

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ДОНБАСЬКА ДЕРЖАВНА МАШИНОБУДІВНА АКАДЕМІЯ
Кафедра «Автоматизація виробничих процесів»

“ЗАТВЕРДЖУЮ”
Ректор ДДМА
В.Д.Ковалев
“ 04 ” 2020 року



РОБОЧА ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ
„КОМП’ЮТЕРНІ ТЕХНОЛОГІЇ ТА ПРОГРАМУВАННЯ”
(назва дисципліни)

Галузь знань: 12 «Інформаційні технології»

Спеціальність 123 «Комп’ютерна інженерія»

Освітній рівень – перший (бакалаврський)

ОПП «Комп’ютерні системи та мережі»

Факультет «Машинобудування»

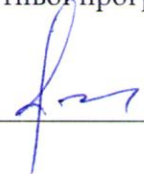
(назва інституту, факультету, відділення)

Робоча програма навчальної дисципліни «Комп'ютерні технології та програмування» для студентів галузі знань 12 «Інформаційні технології» спеціальності 123 «Комп'ютерна інженерія».

Розробник: **Костіков О.А.**, канд. фіз.-мат. наук, доц.

Погоджено з групою забезпечення освітньої програми (для обов'язкових дисциплін).

Керівник групи забезпечення:


_____ О.В. Суботін, к.т.н., доцент


Розглянуто і затверджено на засіданні кафедри «Автоматизація виробничих процесів», протокол № 10 від 22.06.2020 року.

Завідувач кафедри АВП:


_____ Г.П. Клименко, д.т.н., професор

Розглянуто і затверджено на засіданні Вченої ради факультету машинобудування, протокол № 01 від 31.08.2020 року.

20/08
Голова Вченої ради факультету:


_____ В.Д. Кассов, д.т.н., професор

1.Опис навчальної дисципліни

Показники		Галузь знань, спеціальність, ОПП (ОНП), професійне (наукове) спрямування, рівень вищої освіти	Характеристика навчальної дисципліни	
			денна	прискорена
Кількість кредитів		Галузь знань: 12 «Інформаційні технології». Спеціальність: 123 «Комп'ютерна інженерія». ОПП «Комп'ютерні системи та мережі»	Обов'язкова дисципліна	
10	6			
Загальна кількість годин				
300	180			
Модулів – 2		Рівень вищої освіти: <u>перший (бакалаврський)</u>	Рік підготовки	
Змістових модулів –2			1	1
Індивідуальне науково-дослідне завдання _ <u>не передбачено</u>			Семестр	
			1,2	1
Тижневих годин для <u>денної</u> форми навчання: аудиторних –6; самостійної роботи студента – 6			Лекції	
		48	30	
		Лабораторні		
		81	60	
		Практичні		
		Самостійна робота		
		171	90	
		Вид контролю		
		екзамен	екзамен	

2. ЗАГАЛЬНІ ВІДОМОСТІ, МЕТА І ЗАВДАННЯ ДИСЦИПЛІНИ

Мета дисципліни - освоєння загальних принципів розробки програм з використанням сучасних мов програмування та одержання навичок практичного застосування структурного та об'єктного підходів в реалізації алгоритмів на мовах C++, Python.

Завдання дисципліни - дати студентам теоретичну та практичну підготовку з основ структурного та об'єктно-орієнтованого програмування на мовах C++ та Python.

Програмні компетентності. Освітня компонента «Комп'ютерні технології та програмування» повинна сформувати наступні загальні та фахові програмні компетентності, що передбачені освітньо-професійною програмою підготовки бакалаврів «Комп'ютерні системи та мережі»:

ЗК1. Здатність до абстрактного мислення, аналізу і синтезу.

ЗК2. Здатність вчитися і оволодівати сучасними знаннями.

ЗК3. Здатність застосовувати знання у практичних ситуаціях.

ЗК7. Вміння виявляти, ставити та вирішувати проблеми.

ЗК8. Здатність працювати в команді.

ФК1. Здатність застосовувати законодавчу та нормативно-правову базу, а також державні та міжнародні вимоги, практики і стандарти з метою здійснення професійної діяльності в галузі комп'ютерної інженерії.

ФК2. Здатність використовувати сучасні методи і мови програмування для розроблення алгоритмічного та програмного забезпечення.

ФК4. Здатність забезпечувати захист інформації, що обробляється в комп'ютерних та кіберфізичних системах та мережах з метою реалізації встановленої політики інформаційної безпеки.

ФК11. Здатність оформляти отримані робочі результати у вигляді презентацій, науково-технічних звітів.

ФК13. Здатність вирішувати проблеми у галузі комп'ютерних та інформаційних технологій, визначати обмеження цих технологій.

ФК16. Здатність до математичного та логічного мислення, знання понять, ідей і методів фундаментальної математики та фізики, вміння їх використовувати під час розв'язання конкретних завдань.

Програмні результати навчання. Освітня компонента «Комп'ютерні технології та програмування» повинна сформувати наступні програмні результати навчання, що передбачені освітньо-професійною програмою підготовки бакалаврів «Комп'ютерні системи та мережі»:

ПР1. Знати та розуміти наукові положення, що лежать в основі функціонування комп'ютерних засобів, систем та мереж.

ПР3. Знати новітні технології в галузі комп'ютерної інженерії.

ПР11. Вміти розробляти програмне забезпечення для вбудованих і розподілених застосувань, мобільних і гібридних систем, розраховувати, експлуатувати, типове для спеціальності обладнання.

ПР13. Вміти ефективно працювати як індивідуально, так і у складі команди.

ПР17. Вміти оцінювати отримані результати та аргументовано захищати прийняті рішення.

ПР20. Використовувати інформаційні технології та для ефективного спілкування на професійному та соціальному рівнях.

Передумови для вивчення дисципліни: «Вища математика».

Мова викладання: українська.

Обсяг навчальної дисципліни та його розподіл за видами навчальних занять:

- загальний обсяг для денної форми навчання становить 165 годин / 5,5 кредити, в тому числі: лекції - 30 годин, лабораторні роботи - 30 годин, самостійна робота студентів - 105 годин;

- загальний обсяг для прискореної форми навчання становить 105 годин / 3,5 кредити, в тому числі: лекції - 36 годин, практичні заняття - 18 годин, самостійна робота студентів - 66 годин.

3. ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

Змістовий модуль 1.

Введення в розробку і кодування алгоритмів на мові C++.

Тема 1. Алгоритм як основне поняття програмування.

Поняття алгоритму. Властивості алгоритму. Типові алгоритмічні конструкції. Способи завдання алгоритму. Критерії оцінки алгоритмів. Етапи розв'язування задач на комп'ютері. Приклади найпростіших типових алгоритмів.

Тема 2. Основні поняття мови C++.

Структура C++ програми. Лексичні елементи мови C++: алфавіт, коментарі, ідентифікатори, службові слова, дані, вираз, операнд, змінна, операція. Поняття типу даних. Класифікація і представлення даних. Базові типи даних: логічний, символьний, цілий. Перетворення типів. Пріоритети операції. Операції. Стандартні математичні функції. Константні величини: цілі, перелічувальні, символьні (літерні), рядкові (рядки або літерні рядки). Визначення констант.

Тема 3. Введення-виведення даних у мові C++.

Базові положення системи вводу-виводу C++. Потоки і буфери. Стандартні об'єкти вводу-виводу. Ввід даних за допомогою глобального об'єкта cin. Вивід рядків. Введення одного символу. Використовування функції get(): без параметрів, з параметрами. Ввід рядків із стандартного пристрою введення. Використовування функції getline(). Вивід даних за допомогою глобального об'єкта cout. Маніпулятори вводу-виводу. Функції width(), precision(), fill().

Тема 4. Умовні оператори.

Операторний блок. Оператори-вирази. Умовні оператори: умовний оператор if; оператор вибору switch. Вкладеність умовних операторів. Логічні операції. Умовна операція.

Тема 5. Оператори циклу.

Оператори циклу: оператор for; оператор while; оператор do– while. Оператори переходу: оператор goto; оператор break; оператор continue; оператор return. Оператори завершення програми.

Тема 6. Функції користувача.

Загальні відомості про функції. Структура функції. Значення, параметри і аргументи, що повертаються. Оголошення функції. Прототипи функцій. Визначення функції. Виконання функції. Локальні і глобальні змінні. Правило видимості змінних. Список параметрів функції. Параметри за умовчанням. Способи передачі параметрів. Способи повернення значення. Перевантаження функцій. Рекурсія. Робота функцій.

Змістовий модуль 2

Складні структури даних в C++. Об'єктно-орієнтоване програмування на мові C++.

Тема 7. Масиви.

Масиви. Оголошення масивів. Ініціалізація масивів. Обробка одновимірних масивів даних. Алгоритми сортування масивів. Багатовимірні масиви. Ініціалізація багатовимірного масиву. Типові приклади обробки матриць. Масиви як параметри функцій.

Тема 8. Показчики та робота з пам'яттю в C++.

Поняття покажчика, посилання. Покажчики і масиви. Адресна арифметика. Посилання. Приклади використання покажчиків і посилань. Покажчики на функції. Посилання. Параметри функцій як посилання. Організація пам'яті в сучасних процесорах і покажчики мови C++. Моделі пам'яті. Статичні і динамічні змінні. Оператори new і delete.

Тема 9. Динамічні масиви у C++.

Визначення динамічного масиву. Виділення пам'яті для одновимірних та багатовимірних динамічних масивів. Зміна розміру динамічного масиву. Переваги і недоліки використання динамічних масивів. Приклади обробки динамічних масивів.

Тема 10. Робота з рядками.

Рядки як символні масиви. Введення-виведення символних масивів. Основні функції обробки символних типів. Використання рядків типу string.

Тема 11. Структури та об'єднання.

Оголошення та ініціалізація структур. Розмір структури. Операція присвоєння для структур. Вкладені структури. Масиви структур. Оголошення та ініціалізація об'єднань. Звертання до елементів структур і об'єднань. Структури та об'єднання як параметри та результати функцій.

Тема 12. Робота з файлами у мові C++.

Файли і потоки. Стандартні функції відкриття та закриття файлових потоків. Режими відкриття файлових потоків. Стандартні файли введення-виведення. Пряме введення-виведення на консоль. Читання і запис текстових файлів. Форматне введення-виведення даних. Читання і запис двійкових файлів. Стандартні функції позиціонування файлових потоків.

Тема 13. Динамічні структури даних.

Лінійні списки. Стеки. Черги. Бінарні дерева. Реалізація динамічних структур за допомогою масивів.

Тема 14. Робота з класами у мові C++.

Елементи об'єктно-орієнтованого програмування та деякі особливості мови C++. Основні принципи технології ООП. Класи в мові C++. Члени класу – дані та функції, права доступу до членів класу. Приклади визначення класів. Інкапсуляція – механізм приховування даних. Рівні доступу до членів класу. Клас як тип даних. Виклик методів класу. Конструктори та деструктор класу. Автоматична ініціалізація. Конструктор за замовчуванням. Конструктор з параметрами. Деструктори.

Змістовий модуль 3 Основи мови Python.

Тема 15. Типи об'єктів мови Python.

Числа. Базові числові типи. Рядки. Літерали рядків. Рядки в дії. Строкові методи. Вирази форматування рядків. Метод форматування рядків. Загальні категорії типів Списки. Словники. Кортежі. Файли. Ієрархії типів даних в мові Python. Інші базові типи.

Тема 16. Інструкції мови Python.

Структура програми на мові Python. Інструкції присвоювання. Інструкції виразів. Операція print. Умовні інструкції if. Синтаксичні правила мови Python. Перевірка істинності. Тримісний вираз if/else. Цикли while, break, continue, pass і else. Цикл for. Правила програмування циклів.

Тема 17. Функції в Python.

Створення функції. Функції для математичних обчислень. Значення аргументів за замовчуванням. Функція як аргумент. Рекурсія. Лямбда-функції. Локальні і глобальні змінні. Вкладені функції. Функція як результат функції.

Тема 18. Робота зі списками, кортежами, множинами, словниками та текстом в Python.

Знайомство зі списками. Основні операції зі списками. Копіювання і присвоювання списків. Списки і функції. Вкладені списки. Знайомство з кортежами. Визначення множини. Створення множин. Операції із множинами. Дії зі словниками. Робота з текстовими рядками в Python.

Тема 19. Модулі і пакети в Python.

Визначення модуля. Імпорт модулів. Шлях пошуку модулів. Створення модуля. Використання модулів. Простори назв модулів.

Перезавантаження модулів. Поняття пакету. Основи імпортування пакетів. Приклад імпортування пакета. Приховування даних в модулях. Змішані режими використання: `__name__` і `__main__`. Зміна шляху пошуку модулів. Розширення `as` для інструкцій `import` і `from`. Імпорт модулів по імені у вигляді рядка. Концепції проектування модулів. Типові проблеми при роботі з модулями.

Тема 20. Програмування класів в Python.

Класи, об'єкти і екземпляри класів. Конструктор і деструктор екземпляра класу. Атрибут об'єкту класу. Додавання та видалення атрибутів і методів. Методи і функції. Копіювання екземплярів і конструктор створення копії. Спадкування. Простори назв. Класи і модулі.

Змістовий модуль 4 Програмування графічних додатків в Python.

Тема 21. Створення GUI на Python за допомогою бібліотеки Tkinter.

Графічні бібліотеки у Python. Початок роботи з Tkinter. Основні поняття. Основні віджети. Надпис (Label), кнопка команд (Button) та поле введення (Entry). Менеджери розміщення. Менеджер розміщення `pack`. Рамка (Frame). Створення та пакування елементів однією командою. Менеджер розміщення `grid`. Змінні `tkinter`. Список (Listbox) та лінійка прокрутки (Scrollbar). Метод `bind`. Події. Вікно тексту (Text). Меню (Menu). Кнопка вибору (Checkbutton), радіокнопка (Radiobutton) та рамка з заголовком (LabelFrame).

Тема 22. Налаштування елементів інтерфейсу користувача.

Створення вікон повідомлень (інформаційне вікно, вікно з попередженням, вікно з помилкою). Зміна іконки головного вікна додатку. Зміна заголовку вікна. Використання поля введення з попередньо встановленими варіантами (SpinBox). Конфігурування зовнішнього вигляду віджетів. Віджет `ProgressBar`. Використання віджету `Canvas`.

Тема 23. Робота з бібліотекою Matplotlib.

Встановлення бібліотеки `Matplotlib`, Побудова лінійного графіку. Побудова декількох графіків на одному або на різних полях. Побудова діаграми для категоріальних даних. Огляд основних елементів графіку.

4. СТРУКТУРА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

4.1 Розподіл обсягу дисципліни за семестрами:

Семестр	Всього		Розподіл за семестрами та видами занять, год.					Семестрова атестація
	Годин	Кредитів	Лекц.	Практичн.	Лаб.роб.	Курсова робота	СРС	
Повний курс								
1	180	6	30		45		105	Екзамен
2	120	4	18		36		66	Екзамен
Всього	300	10	48		81		171	
Прискорений курс								
1	180	6	30		60		90	Екзамен
Всього	180	6	30		60		90	

4.2 Розподіл обсягу дисципліни за видами навчальних занять і темами:

№ з/п	Назви змістових модулів і тем	Кількість годин				
		Усього	в т.ч.			
			Л	П(С)	Лаб	СРС
Змістовий модуль 1						
Введення в розробку і кодування алгоритмів на мові C++.						
1	Тема 1. Алгоритм як основне поняття програмування..	11	2		2	7
2	Тема 2. Основні поняття мови C++.	11	2		2	7
3	Тема 3. Введення-виведення даних у мові C++.	11	2		2	7
4	Тема 4. Умовні оператори.	11	2		2	7
5	Тема 5. Оператори циклу.	11	2		2	7
6	Тема 6. Функції користувача.	12	2		3	7
Змістовий модуль 2 Складні структури даних в C++. Об'єктно-орієнтоване програмування на мові C++.						
7	Тема 7. Масиви.	13	2		4	7
8	Тема 8. Показчики та робота з пам'яттю в C++.	14	2		4	8
9	Тема 9. Динамічні масиви у C++.	14	2		4	8

№ з/п	Назви змістових модулів і тем	Кількість годин				
		Усього	в т.ч.			
			Л	П(С)	Лаб	СРС
10	Тема 10. Робота з рядками.	14	2		4	8
11	Тема 11. Структури та об'єднання.	14	2		4	8
12	Тема 12. Робота з файлами у мові C++.	14	2		4	8
13	Тема 13. Динамічні структури даних.	15	3		4	8
14	Тема 14. Робота з класами у мові C++.	15	3		4	8
Усього за 1-й семестр		180	30		45	05
Змістовий модуль 3 Основи мови Python.						
15	Тема 15. Типи об'єктів мови Python.	11	2		2	7
16	Тема 16. Інструкції мови Python.	11	2		2	7
17	Тема 17. Функції в Python.	11	2		2	7
18	Тема 18. Робота зі списками, кортежами, множинами, словниками та текстом в Python.	13	2		4	7
19	Тема 19. Модулі і пакети в Python.	13	2		4	7
20	Тема 20. Програмування класів в Python.	13	2		4	7
Змістовий модуль 4 Програмування графічних додатків в Python.						
21	Тема 21. Створення GUI на Python за допомогою бібліотеки Tkinter.	16	2		6	8
22	Тема 22. Налаштування елементів інтерфейсу користувача.	16	2		6	8
23	Тема 23. Робота з бібліотекою Matplotlib.	16	2		6	8
Усього за 2-й семестр		120	18		36	66
Усього годин		300	48		81	171

Л – лекції, П (С) – практичні (семінарські) заняття, Лаб – лабораторні заняття, СРС – самостійна робота студентів.

5. ТЕМИ ЛАБОРАТОРНИХ ЗАНЯТЬ

№ з/п	Тема	Назва лабораторної роботи	Кількість годин
Змістовий модуль 1			
Введення в розробку і кодування алгоритмів на мові C++.			
1	Тема 1	Основи роботи у середовищі Microsoft Visual Studio.	2
2	Тема 2	Основи створення програм мовою C++.	2
3	Тема 3	Створення лінійних програм мовою C++.	2
4	Тема 4	Розробка програм з розгалуженням.	2
5	Тема 5	Створення програм з циклами мовою C++.	2
6	Тема 6	Створення програм з використанням функцій мовою C++.	3
Змістовий модуль 2			
Складні структури даних в C++.			
Об'єктно-орієнтоване програмування на мові C++.			
7	Тема 7	Створення програм обробки одновимірних масивів мовою C++.	2
8	Тема 7	Створення програм обробки двовимірних масивів мовою C++.	2
9	Тема 8	Розробка програм з використанням показчиків.	4
10	Тема 9	Робота з динамічними масивами мовою C++.	4
11	Тема 10	Створення програм обробки рядків мовою C++.	4
12	Тема 11	Організація структур на мові C++.	4
13	Тема 12	Створення програм для роботи з файлами мовою C++	4
14	Тема 13	Створення програм для роботи з динамічними структурами даних.	4
15	Тема 14	Створення програм з використанням класів на мові C++.	4
Змістовий модуль 3			
Основи мови Python.			
16	Тема 15	Змінні, значення і їх типи у мові Python	2
17	Тема 16	Створення програм з циклами та розгалуженням мовою Python.	2
18	Тема 17	Робота з функціями в Python.	2
19	Тема 18	Робота зі списками в Python.	2
20	Тема 18	Робота зі словниками в Python.	2
21	Тема 19	Використання модулів та пакетів при програмуванні мовою Python.	4

№ з/п	Тема	Назва лабораторної роботи	Кількість годин
22	Тема 20	Використання класів в програмах на мові Python.	4
Змістовий модуль 4			
Програмування графічних додатків в Python.			
23	Тема 21	Створення графічного інтерфейсу користувача(GUI) з використанням бібліотеки Tkinter.	6
24	Тема 22	Використання віджету Canvas для побудови і анімації геометричних примітивів.	6
25	Тема 23	Використання бібліотеки Matplotlib для побудови графіків функцій.	6
Усього годин			45

6. САМОСТІЙНА РОБОТА

№ з/п	Зміст самостійної роботи студента	Кількість годин
1 семестр		
1	Історія розвитку мови програмування C++	7
2	Типізація даних в C++: вбудовані, похідні типи і класи	7
3	Призначення і використання покажчиків в C++.	7
4	Масиви та їх реалізація в C++.	7
5	Операції, які використовуються в мові програмування C++.Пріоритети виконання операцій.	7
6	Конструкції мови програмування C++, що управляють обчислювальним процесом	7
7	Робота з рядками в C++	7
8	Типові математичні функції мови C++.	8
9	Стандартні потоки вводу-виводу	8
10	Файлові потоки вводу-виводу.	8
11	Динамічний розподіл і звільнення пам'яті.	8
12	Загальна характеристика динамічних структур даних.	8
13	Робота з лінійним списком	8
14	Робота з чергами та стеком	8
Усього за 1-й семестр		105
2-й семестр		
15	Завантаження та інсталяція середовища розробки. Інтерактивний та сценарний режими роботи. Оператор	7

№ з/п	Зміст самостійної роботи студента	Кількість годин
	виводу повідомлень. Різні формати повідомлень. Результати математичних операцій	
16	Змінні та їх ідентифікатори. Правила побудови ідентифікаторів. Ввід та подання інформації від користувача. Стрічкові методи. Операція поєднання стрічок. Складені оператори присвоєння. Генерування випадкових значень	7
17	Умовні конструкції. Конструкція <code>elif</code> . Цикл із логічним оператором. Вічний цикл та вихід із нього. Логічне значення арифметичного виразу. Складені логічні вирази. Перевірка реєстраційного імені та пароля.	7
18	Застосування циклу <code>for</code> . Зрізи стрічок. Кортєжі. Індксація кортежів. Зрізи кортежів.	7
19	Операції із списками. Спискові методи. Умови застосування кортежів та списків.	7
20	Оголошення функцій. Інкапсуляція функцій. Повторне використання коду. Області видимості. Глобальні змінні. Модифікація глобальних змінних. Умови використання глобальних змінних.	7
21	Способи читання інформації з текстового файлу. Порядкове читання. Запис в текстовий файл. Режими доступу до бінарних файлів. Зберігання та читання об'єктів із файлів.	8
22	Типи виключень. Аргумент виключень. Блок <code>else</code> ехсепт в конструкції з оператором <code>try</code> .	8
23	Класи. Конструктори. Параметр <code>self</code> . Параметри конструктора. Статичні методи. Декоратори. Інкапсуляція об'єктів. Непрямий доступ до атрибутів. Закриті атрибути та методи. Комунікація між об'єктами.	8
	Усього за 2-й семестр	66
Разом		171

5. МЕТОДИ НАВЧАННЯ

В навчальному процесі застосовуються: лекції з використанням мультимедіа матеріалів; лабораторні роботи в комп'ютерному класі з пошуком інформації в Інтернет та самостійна робота. Лабораторні роботи першого та третього змістовного модулів виконуються студентами індивідуально. Для виконання лабораторних робіт другого та четвертого модулів студенти об'єднуються в групи по 2-3 чоловіка з метою формування здатності працювати в команді.

6. МЕТОДИ КОНТРОЛЮ

При вивченні дисципліни використовуються наступні методи контролю: проведення поточного контролю, письмового підсумкового контролю у вигляді екзамену, модульного контролю.

Захист лабораторних робіт показує рівень теоретичного і практичного засвоєння матеріалу і слугує поточним контролем рівня підготовки студента. Розподіл балів, що отримують студенти, наведено в розділі 8.

Основним засобом контролю є модульна контрольна робота. Під час її виконання студент показує повноту засвоєння матеріалу та вміння використовувати свої знання. Тематика модульних контрольних робіт наведена в розділі 7.

7. КОНТРОЛЬНІ РОБОТИ

Контрольні роботи з теоретичної частини розподілені таким чином:

№ роботи	№ теми	Тема контрольної роботи	Кількість варіантів
1-й семестр			
1	1-7	Введення в розробку і кодування алгоритмів на мові C++.	20
2	7-14	Складні структури даних в C++. Об'єктно-орієнтоване програмування на мові C++.	20
2-й семестр			
3	15-20	Основи мови Python.	20
4	21-23	Програмування графічних додатків в Python.	20

8. КРИТЕРІЇ ОЦІНЮВАННЯ ТА РОЗПОДІЛ БАЛІВ, ЯКІ ОТРИМУЮТЬ СТУДЕНТИ

8.1. Розподіл балів, які отримують студенти (кількісні критерії оцінювання)

Складові навчальної роботи	Бали за одне заняття (завдання)	Кількість занять (завдань)	Сумарна кількість балів
1-й семестр			
Змістовий модуль 1			
Виконання і захист лабораторних робіт	0...4	6	0...24
Модульний контроль	0...16	1	0...16

Складові навчальної роботи	Бали за одне заняття (завдання)	Кількість занять (завдань)	Сумарна кількість балів
Змістовий модуль 2			
Виконання і захист лабораторних робіт	0...5	9	0...45
Модульний контроль	0...15	1	0...15
Усього за 1-й семестр			55... 100
2-й семестр			
Виконання і захист лабораторних робіт	0...5	7	0...35
Модульний контроль			0...22
Виконання і захист лабораторних робіт	0...7	3	0...21
Модульний контроль	0...22	1	0...22
Усього за 2-й семестр			55...100

8.2. Критерії оцінки знань студентів протягом семестру

8.2.1. Критерії оцінювання лабораторних робіт

Захист лабораторної роботи складається з двох етапів.

Перший етап – представлення результатів: готового програмного додатку та програмних кодів у відповідності до завдання лабораторної роботи.

Критерії оцінювання результатів роботи:

- робота була виконана у відповідності з завданням за вказаний час – студент отримує 100% від максимальної кількості балів;
- робота була виконана у відповідності з завданням с запізненням – студент отримує 50 % від максимальної кількості балів;
- завдання виконано не повністю, а на x % – студент отримує x% від балів, що мали б бути зараховані у відповідності до строків виконання.

Другий етап здачі лабораторної роботи – відповідь на контрольні питання.

Кількість питань визначає викладач за підсумками представлених результатів, але не менше ніж 2. Кожне контрольне питання оцінюється таким чином:

- повна розгорнута відповідь з прикладами та додатковим завданням, що було опрацьовано на самостійній роботі – кількість балів, що отримані на першому етапі збільшується до 25 % балів;

- повна, але не розгорнута відповідь без додаткового завдання – кількість балів не змінюється;
- неповна відповідь, або відповідь, що містить незначні та некритичні помилки чи суперечності – кількість балів, що було отримано на попередньому етапі зменшується на 25 %;
- відповідь, що містить критичні помилки, або відсутність відповіді – кількість балів, що було отримано на попередньому етапі зменшується на 50%.

8.2.2 Критерії оцінювання модульної контрольної роботи

14-16 балів виставляється, коли студент дає абсолютно правильні відповіді на теоретичні питання з викладенням оригінальних висновків, отриманих на основі програмного, додаткового матеріалу та нормативних документів. При виконанні практичного завдання студент застосовує системні знання навчального матеріалу, передбачені навчальною програмою.

12-13 балів виставляється студенту, який повністю розкрив теоретичні питання на основі програмного та додаткового матеріалу. При виконанні практичних завдань студент застосовує узагальнені знання навчального матеріалу, передбачені навчальною програмою.

10-11 балів виставляється студенту, який повністю розкрив теоретичні питання, а програмний матеріал викладено у відповідності до вимог. Практичні завдання виконані в цілому правильно, але мають місце окремі неточності.

8-9 балів виставляється, коли студент розкрив теоретичні питання, проте при викладенні програмного матеріалу допущені окремі помилки. При виконанні практичних завдань студент припускається помилок, за рахунок недостатнього розуміння програмного матеріалу.

6-7 балів виставляється, коли студент неповністю розкрив теоретичні питання, відповідь містить суттєві помилки. При виконанні практичних завдань студент припускається значних помилок, а виконання завдань викликає значні труднощі у студента.

3-5 бали виставляються студенту, який не розкрив теоретичні питання і не може виконати практичні завдання. Як правило такий студент виявляє здатність до викладення думки лише на елементарному рівні.

0-2 бали виставляється студенту, який не виконав навчальну програму або якийсь елемент її складової, має фрагментарні знання, які не дозволяють розкрити теоретичні питання і виконати практичні завдання. Такий студент не може викласти свою думку навіть на елементарному рівні.

8.3. Підсумкова оцінка за семестр.

Контроль знань здобувачів здійснюється за рейтинговою накопичувальною (100-бальною) системою, яка передбачає складання обов'язкових контрольних точок.

Підсумкова оцінка за дисципліну складається з оцінки, отриманої на протязі семестру, та оцінки, отриманої на екзамені.

Після завершення семестру проводиться семестровий екзамен, завданням якого є перевірка розуміння студентом програмного матеріалу в цілому, логіки та взаємозв'язків між окремими розділами, здатності творчого використання накопичених знань, вміння формулювати своє ставлення до певної проблеми навчальної дисципліни тощо.

Максимальна кількість балів (100) при оцінюванні знань з навчальної дисципліни, яка завершується екзаменом, формується з двох частин, з коефіцієнтом 0,5 кожна:

– за поточну успішність 100 балів (сума балів, зароблена у семестрі, але не менше 55);

– на екзамені 100 балів (мінімально необхідна кількість балів за екзамен 55).

Підсумкові оцінки за триместр в цілому переводяться за національною шкалою та шкалою ECTS відповідно до таблиці перекладу, яка визначається діючим в ДДМА положення про організацію навчального процесу в кредитно-модульній системі підготовки фахівців:

Рейтингова оцінка	У національній шкалі	У шкалі ECTS
90-100	Відмінно (зараховано)	A
81-89	Добре (зараховано)	B
75-80	Добре(зараховано)	C
65-74	Задовільно (зараховано)	D
55-64	Задовільно (зараховано)	E
30-54	Незадовільно (не зараховано)	FX
0-29	Незадовільно (не зараховано)	F

Для отримання позитивної оцінки з дисципліни студент повинен одержати не менше ніж 55 балів сумарної оцінки.

8. НАВЧАЛЬНО-МЕТОДИЧНІ МАТЕРІАЛИ

1. Комп'ютерні технології та програмування. Конспект лекцій (для студентів спеціальності 123 «Комп'ютерна інженерія»).

2. Методичні вказівки до виконання лабораторних робіт з дисципліни ” Комп'ютерні технології та програмування” (для студентів спеціальності 123 «Комп'ютерна інженерія»).

9. РЕКОМЕНДОВАНА ЛІТЕРАТУРА

Література основна

1. Slobodan Dmitrović. Modern C++ for Absolute Beginners: A Friendly Introduction to C++ Programming Language and C++11 to C++20 Standards. - Apress, 2020. – 751 p.
2. John Keyser. Introduction to C++: Programming Concepts and Applications. - The Teaching Company, 2019. – 313 p.
3. Страуструп, Б. Язык программирования C++. Краткий курс. – СПб: ООО «Диалектика», 2020 – 315 с.
4. Страуструп, Б. Программирование: принципы и практика с использованием C++. – М.: ООО «И.Д. Вильямс», 2016. – 1328с.
5. Stroustrup, B. The C++ Programming Language, 4th Edition. - Addison-Wesley, 2013. – 1368 p.
6. Прата, С. Язык программирования C++. Лекции и упражнения, 6-е изд. : Пер.с англ. - М.: ООО «И.Д. Вильямс», 2014. – 1244с.
7. Олексій Васильєв. Програмування мовою Python. - Київ: Навчальна книга Богдан, 2019 - 504 стр.
8. Gaddis, Tony. Starting out with Python, 3th Edition. – Pearson, 2015. – 635p.
9. Lubanovic, Bill. Introducing Python: modern computing in simple packages. - O'Reilly, 2019. – 935p.
10. Meier, B. Python GUI Programming Cookbook: Develop functional and responsive user interfaces with tkinter and PyQt5. - Birmingham, UK:Packt Publishing, 2019. – 486p.
11. Мэтиз Э. Изучаем Python. Программирование игр, визуализация данных, веб-приложения. — СПб.: Питер, 2017. — 496 с.
12. Павловская, Т.А. Программирование на языке C++. – М: НОУ «Интуит», 2016. – 155с.
13. Павловская, Т. А., Щупак. Ю. А. C/C++. Структурное и объектно-ориентированное программирование: практикум. — СПб.: Питер, 2011. — 352 с.

Література додаткова

1. S Martin, Xavier. C++ Programming: Complete Guide to Learn the Basics of C++ Programming in 7 Days. – 2020. – 174 p.
2. Сиддхартха Рао. Освой самостоятельно C++ за 21 день, 7-е изд. - М.: ООО «И.Д. Вильямс», 2013. – 752 с.
3. С. Шапошникова. Основы программирования на Python. Учебник. Вводный курс. — версия 2. — 2011. — 44 с.
4. И. А. Хахаев. Практикум по алгоритмизации и программированию на Python. Учебник —М.: Альт Линукс, 2010. — 126 с.

Інформаційні ресурси в мережі Інтернет

1. <http://cpp.dp.ua/lecture/>
2. <https://uk.wikipedia.org/wiki/C%2B%2B>
3. https://ru.wikipedia.org/wiki/Python_Software_Foundation
4. <https://docs.python.org/3/>
5. <https://docs.python.org/3/using/index.html>

Додаток А

Питання для підготовки до контрольної роботи та екзамену з дисципліни «Комп'ютерні технології та програмування»

1. Структурний підхід до програмування.
2. Алгоритми.
3. Основні структурні блоки програми на C++.
4. Лексичні поняття C++ (алфавіт, типи лексем)
5. Типи даних (прості, похідні, складені).
6. Область видимості та час життя змінної.
7. Оголошення, визначення та ініціалізація змінних.
8. Правило одного визначення.
9. Вирази та операції (арифметичні, логічні, відношення).
10. Посилання.
11. Оголошення та визначення функції.
12. Прототип.
13. Передача параметрів в функції за замовчуванням.
14. Передача аргументів в функцію за значенням та за посиланням.
15. Перевантаження функцій.
16. Вбудовані функції (inline).
17. Керуючі конструкції (умовні оператори та цикли).
18. Директиви препроцесора і вказівки компілятору.
19. Структуровані типи даних: масиви.
20. Доступ до елементів масиву через індексні вирази та через вказівники.
21. Вказівники. Операції над вказівниками.
22. Обробка символічної інформації. Рядки
23. Структуровані типи даних: структури, об'єднання, засіб typedef, перелічення.
24. Динамічне керування пам'яттю.
25. Моделі пам'яті.
26. Введення/виведення в стандартний потік.
27. Доступ до файлів за допомогою потоку введення/виведення (функції високого рівня).
28. Компілятор і інтерпретатор. Достоїнства і недоліки.
29. Назвіть і дайте коротку характеристику основних класів язи-ков програмування.
30. Вбудовані числові типи мови Python.
31. Списки. Створення, основні операції.
32. Основні методи списку.
33. Кортежі. Створення, основні методи і операції.
34. Словники. Створення, основні операції.
35. Методи для роботи зі словниками.
36. Безліч. Створення, основні методи і операції.
37. Змінні. Правила іменування змінних.

38. Динамічна типізація.
39. Оператори порівняння і логічні оператори.
40. Інструкція `if ... else`.
41. Інструкція циклу `while`.
42. Інструкція циклу `for`.
43. Створення і виклик функції.
44. Передача аргументів функцію.
45. Функції-генератори.
46. Лямбда-функції.
47. Модулі. Інструкції `import` і `from`.
48. Базові принципи об'єктно-орієнтованого програмування.
49. Клас, метод класу, атрибут класу. Визначення класу і створення екземпляра класу.
50. Конструктор і деструктор.
51. Спадкування.
52. Абстрактні методи класу.
53. Статичні методи класу.
54. Властивості класу.
55. Винятки. Обробка винятків.
56. Призначені для користувача винятку.
57. Подія. Обробник події. Цикл обробки подій.
58. Елемент Кнопка. Створення та налагодження.
59. Елемент Кнопка. Створення обробника події.
60. Елементи Напис і Текстове поле. Створення та налагодження. Метод `get` ().
61. Елемент Прапорець. Створення, налаштування, отримання статусу прапорця.
62. Елемент перемикач. Створення, налаштування, доступ до значення.
63. Класи `date`, `time` і `datetime`.