

РОБОЧА ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

СИЛАБУС НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

1. Загальна інформація про навчальну дисципліну

Повна назва навчальної дисципліни	Основи програмної інженерії
Повна офіційна назва закладу вищої освіти	Сумський державний університет
Повна назва структурного підрозділу	Навчально-науковий інститут бізнесу, економіки та менеджменту. Кафедра економічної кібернетики
Розробник(и)	Кушнерьов Олександр Сергійович
Рівень вищої освіти	Перший рівень вищої освіти, НРК – 6 рівень, QF-LLL – 6 рівень, FQ-EHEA – перший цикл
Тривалість вивчення навчальної дисципліни	один семестр
Обсяг навчальної дисципліни	Обсяг становить 5 кред. ЄКТС, 150 год. Для денної форми навчання 46 год. становить контактна робота з викладачем (18 год. лекцій, 28 год. лабораторних занять), 104 год. становить самостійна робота. Для заочної форми навчання 10 год. становить контактна робота з викладачем (4 год. лекцій, 6 год. лабораторних занять), 140 год. становить самостійна робота.
Мова викладання	Українська

2. Місце навчальної дисципліни в освітній програмі

Статус дисципліни	Обов'язкова навчальна дисципліна для освітньої програми "Кіберспорт та розробка комп'ютерних ігор"
Передумови для вивчення дисципліни	Передумови для вивчення відсутні
Додаткові умови	Додаткові умови відсутні
Обмеження	Обмеження відсутні

3. Мета навчальної дисципліни

Процеси розробки великих та середніх програмних систем, сучасні гнучкі методології та практики у розробці програмного забезпечення

4. Зміст навчальної дисципліни

<p>Тема 1 Введення у основи програмної</p> <p>Загальне визначення дисциплін програмної інженерії. Характеристика областей знань з інженерії програмного забезпечення - SWEBOOK. Інженерія вимог. Проектування програмного забезпечення. Конструювання програмного забезпечення.</p>
<p>Тема 2 Поняття життєвого циклу</p> <p>Життєвий цикл ПЗ. Моделі життєвого циклу. Методології розробки ПЗ. Етапів життєвого циклу.</p>
<p>Тема 3 Моделювання та проектування програмного забезпечення</p> <p>Основні підходи та концепції моделювання. Введення в UML. Концептуальна модель UML. Діаграми UML. Правила UML.</p>
<p>Тема 4 Основи програмування на Java. Структура програми. Змінні та константи</p> <p>Для зберігання даних у програмі призначені змінні. Змінна являє собою іменовану область пам'яті, яка зберігає значення певного типу. Кожна змінна має тип, ім'я та значення. Тип визначає, яку інформацію може зберігати змінна чи діапазон допустимих значень.</p>
<p>Тема 5 Основи програмування на Java. Типи даних. Арифметические операции</p> <p>Однією з основних особливостей Java є те, що ця мова є строго типізованою. І це означає, кожна змінна і константа представляє певний тип і цей тип суворо визначений. Тип даних визначає діапазон значень, які може зберігати змінна або константа.</p>
<p>Тема 6 Основи програмування на Java. Умовні конструкції. Цикли. Масиви</p> <p>Умовні конструкції є деякою умовою і повертають значення типу boolean, тобто значення true (якщо умова істинно), або значення false (якщо умова хибна). До умовних виразів належать операції порівняння та логічні операції. Ще одним видом керуючих конструкцій є цикли. Цикли дозволяють залежно від певних умов виконувати певну дію багато разів. Масив представляє набір однотипних значень. Оголошення масиву схоже на оголошення звичайної змінної, яка зберігає одиночне значення.</p>
<p>Тема 7 Основи програмування на Java. Методи. Основи ООП. Класи та об'єкти. Модифікатори доступу та інкапсуляція</p> <p>Якщо змінні та константи зберігають деякі значення, то методи містять набір операторів, які виконують певні дії. Java є об'єктно-орієнтованою мовою, тому такі поняття як "клас" та "об'єкт" відіграють у ньому ключову роль. Будь-яку програму на Java можна як набір взаємодіючих між собою об'єктів. Усі члени класу у мові Java – поля та методи – мають модифікатори доступу. У минулих темах ми вже стикалися з модифікатором public. Модифікатори доступу дозволяють задати допустиму область видимості для членів класу, тобто контекст, у якому можна використовувати цю змінну чи спосіб.</p>

5. Очікувані результати навчання навчальної дисципліни

Після успішного вивчення навчальної дисципліни здобувач вищої освіти зможе:

РН1	знати базові технології Java
РН2	знати синтаксис мови програмування Java
РН3	використовувати технології Java

PH4	знати загальні підходи до моделювання ПЗ
PH5	знати основи інженерії вимог до програмного забезпечення

6. Роль навчальної дисципліни у досягненні програмних результатів

Програмні результати навчання, досягнення яких забезпечує навчальна дисципліна.
Для спеціальності 121 Інженерія програмного забезпечення:

ПР3	Знати основні процеси, фази та ітерації життєвого програмного забезпечення
ПР12	Застосовувати на практиці ефективні підходи щодо проєктування програмного забезпечення
ПР13	Знати і застосовувати методи розробки алгоритмів, конструювання програмного забезпечення та структур даних і знань.
ПР15	Мотивовано обирати мови програмування та технології розробки для розв'язання завдань створення і супроводження програмного забезпечення.
ПР16	Мати навички командної розробки, погодження, оформлення і випуску всіх видів програмної документації.

7. Роль навчальної дисципліни у досягненні програмних компетентностей

Програмні компетентності, формування яких забезпечує навчальна дисципліна:
Для спеціальності 121 Інженерія програмного забезпечення:

ПК1	ЗК01. Здатність до абстрактного мислення, аналізу та синтезу
ПК2	ЗК06. Здатність до пошуку, оброблення та аналізу інформації з різних джерел
ПК3	ЗК07. Здатність працювати в команді
ПК4	ФК1. Здатність ідентифікувати, класифікувати та формулювати вимоги до програмного забезпечення.
ПК5	ФК2. Здатність брати участь у проєктуванні програмного забезпечення, включаючи проведення моделювання (формальний опис) його структури, поведінки та процесів функціонування.
ПК6	ФК5. Здатність дотримуватися специфікацій, стандартів, правил і рекомендацій в професійній галузі при реалізації процесів життєвого циклу.
ПК7	ФК10. Здатність накопичувати, обробляти та систематизувати професійні знання щодо створення і супроводження програмного забезпечення та визнання важливості навчання протягом всього життя.
ПК8	ФК11. Здатність реалізовувати фази та ітерації життєвого циклу програмних систем та інформаційних технологій на основі відповідних моделей і підходів розробки програмного забезпечення.
ПК9	ФК13. Здатність обґрунтовано обирати та освоювати інструментарій з розробки та супроводження програмного забезпечення

8. Види навчальних занять

<p>Тема 1. Введення у основи програмної</p>
<p>Лк1 "Введення у основи програмної" (денна) Загальне визначення дисциплін програмної інженерії. Характеристика областей знань з інженерії програмного забезпечення - SWEBOOK. Інженерія вимог. Проектування програмного забезпечення. Конструювання програмного забезпечення.</p>
<p>Лб1 "Розробка технічного завдання" (денна) Технічне завдання – це основний документ всього проекту і опис взаємовідносин між замовником і виконавцем, який дозволяє чітко визначити порядок робіт, обов'язки сторін та терміни виконання проекту.</p>
<p>Лб2 "Розробка технічного завдання (продовження)" (денна) Технічне завдання – це основний документ всього проекту і опис взаємовідносин між замовником і виконавцем, який дозволяє чітко визначити порядок робіт, обов'язки сторін та терміни виконання проекту.</p>
<p>Тема 2. Поняття життєвого циклу</p>
<p>Лк2 "Поняття життєвого циклу" (денна) Життєвий цикл ПЗ. Моделі життєвого циклу. Методології розробки ПЗ. Етапів життєвого циклу.</p>
<p>Лб3 "Розробка та специфікація вимог до програмної системи" (денна) Важливою технічною компетенцією фахівців з програмної інженерії є навички з розробки та аналізу вимог до програмного забезпечення, документарне оформлення зазначених вимог, досвід управління вимогами протягом усього циклу створення програмного забезпечення</p>
<p>Тема 3. Моделювання та проектування програмного забезпечення</p>
<p>Лк3 "Моделювання та проектування програмного забезпечення" (денна) Основні підходи та концепції моделювання. Введення в UML. Концептуальна модель UML. Діаграми UML. Правила UML.</p>
<p>Лб4 "Розробка та специфікація вимог до програмної системи (продовження)" (денна) Важливою технічною компетенцією фахівців з програмної інженерії є навички з розробки та аналізу вимог до програмного забезпечення, документарне оформлення зазначених вимог, досвід управління вимогами протягом усього циклу створення програмного забезпечення</p>
<p>Тема 4. Основи програмування на Java. Структура програми. Змінні та константи</p>
<p>Лк4 "Основи програмування на Java. Структура програми. Змінні та константи" (денна) Для зберігання даних у програмі призначені змінні. Змінна являє собою іменовану область пам'яті, яка зберігає значення певного типу. Кожна змінна має тип, ім'я та значення. Тип визначає, яку інформацію може зберігати змінна чи діапазон допустимих значень.</p>

Лб5 "Створення діаграми класів" (денна)

Class diagram (діаграма класів) основна діаграма для створення коду додатка. За допомогою діаграми класів створюється внутрішня структура системи, описується спадкування й взаємне положення класів друг щодо друга. Тут описується логічне представлення системи. Саме логічне, тому що класи - це лише заготовки, на основі яких потім будуть визначені фізичні об'єкти.

Тема 5. Основи програмування на Java. Типи даних. Арифметические операции

Лк5 "Основи програмування на Java. Типи даних. Арифметические операции" (денна)

Однією з основних особливостей Java є те, що ця мова є строго типізованою. І це означає, кожна змінна і константа представляє певний тип і цей тип суворо визначений. Тип даних визначає діапазон значень, які може зберігати змінна або константа.

Лб6 "Створення діаграми класів (продовження)" (денна)

Class diagram (діаграма класів) основна діаграма для створення коду додатка. За допомогою діаграми класів створюється внутрішня структура системи, описується спадкування й взаємне положення класів друг щодо друга. Тут описується логічне представлення системи. Саме логічне, тому що класи - це лише заготовки, на основі яких потім будуть визначені фізичні об'єкти.

Лб7 "Середовище розробки (IDE) NetBeans. Структура проекту" (денна)

Створимо за допомогою середовища NetBeans додаток Java. Для цього запустимо інтегроване середовище розробки (IDE) NetBeans.

Тема 6. Основи програмування на Java. Умовні конструкції. Цикли. Масиви

Лк6 "Основи програмування на Java. Умовні конструкції" (денна)

Умовні конструкції є деякою умовою і повертають значення типу boolean, тобто значення true (якщо умова істинно), або значення false (якщо умова хибна). До умовних виразів належать операції порівняння та логічні операції.

Лк7 "Основи програмування на Java. Цикли. Масиви" (денна)

Ще одним видом керуючих конструкцій є цикли. Цикли дозволяють залежно від певних умов виконувати певну дію багато разів. Масив представляє набір однотипних значень. Оголошення масиву схоже на оголошення звичайної змінної, яка зберігає одиночне значення

Лб8 "Базові типи даних. Класи оболонки" (денна)

Примітивні типи даних можна використовувати безпосередньо або для створення власних типів класів. Примітивні типи служать основою для всіх інших типів даних.

Лб9 "Цикли" (денна)

Програма, написана мовою Java, складається з певного коду. Зазвичай він виконується послідовно: рядок за рядком, зверху донизу. Але є такі конструкції коду, які змінюють лінійне виконання програми. Їх називають керуючими конструкціями.

<p>Лб10 "Масиви" (денна)</p> <p>Масив – це структура даних, у якій зберігаються елементи одного типу. Його можна уявити, як набір пронумерованих осередків, у кожному з яких можна помістити якісь дані (один елемент даних в одну комірку). Доступ до конкретного осередку здійснюється через її номер. Номер елемента у масиві також називають індексом.</p>
<p>Тема 7. Основи програмування на Java. Методи. Основи ООП. Класи та об'єкти. Модифікатори доступу та інкапсуляція</p>
<p>Лк8 "Основи програмування на Java. Методи. Основи ООП. Класи та об'єкти. Модифікатори доступу та інкапсуляція" (денна)</p> <p>Якщо змінні та константи зберігають деякі значення, то методи містять набір операторів, які виконують певні дії. Java є об'єктно-орієнтованою мовою, тому такі поняття як "клас" та "об'єкт" відіграють у ньому ключову роль. Будь-яку програму на Java можна як набір взаємодіючих між собою об'єктів.</p>
<p>Лк9 "Модифікатори доступу та інкапсуляція" (денна)</p> <p>Усі члени класу у мові Java – поля та методи – мають модифікатори доступу. У минулих темах ми вже стикалися з модифікатором <code>public</code>. Модифікатори доступу дозволяють задати допустиму область видимості для членів класу, тобто контекст, у якому можна використовувати цю змінну чи спосіб.</p>
<p>Лб11 "Методи" (денна)</p> <p>Методи в Java — це закінчена послідовність дій (інструкцій), спрямованих на вирішення окремого завдання. По суті це добре знайомі функції (вони ж процедури, підпрограми) більш ранніх, не ООП мов.</p>
<p>Лб12 "Класи та відносини" (денна)</p> <p>Класи визначають структуру та поведінку деякого набору елементів предметної галузі, для якої розробляється програмна модель.</p>
<p>Лб13 "Абстрактні класи та методи" (денна)</p> <p>Абстрактні класи оголошуються з ключовим словом <code>abstract</code> і містять оголошення абстрактних методів, які не реалізовані в цих класах, а будуть реалізовані в підкласах. Об'єкти таких класів створити не можна, але можна створити об'єкти підкласів, які реалізують ці методи. Абстрактні класи можуть містити і повністю реалізовані методи.</p>
<p>Лб14 "Інтерфеи" (денна)</p> <p>Інтерфейси являють собою повністю абстрактні класи: жоден з оголошених методів не може бути реалізований. Всі оголошені методи автоматично трактуються як <code>public</code> і <code>abstract</code>, а всі атрибути – як <code>public</code>, <code>static</code> і <code>final</code>. Кожен інтерфейс може бути реалізований одним або декількома класами.</p>

9. Стратегія викладання та навчання

9.1 Методи викладання та навчання

Дисципліна передбачає навчання через:

МН1	інтерактивні, тематичні, оглядові лекції;
МН2	лабораторні заняття;

Лекції надають студентам матеріали з основних підходів програмування засобами Java, що є основою для самостійного навчання здобувачів вищої освіти (РН 1, РН 2, РН 3). Лекції доповнюються лабораторними заняттями, що надають студентам можливість застосовувати теоретичні знання на практичних прикладах (РН 4, РН5). Зміст лабораторних робіт напрямлений на практико-орієнтоване навчання, що передбачає визначення студентами доцільності застосування відповідних технологій створення додатків з метою оптимізації їх діяльності (РН 2, РН 3, РН 4, РН 5).

9.2 Види навчальної діяльності

НД1	Самостійна виконання та підготовка до захисту лабораторних робіт
НД2	Захист лабораторних робіт
НД3	Підготовка та виконання індивідуального завдання

10. Методи та критерії оцінювання

10.1. Критерії оцінювання

Визначення	Чотирибальна національна шкала оцінювання	Рейтингова бальна шкала оцінювання
Відмінне виконання лише з незначною кількістю помилок	5 (відмінно)	$90 \leq RD \leq 100$
Вище середнього рівня з кількома помилками	4 (добре)	$82 \leq RD < 89$
Загалом правильна робота з певною кількістю помилок	4 (добре)	$74 \leq RD < 81$
Непогано, але зі значною кількістю недоліків	3 (задовільно)	$64 \leq RD < 73$
Виконання задовольняє мінімальним критеріям	3 (задовільно)	$60 \leq RD < 63$
Можливе повторне складання	2 (незадовільно)	$21 \leq RD < 59$
Можливе одноразове повторне складання	2 (незадовільно)	$0 \leq RD < 20$

10.2 Методи поточного формативного оцінювання

	Характеристика	Дедлайн, тижні	Зворотний зв'язок
МФО1 Виконання лабораторних робіт	Виконання роботи відбувається студентом самостійно після отримання завдання від лектора	1 - 16	Viber, mix.sumdu.edu.ua
МФО2 Захист лабораторних робіт	Захист роботи відбувається у форматі "презентації" та усних (письмових) відповідей на контрольні питання до теми, "на запитання лектора"	1 - 16	Viber, mix.sumdu.edu.ua

МФО3 Виконання та захист індивідуального завдання	Захист індивідуального завдання відбувається у форматі "презентації" та усних (письмових) відповідей на контрольні питання до теми, "на запитання лектора"	15	Viber, mix.sumdu.edu.ua
МФО4 Складання комплексного письмового модульного контролю	Складання комплексного контролю відбувається у форматі письмових відповідей	16	Viber, mix.sumdu.edu.ua

10.3 Методи підсумкового сумативного оцінювання

	Характеристика	Дедлайн, тижні	Зворотний зв'язок
МСО1 Виконання лабораторних робіт	Виконання роботи відбувається студентом самостійно після отримання завдання від лектора	1 - 16	Viber, mix.sumdu.edu.ua
МСО2 Звіт за результатами виконання лабораторних робіт	Захист роботи відбувається у форматі "презентації" та усних (письмових) відповідей на контрольні питання до теми, "на запитання лектора"	1 - 16	Viber, mix.sumdu.edu.ua
МСО3 Захист індивідуального завдання	Захист індивідуального завдання відбувається у форматі "презентації" та усних (письмових) відповідей на контрольні питання до теми, "на запитання лектора"	15	Viber, mix.sumdu.edu.ua
МСО4 Складання комплексного письмового модульного контролю	Складання комплексного контролю відбувається у форматі письмових відповідей	16	Viber, mix.sumdu.edu.ua

Контрольні заходи:

		Максимальна кількість балів	Можливість перескладання з метою підвищення оцінки
Перший семестр вивчення		100 балів	
МСО1. Виконання лабораторних робіт		56	
	14x4	56	Так
МСО2. Звіт за результатами виконання лабораторних робіт		28	

	14x2	28	Так
МСО3. Захист індивідуального завдання		6	
		6	Так
МСО4. Складання комплексного письмового модульного контролю		10	
		10	Так

Дисципліна передбачає такі методи узагальнюючої підсумкової оцінки, як захист лабораторних робіт, перевірка та оцінка індивідуальної роботи. Форма підсумкового контролю - диференційований залік. Загальну позитивну оцінку дисципліни можна отримати, якщо за завдання набрано щонайменше 60% балів.

11. Ресурсне забезпечення навчальної дисципліни

11.1 Засоби навчання

ЗН1	Windows 10
ЗН2	AWS SDK для Java

11.2 Інформаційне та навчально-методичне забезпечення

Основна література	
1	Васильєв, О. М. Програмування мовою Java [Текст] / О. М. Васильєв. — Тернопіль : Навчальна книга - Богдан, 2022. — 696 с.
2	Кадомський, К. К. Java. Теорія і практика [Електронний ресурс] : навч. посіб. / К. К. Кадомський, П. К. Ніколюк. — Вінниця : Донецький нац. ун-т ім. Василя Стуса, 2019. — 197 с.
Допоміжна література	
1	Р.А. Мельник. Програмування веб- застосувань (фронт-енд та бек-енд). Навчальний посібник. - Львів : Видавництво Львівської політехніки, - 2018. - 248 с.
2	Олецький О. В. Перші кроки в Java [електронний ресурс] : навчальний посібник для студентів, які навчаються за спеціальностями "Інформатика", "Програмна інженерія", "Комп'ютерні науки", "Прикладна математика" / О. В. Олецький - Київ : [б. в.], 2017. - 144 с.
3	Прикладне програмування у комп'ютерних мережах : навчальний посібник / О. Д. Азаров, О. І. Черняк., Л. А. Савицька – Вінниця : ВНТУ, 2016. – 130 с.
Інформаційні ресурси в Інтернеті	
1	CodeRoad. https://coderoad.io/
2	Oracle Java Tutorials . https://docs.oracle.com/javase/tutorial/
3	Overview (Java SE 20 & JDK 20). https://docs.oracle.com/en/java/javase/20/docs/api/index.html

ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

№ з/п	Програма навчальної дисципліни	Усього годин	Навчальна робота, аудиторних годин				Самостійна робота здобувача вищої освіти за видами, годин					
			Усього, ауд. год.	Лекції	Практичні заняття	Лабораторні роботи	Усього, год.	Самостійне опрацювання матеріалу	Підготовка до практичних занять	Підготовка до лабораторних робіт	Підготовка до контрольних заходів	Виконання самостійних позааудиторних завдань
1	2	3	4	5	6	7	8	9	10	11	12	13
денна форма навчання												
1	Введення у основи програмної	8.5	6	2	0	4	2.5	0.5	0	2	0	0
2	Поняття життєвого циклу	5.5	4	2	0	2	1.5	0.5	0	1	0	0
3	Моделювання та проектування програмного забезпечення	5.5	4	2	0	2	1.5	0.5	0	1	0	0
4	Основи програмування на Java. Структура програми. Змінні та константи	5.5	4	2	0	2	1.5	0.5	0	1	0	0
5	Основи програмування на Java. Типи даних. Арифметические операции	8.5	6	2	0	4	2.5	0.5	0	2	0	0
6	Основи програмування на Java. Умовні конструкції. Цикли. Масиви	14	10	4	0	6	4	1	0	3	0	0
7	Основи програмування на Java. Методи. Основи ООП. Класи та об'єкти. Модифікатори доступу та інкапсуляція	17	12	4	0	8	5	1	0	4	0	0
Контрольні заходи												
1	диференційний залік	6	0	0	0	0	6	0	0	0	6	0
Індивідуальні завдання												
1	інші індивідуальні завдання	79.5	0	0	0	0	79.5	0	0	0	0	79.5
<i>Всього з навчальної дисципліни (денна форма навчання)</i>		<i>150</i>	<i>46</i>	<i>18</i>	<i>0</i>	<i>28</i>	<i>104</i>	<i>4.5</i>	<i>0</i>	<i>14</i>	<i>6</i>	<i>79.5</i>

№ з/п	Програма навчальної дисципліни	Усього годин	Навчальна робота, аудиторних годин				Самостійна робота здобувача вищої освіти за видами, годин					
			Усього, ауд. год.	Лекції	Практичні заняття	Лабораторні роботи	Усього, год.	Самостійне опрацювання матеріалу	Підготовка до практичних занять	Підготовка до лабораторних робіт	Підготовка до контрольних заходів	Виконання самостійних позааудиторних завдань
1	2	3	4	5	6	7	8	9	10	11	12	13
заочна форма навчання												
1	Введення у основи програмної	8.5	0	0	0	0	8.5	8.5	0	0	0	0
2	Поняття життєвого циклу	5.5	0	0	0	0	5.5	5.5	0	0	0	0
3	Моделювання та проектування програмного забезпечення	5.5	0	0	0	0	5.5	5.5	0	0	0	0
4	Основи програмування на Java. Структура программы. Змінні та константи	5.5	0	0	0	0	5.5	5.5	0	0	0	0
5	Основи програмування на Java. Типи даних. Арифметические операции	8.5	0	0	0	0	8.5	8.5	0	0	0	0
6	Основи програмування на Java. Умовні конструкції. Цикли. Масиви	14	0	0	0	0	14	14	0	0	0	0
7	Основи програмування на Java. Методи. Основи ООП. Класи та об'єкти. Модифікатори доступу та інкапсуляція	17	0	0	0	0	17	17	0	0	0	0
Контрольні заходи												
1	диференційний залік	6	0	0	0	0	6	0	0	0	6	0
Індивідуальні завдання												
1	інші індивідуальні завдання	79.5	0	0	0	0	79.5	0	0	0	0	79.5
<i>Всього з навчальної дисципліни (заочна форма навчання)</i>		<i>150</i>	<i>0</i>	<i>0</i>	<i>0</i>	<i>0</i>	<i>150</i>	<i>64.5</i>	<i>0</i>	<i>0</i>	<i>6</i>	<i>79.5</i>