



ОСНОВИ ПРОГРАМУВАННЯ. КУРСОВА РОБОТА

Робоча програма навчальної дисципліни (Силабус)

Реквізити навчальної дисципліни

Рівень вищої освіти	<i>Перший (бакалаврський)</i>
Галузь знань	<i>F Інформаційні технології</i>
Спеціальність	<i>F2 Інженерія програмного забезпечення</i>
Освітня програма	<i>Інженерія програмного забезпечення інформаційних систем</i>
Статус дисципліни	<i>Нормативна</i>
Форма навчання	<i>очна (денна)</i>
Рік підготовки, семестр	<i>1 курс, весняний семестр</i>
Обсяг дисципліни	<i>1 кредит</i>
Семестровий контроль/ контрольні заходи	<i>захист курсової роботи (залік)</i>
Розклад занять	<i>Індивідуальні консультації</i>
Мова викладання	<i>Українська</i>
Інформація про керівника курсу / викладачів	<i>ст. викл., PhD Головченко Максим Миколайович, k.o.o.v.g.v.n.s@gmail.com</i>
Розміщення курсу	<i>https://classroom.google.com/c/ODM1MTc3MDEyNjk3?cjc=mojdaeem</i>

Програма навчальної дисципліни

1. Опис навчальної дисципліни, її мета, предмет вивчення та результати навчання

Метою дисципліни "Основи програмування. Курсова робота" є закріплення у студентів здатності розробляти програмне забезпечення для вирішення прикладних задач різного ступеня складності на мовах програмування C++ / C# з використанням методології об'єктно-орієнтованого програмування.

Курсова робота – це вид самостійної роботи з елементами дослідження, що виконується студентами протягом семестру з метою закріплення, поглиблення і узагальнення знань, здобутих за час вивчення дисципліни "Основи програмування", та їхнього застосування до комплексного вирішення конкретного фахового завдання. Її написання є обов'язковою складовою індивідуального навчального плану бакалавра спеціальності 121 «Інженерія програмного забезпечення» за освітньою програмою «Інженерія програмного забезпечення інформаційних систем».

Предмет даної навчальної дисципліни – парадигма об'єктно-орієнтованого програмування, об'єктно-орієнтовані мови програмування, інструментальні засоби створення програмного забезпечення на мовах C++ / C#.

Написання курсової роботи сприяє формуванню у студентів наступних компетентностей.

Загальні компетентності:

ЗК2. Здатність застосовувати знання у практичних ситуаціях.

ЗК6. Здатність до пошуку, оброблення та узагальнення інформації з різних джерел.

Фахові компетентності, формуванню яких сприяє даний кредитний модуль:

ФК1. Здатність ідентифікувати, класифікувати та формулювати вимоги до програмного забезпечення.

ФК2. Здатність брати участь у проектуванні програмного забезпечення, включаючи проведення моделювання (формальний опис) його структури, поведінки та процесів функціонування.

ФК3. Здатність розробляти архітектури, модулі та компоненти програмних систем.

ФК7. Володіння знаннями про інформаційні моделі даних, здатність створювати програмне забезпечення для зберігання, видобування та опрацювання даних.

ФК8. Здатність застосовувати фундаментальні і міждисциплінарні знання для успішного розв'язання завдань інженерії програмного забезпечення.

ФК10. Здатність накопичувати, обробляти та систематизувати професійні знання щодо створення і супроводження програмного забезпечення та визнання важливості навчання протягом всього життя.

ФК11. Здатність реалізовувати фази та ітерації життєвого циклу програмних систем та інформаційних технологій на основі відповідних моделей і підходів розробки програмного забезпечення.

ФК13. Здатність обґрунтовано обирати та освоювати інструментарій з розробки та супроводження програмного забезпечення.

ФК14. Здатність до алгоритмічного та логічного мислення.

Формування зазначених компетентностей забезпечує досягнення наступних програмних результатів навчання:

ПРН1. Аналізувати, цілеспрямовано шукати і вибирати необхідні для вирішення професійних завдань інформаційно-довідникові ресурси і знання з урахуванням сучасних досягнень науки і техніки.

ПРН3. Знати основні процеси, фази та ітерації життєвого циклу програмного забезпечення.

ПРН4. Знати і застосовувати професійні стандарти і інші нормативноправові документи в галузі інженерії програмного забезпечення.

ПРН6. Уміння вибирати та використовувати відповідну задачі методологію створення програмного забезпечення.

ПРН7. Знати і застосовувати на практиці фундаментальні концепції, парадигми і основні принципи функціонування мовних, інструментальних і обчислювальних засобів інженерії програмного забезпечення.

ПРН12. Застосовувати на практиці ефективні підходи щодо проектування програмного забезпечення.

ПРН13. Знати і застосовувати методи розробки алгоритмів, конструювання програмного забезпечення та структур даних і знань.

ПРН14. Застосовувати на практиці інструментальні програмні засоби доменного аналізу, проектування, тестування, візуалізації, вимірювань та документування програмного забезпечення.

ПРН23. Вміти документувати та презентувати результати розробки програмного забезпечення.

2. Пререквізити та постреквізити дисципліни (місце в структурно-логічній схемі навчання за відповідною освітньою програмою)

Для успішного засвоєння дисципліни необхідно мати базові знання з

- дисципліни «Комп'ютерна дискретна математика» (ЗО 01),
- дисципліни «Лінійна алгебра та аналітична геометрія» (ЗО 03),
- кредитного модуля «Основи програмування – 1. Базові конструкції» (ПО 02.1),

- дисципліни «Алгоритми і структури даних" (ПО 01).

Успішне оволодіння знаннями з дисципліни готує студентів до вивчення наступних дисциплін, таких як:

- «Компоненти програмної інженерії» (ПО 8),
- «Проектування алгоритмів» (ПО 12),
- «Системне програмне забезпечення» (ПО 13),
- «Програмування веб-застосунків» (ПО 14),
- «Технології паралельних обчислень» (ПО 20).

3. Зміст навчальної дисципліни

Основні етапи виконання курсової роботи:

Етап	Найменування та зміст	Результат	Години
1	<i>Підготовчий етап</i> Ознайомлення з темою, узгодження ТЗ.	Узгоджене ТЗ	2
2	<i>Етап виконання</i> Аналіз предметної області задачі, постановка задачі, розробка алгоритмічної складової задачі, об'єктно-орієнтоване проектування та програмування, тестування програмного забезпечення, інструкція користувача.	Працездатне ПЗ	23
3	<i>Заключний етап</i> Аналіз результатів роботи програмного забезпечення. Оформлення пояснювальної записки (згідно ДСТУ «Документація. Звіти у сфері науки і техніки. Структура і правила оформлення»).	Пояснювальна записка КР	4.5
4	<i>Захист курсової роботи.</i>		0.5

Структурні частини пояснювальної записки:

Курсова робота складається з пояснювальної записки, разом з супроводжувальною документацією, та не обов'язкового графічного матеріалу (креслень).

Структурні частини пояснювальної записки:

1. Титульний аркуш.
2. Лист-завдання на курсову роботу.
3. Календарний план-графік виконання роботи.
4. Анотація.
5. Зміст.
6. Вступ.
7. Основна частина:
 - постановка задачі;
 - аналіз предметної області;
 - розробка алгоритмічної складової задачі;
 - об'єктно-орієнтоване проектування та програмування;
 - тестування програмного забезпечення;

- інструкція користувача;
- аналіз результатів роботи програмного забезпечення.

8. Висновок.

9. Перелік посилань.

10. Додатки до пояснювальної записки (Додаток А Технічне завдання, Додаток Б Тексти програмного коду).

4. Навчальні матеріали та ресурси

Базова література

1. Офіційна нотація мови UML (OMG UML) Version 2.5.1 [Електронний ресурс]. – Режим доступу: <https://www.omg.org/spec/UML/> (дата звертання 25.03.2025)
2. Основи програмування. Курсова робота [Електронний ресурс] : навч. посіб. для студ. спеціальності 121 «Інженерія програмного забезпечення» / КПІ ім. Ігоря Сікорського; уклад.: І.П. Муха, І.І.Вітковська, М.М. Головченко. – Київ: КПІ ім. Ігоря Сікорського, 2022. – 52 с.
3. Andrew Troelsen, Phil Japikse «Pro C# 10 with .NET 6» – Apress, 2022. – 1680 p. [Електронний ресурс]. – Режим доступу: <https://dl.ebooksworld.ir/books/Pro.CSharp.10.with.NET.6.Andrew.Troelsen.Phil.Japikse.Apress.9781484278680.EBooksWorld.ir.pdf>

Допоміжна література

1. C++. Основи програмування. Теорія та практика: підручник / [О. Г. Трофименко, Ю. В. Прокоп, І. Г. Швайко, Л. М. Буката та ін.] ; за ред. О. Г. Трофименко. – Одеса : Фенікс, 2010. – 544 с.
2. Головна сторінка середовища MS Visual Studio [Електронний ресурс]. – Режим доступу: <http://visualstudio.com>
3. Головна сторінка MSDN [Електронний ресурс]. – Режим доступу: <http://msdn.microsoft.com>

Політика та контроль

5. Політика навчальної дисципліни (освітнього компонента)

Студент має виконувати КР протягом семестру, дотримуючись затвердженого календарного плану.

Усі етапи виконання КР студент має виконувати **самостійно**, дотримуючись Кодексу честі Національного технічного університету України «Київський політехнічний інститут імені Ігоря Сікорського» (kpi.ua/code). У випадку виявлення факту академічної недоброчесності робота не зараховується (ставиться 0 балів).

КР вважається виконаною, якщо студент надав текст пояснювальної записки КР (у вигляді файлу(-ів) і у паперовому вигляді) та захистив її перед комісією (показав працездатність програмного забезпечення, виконав практичне завдання).

Студенти мають право оскаржити результати поточного контролю ходу виконання курсової роботи, аргументовано пояснивши викладачу, з якою оцінкою не погоджуються відповідно до оціночного листа та/або зауважень.

Відвідування лекційних, лабораторних та практичних занять з КМ «Основи програмування. Частина 2. Методології програмування» та консультацій з КМ «Основи програмування. Курсова робота» є обов'язковою складовою для успішного виконання та захисту КР.

Пріоритет відповідей на практичних заняттях, консультаціях або захисті курсової роботи надається тим студентам, які першими виявили бажання відповідати. При відсутності бажаних відповідати, студенти викликаються відповідно до списку у академічній групі.

Захист курсових робіт для кожної академічної групи проводиться в окремий день, у рамках залікових тижнів.

Захист курсової роботи здійснюється студентом особисто з обов'язковим представленням документів, що підтверджують особистість студента. У випадку дистанційного навчання для захисту курсової роботи студент повинен мати персональний комп'ютер та мати можливість демонструвати повний робочий екран з результатами захисту КР, а також мати веб-камеру для демонстрації документів, що підтверджують особистість студента та демонстрацію робочого місця під час виконання практичного завдання у рамках захисту курсової роботи.

Захист курсової роботи відбувається без застосування студентами допоміжних засобів (мобільні телефони, планшети, додатковий екран тощо), а також без використання допоміжних середовищ (Chat GPT, Bard, AI плагіни до IDE тощо). Під час захисту дозволяється використовувати наперед підготовлену довідкову інформацію з офіційних джерел за згодою викладача (без прикладів коду).

6. Види контролю та рейтингова система оцінювання результатів навчання (PCO)

6.1 Поточний контроль

Поточний контроль виконання курсової роботи виконується шляхом перевірки розділів курсової роботи.

6.2 Календарний контроль

Календарний контроль студентів проводиться двічі на семестр, як моніторинг поточного стану виконання вимог силабусу. Умови позитивного календарного контролю:

- за результатами навчальної роботи на першому календарному контролі (8-й тиждень) студент отримує «атестований», якщо його поточний рейтинг не менше 50% від максимально можливої кількості балів, які студент міг отримати за перші 7 тижнів;
- за результатами навчальної роботи на другому календарному контролі (14-й тиждень) студент отримує «атестований», якщо його поточний рейтинг не менше 50% від максимально можливої кількості балів, які студент міг отримати за перші 13 тижнів.

6.3 Система рейтингових (вагових) балів та критерії оцінювання

Рейтингова оцінка з курсової роботи має дві складові:

- оцінка виконання курсової роботи;
- оцінка захисту курсової роботи.

Перша (стартова) складова характеризує роботу студента з курсового проектування та її результат – якість пояснювальної записки та розробленого програмного забезпечення; друга складова характеризує якість захисту студентом курсової роботи шляхом виконання практичного завдання.

Розмір шкали першої складової ($r1$) - 50 балів, а другої складової ($r2$) - 50 балів.

Стартова складова виконання курсової роботи ($r1$):

- аналіз предметної області та постановка задачі КР – до 5 балів;
- розробка алгоритмічної складової задачі КР – до 5 балів;
- об'єктно-орієнтоване проектування КР – до 5 балів;
- написання та оформлення програмного коду КР – до 5 балів;
- розробка інтерфейсу користувача ПЗ КР (меню, екранні форми, графіки тощо) та графічне представлення усіх необхідних результатів відповідно до ТЗ – до 10 балів;
- тестування програмного забезпечення КР – до 5 балів;
- написання інструкції користувача ПЗ КР та аналіз результатів – до 5 балів;
- якість оформлення пояснювальної записки з урахуванням виконання вимог нормативних документів – до 10 балів;

–несвоєчасність виконання курсової роботи – -1 бал за кожен день запізнення відповідно графіку (25.05.2025), але не більше -10 балів.

Складова захисту курсової роботи (r2):

–відповіді на 5 питань за змістом курсової роботи – до 50 балів.

Теоретичне завдання захисту курсової роботи включає в себе п'ять підзавдань, кожне з яких оцінюється до 10 балів. Для зарахування захисту курсової роботи студент повинен повністю правильно виконати щонайменше три з п'яти підзавдань (відповідно до постановки) і набрати за захист курсової роботи щонайменше 30 балів. Якщо студент не виконує повністю правильно три з п'яти підзавдань (відповідно до постановки), складова (r2) не зараховується (проставляється 0 балів).

Увага!! До захисту не допускаються роботи, у яких відсутнє програмне забезпечення з графічним інтерфейсом користувача, програмне забезпечення містить критичні помилки (некоректно розв'язує задачу КР, містить необроблені виключні ситуації тощо), відсутня пояснювальна записка КР.

Більш деталізовані критерії оцінювання можна побачити у методичних вказівках до кредитного модуля.

Сума балів двох складових переводиться до залікової оцінки згідно з таблицею:

<i>Кількість балів</i>	<i>Оцінка</i>
100-95	Відмінно
94-85	Дуже добре
84-75	Добре
74-65	Задовільно
64-60	Достатньо
Менше 60	Незадовільно

7. Додаткова інформація з дисципліни (освітнього компонента)

Студент може закрити другу складову (r2) – захист курсової роботи за рахунок трьох успішних відповідей (розв'язання трьох задач протягом семестру) на практичних заняттях з КМ «Основи програмування. Частина 2. Методології програмування». У цьому випадку за згодою студента, йому проставляється за другу складову (r2) 50 балів, без виконання завдань захисту.

Студенти, які мають одну чи дві успішні відповіді (розв'язання однієї або двох задач протягом семестру) на практичних заняттях з КМ «Основи програмування. Частина 2. Методології програмування», отримують додаткові 10 та 20 балів відповідно до складової (r2) – захист курсової роботи, але не більше 50 балів у сумі за складову. Додаткові бали нараховуються виключно за згодою студента. Додаткові 10 чи 20 балів не можуть бути нараховані у випадку не зарахування захисту курсової роботи.

Робочу програму навчальної дисципліни (силабус):

Складено ст. вик., PhD кафедри ІІІ М. Головченком

Ухвалено кафедрою ІІІ (протокол № 2/1_ від _10.10.2025)

Погоджено Методичною комісією факультету (протокол № 3 від 17.10.2025)