы National Instruments: [Електронний ресурс] // National Instruments Corporation, 2019. URL: <http://www.ni.com/tutorials/>