

**ДЕРЖАВНА СЛУЖБА УКРАЇНИ З НАДЗВИЧАЙНИХ СИТУАЦІЙ
ЛЬВІВСЬКИЙ ДЕРЖАВНИЙ УНІВЕРСИТЕТ БЕЗПЕКИ
ЖИТТЄДІЯЛЬНОСТІ**

Навчально-науковий інститут цивільного захисту

Кафедра інформаційних технологій та систем електронних комунікацій

**РОБОЧА ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ (СИЛАБУС)
«Основи програмування»**

Рівень вищої освіти: перший (бакалаврський)

Галузь знань: F Інформаційні технології

Спеціальність (спеціалізація) : F3 Комп'ютерні науки

Спеціалізація (за наявності): _____

Освітня програма: Комп'ютерні науки

РОЗРОБНИК

Професор кафедри інформаційних технологій та систем електронних комунікацій
кандидат технічних наук, доцент

_____ Ольга СМОТР

«____» _____

ЗАТВЕРДЖЕНО

Начальник кафедри інформаційних технологій та систем електронних комунікацій
кандидат технічних наук, доцент

_____ Назарій БУРАК

«____» _____

ПОГОДЖЕНО

Гарант освітньої програми
«Комп'ютерні науки» першого
(бакалаврського) рівня вищої освіти
кандидат технічних наук, доцент

_____ Ольга СМОТР

«____» _____

ПОГОДЖЕНО

Заступник начальника навчально-наукового інституту цивільного захисту
кандидат фізико-математичних наук, доцент

_____ Ольга МЕНЬШИКОВА

«____» _____

Розглянуто та затверджено на засіданні кафедри інформаційних технологій та систем електронних комунікацій, протокол від «__» _____ 20__ №__

Розглянуто вченою радою навчально-наукового інституту цивільного захисту,
протокол від «__» _____ 20__ №__

Актуалізовано:

Дата перегляду / внесення змін					
Підпис					
Ім'я, прізвище викладача					

1. Загальна інформація

Назва дисципліни	Основи програмування
Статус дисципліни	Нормативна
Рік навчання, семестр	1-й рік (1 та 2 семестри)
Мова викладання	українська
Викладач	Смотр Ольга Олексіївна, к.т.н, доцент, професор кафедри інформаційних технологій та систем електронних комунікацій; Головатий Роман Русланович, к.т.н, доцент кафедри інформаційних технологій та систем електронних комунікацій; Назар Юлія Сергіївна, к.т.н., заступник начальника кафедри інформаційних технологій та систем електронних комунікацій;
Контактний телефон	
E-mail	olgasmotr@gmail.com
Сторінка курсу в ВУ	I семестр http://virt.ldubgd.edu.ua/course/view.php?id=1528 II семестр http://virt.ldubgd.edu.ua/course/view.php?id=1778
Консультації	Згідно з розкладом консультацій кафедри інформаційних технологій та систем електронних комунікацій

2. Анотація до курсу

Освітня програма підготовки бакалавра з спеціальності «Комп'ютерні науки» передбачає оволодіння здобувачами вищої освіти низки фахових компетенцій в області програмування, досягнення яких організовано шляхом вивчення низки курсів, а саме «Дискретна математика», «Чисельні методи», «Об'єктно-орієнтоване програмування», «Операційні системи та системне програмування», «Клієнт-серверне програмування», «Алгоритми та структури даних», «Web-технології та web-програмування», «Програмування для мобільних платформ», «Людино-машинна взаємодія» тощо. Проте одним із базових та початкових курсів, які являються фундаментальними для вивчення зазначених та інших дисциплін, є курс «Основи програмування».

Предметом вивчення навчального курсу є основи алгоритмічного мислення, а також фундаментальні підходи до процедурного та об'єктно-орієнтованого програмування. Інструментальними засобами для оволодіння предметом вивчення навчальної дисципліни є мови програмування Python, Java та середовища розробки PyCharm, Eclipse, IntelliJ IDEA.

До розробки курсу долучено зовнішніх стейкхолдерів: Кокотко Богдан Ярославович, Software engineer, компанія Scanmarket.

3. Мета і завдання курсу

Метою навчальної дисципліни є формування у здобувачів освіти системи теоретичних знань щодо базових технологій програмування та алгоритмізації, а також набуття практичних навиків створення програмних продуктів, аналізу і реалізації базових алгоритмів використовуючи мови програмування Python та Java.

Завдання:

- формування основних понять про структуру середовища розробки програмного забезпечення та особливості роботи з ним;
- формування та розвиток навиків з основ процедурного і об'єктно-орієнтованого програмування мовами Python та Java;
- розвиток фундаментальних понять алгоритмічного програмування.

Процес вивчення дисципліни спрямований на формування елементів таких компетентностей:

Спеціальні (фахові) компетентності:

- СК08 Здатність проектувати та розробляти програмне забезпечення із застосуванням різних парадигм програмування: узагальненого, об'єктно-орієнтованого, функціонального, логічного, з відповідними моделями, методами й алгоритмами обчислень, структурами даних і механізмами управління.

Програмні результати навчання (ПРН):

- РН06 Використовувати методи чисельного диференціювання та інтегрування функцій, розв'язання звичайних диференціальних та інтегральних рівнянь, особливостей чисельних методів та можливостей їх адаптації до інженерних задач, мати навички програмної реалізації чисельних методів;
- РН09 Розробляти програмні моделі предметних середовищ, вибирати парадигму програмування з позицій зручності та якості застосування для реалізації методів та алгоритмів розв'язання задач в галузі комп'ютерних наук.

У результаті вивчення навчальної дисципліни здобувач вищої освіти повинен:

знати:

- загальні відомості про архітектуру та принцип роботи ЕОМ, а також загальні засади кодування алфавітно-цифрової інформації в ЕОМ;
- основні позиції систем числення, правила переведення чисел між системами числення та виконання арифметичних операцій над ними;
- класифікацію програмного забезпечення, різновиди і основні характеристики високо- та низькорівневих мов програмування;
- основні принципи та порядок виконання трансляції (компіляції та інтерпретації) коду, принципи кросплатформності;
- основні технології та парадигми програмування;
- особливості роботи в інтегрованих середовищах розробки мовами Python і Java;
- типи даних та змінні в програмуванні;
- арифметичні, логічні оператори та оператори відношення в програмуванні;
- фундаментальні (базові) алгоритми програмування (лінійний, циклічний із розгалуженням);

- умовний оператор вибору if та особливості його застосування в мовах Python і Java;
- оператор множинного вибору switch та особливості його застосування в мовах Python і Java;
- цикли for, while та do-while та особливості їх застосування в мовах Python і Java;
- елементарні структури даних та масиви;
- особливості застосування потоків вводу виводу інформації;
- особливості застосування класу Math та стрічок в програмуванні;
- методи та інструменти побудови елементарних структур об'єктно-орієнтованих програм;
- поняття класів та екземплярів класів в ООП;
- особливості застосування методів (функцій) в ОПП;
- поняття змінних екземплярів класів та конструкторів;
- поняття констант, статичних змінних і методів в ООП;
- різновиди модифікаторів доступу та особливості їх застосування;
- базові поняття інкапсуляції в програмуванні;
- базові поняття поліморфізму в програмуванні;
- базові поняття наслідування в програмуванні;
- метакласи в програмуванні мовою Java;
- базові поняття про колекції та бібліотеки в програмуванні;
- основні принципи розробки графічного інтерфейсу користувача.

вміти:

- вибирати парадигму програмування з позицій зручності та якості застосування для реалізації методів і алгоритмів розв'язування задач у галузі комп'ютерних наук;
- створювати надійне та ефективне програмне забезпечення;
- реалізовувати програмно лінійні алгоритми, алгоритми з розгалуженням та циклічні алгоритми;
- створювати та циклічно наповнювати елементарні структури даних, працювати зі збереженими у масивах даними;
- реалізовувати можливість введення/виведення даних при виконанні алгоритмічних операцій;
- застосовувати методи класу Math з метою програмування арифметичних операцій;
- створювати класи та екземпляри класів, реалізовувати методи та конструктори в ООП;
- програмно реалізовувати основні парадигми ОПП (наслідування, поліморфізм та інкапсуляція);
- використовувати метакласи в об'єктно-орієнтованому програмуванні;
- застосовувати отримані знання з основ процедурного та об'єктно-орієнтованого програмування в практичних ситуаціях (розробляти програмне забезпечення із застосуванням процедурної та об'єктно-орієнтованої парадигм програмування);
- самостійно вчитись та оволодівати сучасними підходами, механізмами та технологіями створення програмних продуктів.

4. Передумови для вивчення дисципліни

Курс є базовим в програмі підготовки бакалавра за спеціальністю 122 «Комп'ютерні науки» та являється основою для вивчення таких дисциплін: «Об'єктно-орієнтоване програмування», «Операційні системи та системне програмування», «Програмування для мобільних платформ», «Людино-машинна взаємодія», «Клієнт-серверне програмування», «Алгоритми та структури даних», тощо. Сам курс опирається на фундаментальні курси «Дискретна математика», «Лінійна алгебра та аналітична геометрія», «Математичний аналіз» та отримані знання з шкільного курсу «Інформатики».

5. Формат і обсяг дисципліни

Формат курсу: Навчальний матеріал дисципліни структурований за двома змістовими модулями і складається з шести тем, які є логічно завершеними, відносно самостійними, цілісними частинами. Засвоєння теоретичного матеріалу курсу передбачає відвідування 24-х лекційних занять та здачу тестових завдань на базі електронного освітнього середовища (до кожної лекції). Освоєння практичної частини курсу зосереджено в рамках 32-х лабораторних занять, після яких здобувачі освіти отримують індивідуальні практичні завдання. Виконані завдання необхідно завантажувати до відповідної категорії електронного освітнього середовища для їх подальшого захисту на оцінку. В процесі вивчення курсу здобувачі вищої освіти можуть долучатись до розробки студентських R&D проєктів.

Здобувачам освіти в ході вивчення курсу надана можливість здобуття додаткових знань шляхом проходження відкритих он-лайн курсів «Основи програмування» (1 семестр), «Основи програмування на Java» (2 семестр) та «Основи програмування CS50 2019» (без семестрової прив'язки), які розміщено на платформі Prometheus. Здобувачі які успішно вивчили он-лайн курс з рекомендованого переліку, що засвідчено відповідним сертифікатом, звільняються від виконання обов'язкових індивідуальних практичних завдань та отримують допуск до підсумкового контролю (екзамену).

Обсяг дисципліни: 7,5 кредитів / 225 академічних годин, з яких: лекцій 48 годин, лабораторних 64 години, самостійної роботи 113 години.

Форми навчання: лекції, лабораторні заняття, консультації, самостійна робота (в тому числі здача тестових завдань та виконання індивідуальних практичних завдань у позааудиторний час з подальшим їх захистом на лабораторних заняттях).

6. Тематика та зміст курсу

Назви змістових модулів і тем	Кількість годин (денна форма)				
	усього	у тому числі			
		л	п	лаб	с.р.
1-ий семестр					
Змістовний модуль 1. Основи програмування мовою Python					
Тема 1. Основи програмного середовища					
Тема №1.1. Вступ до курсу. Апаратні та програмні засоби ЕОМ	8	2	2	-	4
Тема №1.3. Системи числення	9	2	2	-	5
Тема №1.5. Алгоритми. Базові структури алгоритмів	9	2	2	-	5
Тема 1.7. Огляд мов програмування	8	2	2	-	4
Разом за темою	34	8	8	-	18
Тема 2. Вступ до програмування мовою Python					
Тема №2.1. Середовище розробки мовою Python	8	2	2	-	4
Тема №2.3. Типи об'єктів мови Python. Числа та операції над ними	8	2	2	-	4
Тема №2.5. Алгоритми послідовної (лінійної) структури. Алгоритми розгалуженої структури (інструкція if)	9	2	2	-	5
Тема №2.7. Алгоритми циклічної структури (інструкція while)	8	2	2	-	4
Тема №2.9. Алгоритми циклічної структури (інструкція for)	9	2	2	-	5
Разом за темою	42	10	10	-	22
Тема 3. Функції та структури даних					
Тема №3.1. Списки: одновимірні та двовимірні масиви. Генерація випадкових чисел	9	2	2	-	5

Назви змістових модулів і тем	Кількість годин (денна форма)				
	усього	у тому числі			
		л	п	лаб	с.р.
Тема №3.3. Рядки символів. Множини	8	2	2	-	4
Тема №3.5. Функції користувача. Рекурсивні функції	9	2	2	-	5
Тема №3.7. Словник. Кортеж	8	2	2	-	4
Разом за темою	34	8	8	-	18
Тема 4. Організація проекту застосування					
Тема №4.1. Файли даних. Модульний принцип організації програми	8	2	2	-	4
Тема №4.3. Парадигма об'єктно-орієнтованого програмування (ООП)	9	2	2	-	5
Тема №4.5. Розробка додатків з графічним інтерфейсом на основі бібліотеки Tkinter	8	2	2	-	4
Разом за темою	25	6	6	-	13
Разом за семестр	135	32	32	-	71
2-ий семестр					
Змістовний модуль 2. Основи програмування мовою Java					
Тема 5. Основи процедурного програмування					
Тема №5.1. Серєовище розробки мовою Java. Базові елементи мови Java.	11	2	4	-	5
Тема №5.4. Оператори вибору	11	2	4	-	5
Тема №5.7. Циклічні оператори	11	2	4	-	5
Тема №5.10. Масиви	11	2	4	-	5
Тема №5.13. Потoki введення/виведення, рядки та числові методи в Java	11	2	4	-	5
Разом за темою	55	10	20	-	25
Тема 6. Основи об'єктно-орієнтованого програмування					
Тема №6.1. Введення в класи та методи	11	2	4	-	5
Тема №6.4. Поліморфізм. Інкапсуляція	12	2	4	-	6
Тема №6.7. Ієрархія класів. Наслідування. Абстракція	12	2	4	-	6
Разом за темою	35	6	12	-	17
Разом за семестр	90	16	32	-	42
Усього годин	225	48	64	-	113

7. Інформаційний обсяг навчальної дисципліни

Короткий зміст лекційних занять

Змістовий модуль 1. Основи програмування мовою Python.

Тема 1. Основи програмного середовища.

Загальні відомості про ЕОМ. Архітектура та принцип роботи ЕОМ. Програмне забезпечення та його класифікація. Файлова система. Загальні засади кодування алфавітно-цифрової інформації в ЕОМ. Основні позиції систем числення. Переведення чисел між системами числення. Арифметичні операції в системах числення. Загальні відомості про мови програмування. Мови низького рівня. Огляд основних мов програмування високого рівня. Поняття алгоритмів. Базові структури алгоритмів. Представлення алгоритмів. Алгоритми лінійних програм, програм з розгалуженням та циклічних програм.

Тема 2. Вступ до програмування мовою Python

Типи даних та змінні. Арифметичні оператори. Логічні оператори. Оператори відношення. Приведення типів. Оголошення та ініціалізація змінних. Середовище розробки мовою Python. Типи об'єктів мови Python. Числа та операції над ними. Реалізація алгоритмів послідовної (лінійної) структури. Реалізація алгоритмів розгалуженої структури (інструкція if). Цикли в програмуванні. Реалізація алгоритмів циклічної структури (інструкція while). Реалізація алгоритмів циклічної структури (інструкція for).

Тема 3. Функції та структури даних

Списки в мові Python. Одновимірні масиви. Двовимірні масиви. Генерація випадкових чисел. Рядки символів в Python. Множини. Функції користувача. Рекурсивні функції. Словник. Кортеж.

Тема 4. Організація проекту застосування

Файли даних. Модульний принцип організації програми в Python. Парадигма об'єктно-орієнтованого програмування в Python. Наслідування атрибутів класу в ООП. Розробка додатків з графічним інтерфейсом на основі бібліотеки tkinter.

Змістовий модуль 2. Основи програмування мовою Java.

Тема 5. Основи процедурного програмування мовою Java

Встановлення та налаштування JDK. Встановлення та налаштування IDE. Огляд об'єктно-орієнтованої мови Java. Інтегроване середовище розробки Eclipse. Інтегроване середовище розробки IntelliJ IDEA. Перший застосунок в середовищі розробки "Hello world!". Структура коду програми. Приведення типів. Оголошення та ініціалізація змінних. Умовний оператор if. Оператор розгалуження (множинного вибору) switch. Написання програм з розгалуженням використовуючи умовний оператор if. Написання програм з розгалуженням використовуючи оператор множинного вибору switch. Цикл з передумовою while. Цикл з післяумовою do-while. Цикл з лічильником for. Написання циклічних програм з використанням циклів for, while, do-while. Одновимірні та багатовимірні масиви в Java. Створення та наповнення масивів. Циклічне наповнення масивів. Потоки вводу та виводу інформації. Написання програм із можливістю введення даних з клавіатури. Стрічки в програмуванні мовою Java. Клас String. Математичний клас Math. Реалізація найпростіших алгоритмів (лінійні, циклічні, з розгалуженням) методами процедурного

програмування.

Тема 6. Основи об'єктно-орієнтованого програмування мовою Java

Основи ООП мовою Java. Класи та екземпляри класів. Змінні екземпляра класу. Конструктори. Методи класу. Типізовані та void-методи. Методи, що приймають параметри. Використання об'єктів в якості параметрів методу. Аргументи динамічної довжини. Парадигми ООП. Модифікатори доступу. Інкапсуляція. Керування доступом. Методи get та set. Поліморфізм. Перевизначення методів. Перевизначення конструкторів. Наслідування. Ієрархія класів. Ключове слово super. Статичні змінні та методи. Ключові слова static та final. Абстракція. Написання найпростіших програм із використанням принципів об'єктно-орієнтованого підходу.

8. Теми практичних, семінарських та лабораторних занять

8.1. Теми практичних занять

№ з/п	Назва теми	Кількість годин
1	ЗМ1. Тема №1.2. Правила техніки безпеки при роботі з ПК. Ознайомлення з віртуальним навчальним середовищем «Віртуальний університет», реєстрація у середовищі moodle	2
2	ЗМ1. Тема №1.4. Переведення чисел в різні системи числення. Арифметичні операції в системах числення	2
3	ЗМ1. Тема №1.6. Побудова алгоритмів, розв'язок алгоритмічних задач	2
4	ЗМ1. Тема №1.8. Встановлення і налаштування середовища розробки	2
5	ЗМ1. Тема №2.2. Знайомство з роботою в середовищі розробки. Введення даних з клавіатури	2
6	ЗМ1. Тема №2.4. Реалізація алгоритмів послідовної (лінійної) структури моаю Python	2
7	ЗМ1. Тема №2.6. Реалізація алгоритмів розгалуженої структури (інструкція if) на мові Python	2
8	ЗМ1. Тема №2.8. Реалізація алгоритмів циклічної структури (інструкція while) мовою Python	2
9	ЗМ1. Тема №2.10. Реалізація алгоритмів циклічної структури (інструкція for) мовою Python	2
10	ЗМ1. Тема №3.2. Робота зі списками, масивами. Модуль random	2
11	ЗМ1. Тема №3.4. Робота з рядками символів із використанням функцій мови Python	2
12	ЗМ1. Тема №3.6. Побудова функцій користувача мовою Python	2
13	ЗМ1. Тема №3.8. Реалізація алгоритмів з використанням словника і кортежу мовою Python	2
14	ЗМ1. Тема №4.2. Реалізація алгоритмів з використанням бінарного та текстового файлів (Python)	2
15	ЗМ1. Тема №4.4. Створення класу та об'єктів-екземплярів класу	2

№ з/п	Назва теми	Кількість годин
	(Python)	
16	ЗМ1. Тема №4.6. Розробка елементів графічного інтерфейсу за допомогою стандартної бібліотеки Tkinter	2
17	ЗМ2. Тема №5.2. Встановлення і налаштування середовища розробки.	2
18	ЗМ2. Тема №5.4. Перший застосунок в IDE	2
19	ЗМ2. Тема №5.6. Написання лінійних програм. Застосування операторів (Java)	2
20	ЗМ2. Тема №5.8. Написання програм з розгалуженням (Java)	2
21	ЗМ2. Тема №5.10. Написання циклічних програм (Java)	2
22	ЗМ2. Тема №5.12. Створення та обробка елементарних структур даних (масиви) (Java)	2
23	ЗМ2. Тема №5.14. Застосування потоків введення/виведення та рядків в програмуванні (Java)	2
24	ЗМ2. Тема №5.16. Підсумкове заняття з основ процедурного програмування мовою Java. Створення програми на основі процедурного підходу	2
25	ЗМ2. Тема №6.2. Створення власних класів. Застосування конструкторів	2
26	ЗМ2. Тема №6.4. Створення власних методів з довільною логікою	2
27	ЗМ2. Тема №6.6. Робота з методами: застосування поліморфізму	2
28	ЗМ2. Тема №6.8. Робота з змінними та методами: інкапсуляція. Використання методів get() та set()	2
29	ЗМ2. Тема №6.10. Застосування наслідування	2
30	ЗМ2. Тема №6.12. Використання Enums та Interface в об'єктно-орієнтованому програмуванні	2
31	ЗМ2. Тема №6.14. Робота з Map, Set та List	2
32	ЗМ2. Тема №6.16. Проектування та розробка елементів графічного інтерфейсу користувача	2
	Разом	64

8.2. Теми семінарських занять

Семінарські заняття не передбачені

8.3. Теми лабораторних занять

Лабораторні заняття не передбачені

9. Самостійна робота

№ з/п	Назва теми	Кількість годин (денна форма)
1	Підготовка до практичних занять, складання тестових завдань	44
2	Виконання індивідуальних практичних завдань. Підготовка звітів	44
3	Підготовка до екзамену	25
	Разом	113

1. З метою закріплення отриманих практичних навиків, здобувачі освіти виконують індивідуальні практичні завдання, які отримують в кінці лабораторного заняття. Зміст та варіанти виконання практичних завдань (методичні рекомендації) відображені на платформі електронного освітнього середовища «Віртуальний університет». Звіт про виконання індивідуальних практичних завдань завантажується у відповідну категорію електронного освітнього середовища для подальшої перевірки викладачем та його захисту на оцінку.

2. Проходження, за бажання, он-лайн курсу (неформальна освіта) на платформі відкритих он-лайн курсів Prometheus:

- [основи програмування](#) (1 семестр);
- [основи програмування на Java](#) (2 семестр);
- [основи програмування CS50 2019](#) (без семестрової прив'язки).

Здобувачі які успішно пройшли он-лайн курс з рекомендованого переліку, що засвідчено відповідним сертифікатом, за бажання звільняються від обов'язкового виконання індивідуальних практичних завдань (перезараховується 60 балів за практичну частину). Сертифікати про проходження курсу завантажуються у відповідну категорію електронного освітнього середовища.

10. Методи навчання

Основні форми організації навчання: лекції; лабораторні заняття з проведенням зрізів знань; індивідуальні практичні завдання, консультації.

Методи організації та здійснення навчально-пізнавальної діяльності:

- лекції – словесні та наочні методи навчання із елементами мозкового штурму;
- лабораторні роботи – репродуктивний метод, метод проблемного викладу матеріалу;
- індивідуальні практичні завдання – частково-пошуковий метод навчання;
- консультації – словесний та дискусійний методи.

11. Технічне й програмне забезпечення /обладнання

Комп'ютери на базі процесорів Intel Pentium Gold G5400 або Intel Core i5, компоненти програмного забезпечення MS Office 365 (Teams, PowerPoint, Word, SharePoint, PowerApps), інтегровані середовища розробки: Eclipse, IntelliJ IDEA, PyCharm, комплект розробника java (JDK), інтерактивна система управління комп'ютерами Veyon, електронне освітнє середовище «Віртуальний університет» (на базі платформи Moodle).

12. Критерії оцінювання

Оцінювання результатів навчання здобувачів вищої освіти здійснюється відповідно до «Положення про організацію освітнього процесу у ЛДУ БЖД» <https://cutt.ly/0WRAkEh> та «Положення про порядок та критерії оцінювання результатів навчання здобувачів вищої освіти у ЛДУ БЖД» <https://cutt.ly/iWRAWF3>.

Поточний контроль	
Поточний контроль проводиться у формі виконання тестових завдань на базі платформи електронного освітнього середовища «Віртуальний університет», усного опитування, виконання індивідуальних практичних завдань та їх захисту на оцінку. Оцінювання результатів поточного контролю здійснюється за накопичувальною (100-бальною) шкалою. Результати поточного контролю (поточна успішність) враховуються викладачем при визначенні допуску до підсумкового контролю.	
1-й семестр	
Вид робіт	Формат проведення та критерії оцінювання
Тестові завдання	В першій частині курсу здобувачі освіти мають можливість одержати за проходження усіх тестових завдань сумарно до ~30 балів . Семестровою частиною курсу передбачено проходження 6 тестових завдань (0 –5 балів за тестове завдання). Критерії оцінювання наведені в електронному курсі «Віртуального університету».
Індивідуальні практичні завдання	Першою частиною курсу передбачено виконання 12 індивідуальних практичних завдань, за успішне виконання та захист яких здобувачі можуть отримати сумарно до 60 балів (0 – 5 балів за практичне завдання). Критерії оцінювання наведені в електронному курсі «Віртуального університету». Здобувачі які успішно вивчили он-лайн курс з рекомендованого переліку, що засвідчено відповідним сертифікатом, звільняються від виконання індивідуальних практичних завдань (перезараховується 60 балів до поточного контролю).
Робота в аудиторії	Відвідуваність, виконання домашніх завдань, активність під час заняття, ведення конспекту, участь у конференціях, олімпіадах, хакатонах, - сумарно до 10 балів
2-й семестр	
Вид робіт	Формат проведення та критерії оцінювання
Тестові завдання	В другій частині курсу здобувачі освіти мають можливість одержати за проходження усіх тестових завдань сумарно до 40 балів . Семестровою частиною курсу передбачено проходження 8 тестових завдань (0 – 5 балів за тестове завдання). Критерії оцінювання наведені в електронному курсі «Віртуального університету».
Індивідуальні практичні завдання	Першою частиною курсу передбачено виконання 15 індивідуальних практичних завдань, за успішне виконання та захист яких здобувачі можуть отримати сумарно до 60 балів (0 – 4 балів за практичне завдання). Критерії оцінювання наведені в електронному курсі «Віртуального університету». Здобувачі які успішно вивчили он-лайн курс з рекомендованого переліку, що засвідчено відповідним сертифікатом, звільняються від виконання індивідуальних практичних завдань (перезараховується 60 балів до поточного контролю).
Підсумковий контроль	
1-ий семестр	
Семестровий контроль проводиться у формі екзамену. До підсумкового контролю допускаються здобувачі освіти, які мають належний рівень поточних знань та одержали понад 60 зі 100 можливих балів за результатами проходження курсу (поточного оцінювання) на базі електронного освітнього середовища «Віртуальний університет». Одержання необхідної кількості балів поточного контролю можливе за рахунок	

проходження тестових завдань (теоретична частина), виконання індивідуальних практичних завдань (практична частина) та активної участі в навчальному процесі (Робота в аудиторії) . Розподіл між запропонованими складовими поточного контролю знань за навчальний семестр наступний: тестові завдання – до 30 балів, індивідуальні практичні завдання – до 60 балів, робота в аудиторії – до 10 балів.

Екзамен (максимально 100 балів)

Усі отримані бали протягом навчання враховуються у екзаменаційну оцінку з коефіцієнтом 0,25 (максимально 25 балів). Іспит складається із теоретичної частини (35 балів) та практичної частини (40 балів) і проводиться на базі електронного освітнього середовища «Віртуальний університет». На теоретичну частину виноситься 35 випадково згенерованих тестових питань закритої форми з вибором однієї правильної відповіді за матеріалами семестрової частини курсу. Вага кожного питання тесту = 1 балу. Практична частина передбачає розв'язок 2-х практичних завдань, а саме написання прикладного програмного коду згідно отриманого завдання. Кожне питання у білеті оцінюється у 20 балів (максимально).

Критерії оцінювання виконаних завдань:

- 20 балів – написаний код є працездатним, а структура та результат його виконання в повній мірі відповідають поставленому завданню;
- 11-19 балів – написаний код є працездатним, але його структура або результат виконання не в повній мірі відповідають поставленому завданню;
- 6-10 балів – написаний код є працездатним, але його структура або результат виконання лише частково відтворюють умову поставленого завдання;
- 1-5 балів – написаний код не працездатний, проте в його структурі спостерігається вірно обраний напрям вирішення завдання;
- 0 балів – завдання не виконане або не відповідає поставленому завданню.

Підсумкова семестрова оцінка обчислюється як сума балів поточного (з коефіцієнтом 0,25 та округлення до цілого числа) та підсумкового контролю за 100-бальною шкалою і переводяться в національну (чотирибальну) шкалу (“відмінно”, “добре”, “задовільно”, “незадовільно”).

Підсумкові оцінки виставляються та вносяться до екзаменаційної відомості в національній, 100-бальній шкалі та шкалі ЄКТС відповідно до співвідношень, поданих у наступній таблиці.

2-ий семестр

Семестровий контроль проводиться у формі екзамену. До підсумкового контролю допускаються здобувачі освіти, які мають належний рівень поточних знань та одержали понад **60** зі **100** можливих балів за результатами проходження курсу (поточного оцінювання) на базі електронного освітнього середовища «Віртуальний університет».

Одержання необхідної кількості балів поточного контролю можливе за рахунок проходження тестових завдань (теоретична частина) та виконання індивідуальних практичних завдань (практична частина). Розподіл між запропонованими складовими поточного контролю знань за навчальний семестр наступний: тестові завдання – до 40 балів, індивідуальні практичні завдання – до 60 балів.

Екзамен (максимально 100 балів) складається із теоретичної та практичної частини (по 50 балів кожна) і проводиться на базі електронного освітнього середовища «Віртуальний університет». На теоретичну частину виноситься 50 випадково згенерованих тестових питань закритої форми з вибором однієї правильної відповіді за матеріалами семестрової частини курсу. Вага кожного питання тесту = 1 балу. Практична частина передбачає розв'язок 2-х практичних завдань, а саме написання прикладного програмного коду згідно отриманого завдання. Кожне питання у білеті оцінюється у 25 балів (максимально).

Критерії оцінювання виконаних завдань:

- 25 балів – написаний код є працездатним, а структура та результат його виконання в повній мірі відповідають поставленому завданню;
- 20-24 балів – написаний код є працездатним, але його структура або результат виконання не в повній мірі відповідають поставленому завданню;
- 10-19 балів – написаний код є працездатним, але його структура або результат виконання лише частково відтворюють умову поставленого завдання;
- 1-9 балів – написаний код не працездатний, проте в його структурі спостерігається вірно обраний напрям вирішення завдання;
- 0 балів – завдання не виконане або не відповідає поставленому завданню.

Підсумкову семестрову оцінку розраховують як суму балів теоретичної та практичної частини за 100-бальною шкалою і переводяться в національну (чотирибальну) шкалу (“відмінно”, “добре”, “задовільно”, “незадовільно”, для заліків – “зараховано”, “не зараховано”).

Підсумкові оцінки заносяться до екзаменаційної відомості в національній, 100-бальній шкалі та шкалі ЄКТС відповідно до співвідношень, поданих у наступній таблиці.

Шкала оцінювання результатів навчання здобувачів вищої освіти

Сума балів за всі види навчальної діяльності	Оцінка а ECTS	Оцінка за національною шкалою	
		для екзамену, диференційованого заліку, курсового проекту (роботи), практики	для заліку
91 – 100	A	відмінно	зараховано
81-90	B	добре	
71-80	C		
61-70	D	задовільно	
51-60	E		
36-50	FX	незадовільно	не зараховано
0-35	F		

13. Політика курсу

Виконання навчальних завдань і робота на курсі має відповідати вимогам «Кодекс академічної доброчесності та корпоративної культури ЛДУ БЖД» <https://cutt.ly/5WTlc6y>.

Академічні очікування від здобувачів: своєчасне виконання тестових завдань, передбачених силабусом дисципліни; обов’язкове відвідування лабораторних занять і виконання індивідуальних практичних (завдань самостійної роботи).

Політика щодо термінів виконання завдань та ліквідації академічної заборгованості: терміни виконання завдань вказуються у електронному курсі «Віртуального університету». Після завершення терміну прийому завдань, система блокує можливість їх завантаження для подальшої оцінки викладачем, окрім випадків пов’язаних із поважними причинами, про що здобувач особисто повідомляє викладача. Відпрацювання академічної заборгованості з дисципліни можливо до дня проведення підсумкового контролю (залік, екзамен) згідно графіку консультацій випускової кафедри або за індивідуальними домовленостями з викладачем.

Недопущені до підсумкового контролю здобувачі освіти здійснюють перездачу в терміни, відведені для усунення академічної заборгованості у два етапи:

- заборгованість із поточного контролю;
- заборгованість із підсумкового контролю.

Ліквідація заборгованості поточного контролю відбувається шляхом проходження тестових завдань та виконання індивідуальних практичних завдань згідно із тематичним планом курсу. Ліквідація заборгованості з підсумкового контролю організовується в форматі перездачі диф.залику/екзамену.

Дотримання принципів академічної доброчесності: роботи (завдання) виконуються здобувачами самостійно, ідеї та ініціативи інших авторів використовуються лише при належно оформленому цитуванні. Групове вирішення завдань дозволяється за попереднього декларування ідеї та в рамках реалізації студентських R&D проєктів.

Поведінка в аудиторії: неприпустимо запізнення на заняття; користування телефоном або іншими гаджетами на занятті, за винятком виконання громіздких обчислень та використання додаткових програм в освітніх цілях; повага до думки інших колег; дотримання норм культури мовлення; соціальна солідарність та ін.

14. Рекомендована література

14.1. Основна:

1. *Основи програмування (мовою Java) : курс лекцій* / Придатко О., Хлєвной О., Бурак Н. – Львів : ЛДУ БЖД, 2019. – 180 с.
2. *Основи програмування (Python, Java) : лабораторний практикум* / Смотр О., Придатко О., Малець І. – Львів : ЛДУ БЖД, 2019. – 134 с.
3. *Java 8. The Complete Reference. 9 edition* / Herbert Schildt. – New York : Oracle Press, 2015. – 1274 с.
4. *Java. Бібліотека професіонала. Т 1. Основи. 11-е видання* / Кей С. Хорстманн. – Наука і техінка, 2020. – 864 с.
5. *Spring 5 для професіоналів* / Кларенс Хо, Кріс Шефер, Роб Харрон, Юліана Козміна. – Наука і техінка, 2020. – 1120 с.
6. *Вивчаємо Python : 4-е вид. пер. з англ.* / Лутц М. – Символ-Плюс, 2020. – 832 с.
7. *Програмування числових методів мовою Python : підруч.* / за ред. А. В. Анісімова. – К. : Видавничо-поліграфічний центр "Київський університет", 2014. – 640 с.
8. *Head First Java* / Kathy Sierra, Bert Bates. – «O'REILLY», 2012. – 678 с.
9. *Легкий метод вивчити Python : пер. з англ.* / Зед Шоу. – Видавництво «Е», 2017. – 352 с.

14.2. Додаткова:

1. *Придатко О. В. Інтеграція новаційного методу мобільного навчання в освітні проєкти підготовки розробників програмного забезпечення* / О. В. Придатко, В. В. Придатко, Ю. О. Борзов, В. Є. Дзень // Вісник ЛДУБЖД: Зб. наук. праць. Львів: ЛДУ БЖД, 2018. – №18. – С.70-80.
2. *Martyn Ye. Software for Shelter's Fire Safety and Comfort Levels Evaluation* / Martyn Ye., Smotr O., Burak N., Prydatko O., Malets I. // Communications in Computer and Information Science, Springer, Cham. – Vol. 1158, 2020. pp. 457-469 https://doi.org/10.1007/978-3-030-61656-4_31
3. *Kordunova, Y., Prydatko, O., Smotr, O., Golovaty, R. (2023). Expert Decision Support System Modeling in Lifecycle Management of Specialized Software. In: Babichev, S., Lytvynenko, V. (eds) Lecture Notes in Data Engineering, Computational Intelligence, and Decision Making. ISDMCI 2022. Lecture Notes on Data Engineering and Communications Technologies, vol 149, pp 367–383. Springer, Cham. https://doi.org/10.1007/978-3-031-16203-9_22*
4. *Придатко О. В. Адаптивна інформаційно-довідкова система "UniBell" як складова частина проєкту "Smart-університет"* / О. В. Придатко, Н. Є. Бурак, В. Є. Дзень, М. С. Кунинець // Науковий вісник НЛТУ України : Зб. наук. праць. Львів: НЛТУ, 2020. - т. 30, № 5 – С. 113–121.
5. *Головатий Р.Р. Модель проєкту інтелектуальної аналітичної системи обліку наукових досягнень* / Р.Р. Головатий, О.О. Смотр, І.О. Малець, Н.Є. Бурак // Středoevropský věstník pro vědu a výzkum: Central European Journal for Science and Research – Praha. – 2019. – Bun. 3(55). – С. 88-94.
6. *Nazarkevych M. Identification of Biometric Images Using Latent Elements* / Nazarkevych M., Logoyda M., Dmytruk S., Voznyi Ya., Smotr O. // CEUR Workshop Proceedings. – Vol. 2488, 2019. –P.99-108.

7. Smotr, O. Increasing the animation study management services functioning efficiency / O. Smotr , S. Ljaskovska, I. Malets, O. Karabyn // In: *Proceedings of the 2020 IEEE Third International Conference on Data Stream Mining & Processing (DSMP)*, Lviv, Ukraine, August 21-25, 2020. – pp. 404-408.
8. Рижавський К. Є. Комп'ютерні графічні технології у підготовці фахівців технічного спрямування / К. Є. Рижавський, Є. В. Мартин, О. В. Придатко // Сучасні проблеми моделювання: Зб. наук. праць. Мелітопіль: МДПУ ім. Б. Хмельницького, 2016. - №7 – С.130-137.
9. Придатко О. В. Інтеграція 3D-інтерактивних технологій навчання в освітні проекти безпеко-орієнтованих спеціальностей / О. В. Придатко, А. Г. Ренкас, Н. Є. Бурак, М. В. Лемішко // Вісник ЛДУБЖД: Зб. наук. праць. Львів: ЛДУ БЖД, 2017. – №15. – С.46-54.
10. Андрушко О. А. Аналіз процесів використання Docker для побудови мікросервісів / О. А. Андрушко, Ю. О. Борзов, І. О. Малець, О. В. Придатко // Науковий вісник НЛТУ України: Зб. наук. праць. Львів: НЛТУ, 2017. - №9(27) – С.95-98.
11. Prydatko O. Informational System of Project Management in the Areas of Regional Security Systems' Development / O. Prydatko, O. Smotr, Yu. Borzov, I. Solotvinskyi, O. Didyk // 2018 IEEE Second Conference on Data Stream Mining & Processing. Lviv, 2018. – №2 – 187-192.
12. Ljaskovska S. Information Technology of Process Modeling in the Multiparameter Systems / S. Ljaskovska, I. Malets, Ye. Martyn, O. Prydatko // 2018 IEEE Second Conference on Data Stream Mining & Processing. Lviv, 2018. – №2 – 177-182.
13. Malets I. Interactive Computer Simulators in Rescuer Training and Research of their Optimal Use Indicator / I. Malets, O. Prydatko, V. Popovych, A. Dominik // 2018 IEEE Second Conference on Data Stream Mining & Processing. Lviv, 2018. – №2 – 558-562.

14.3. Інформаційні ресурси:

1. Prometheus. Курс «*Основи програмування*». [Електронний ресурс]. – Доступний з <https://edx.prometheus.org.ua/courses/KPI>
2. Prometheus. Курс «*Основи програмування на Java*». [Електронний ресурс]. – Доступний з <https://edx.prometheus.org.ua/courses/KPI>
3. Prometheus. Курс «*Курс CS-50*». [Електронний ресурс]. – Доступний з <https://edx.prometheus.org.ua/courses/KPI>
4. *Освоюємо Java. Вікіпідручник* [Електронний ресурс]. – Доступний з https://uk.wikibooks.org/wiki/Освоюємо_Java