

**ДЕРЖАВНИЙ ВИЩИЙ НАВЧАЛЬНИЙ ЗАКЛАД
“УЖГОРОДСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ”
ІНЖЕНЕРНО-ТЕХНІЧНИЙ ФАКУЛЬТЕТ
КАФЕДРА КОМП’ЮТЕРНИХ СИСТЕМ ТА МЕРЕЖ**

ЗАТВЕРДЖУЮ

**Декан інженерно-технічного
факультету**

доц. Йолана ГОЛИК

“ 14.06.2025 ” 2025 р.

**РОБОЧА ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ
СИСТЕМНЕ ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ**

Рівень вищої освіти – перший (бакалавр)

Галузь знань – 12 – інформаційні технології

Спеціальність 123 – комп’ютерна інженерія

Освітня програма - “комп’ютерні системи та мережі”

Статус дисципліни – обов’язкова

Мова навчання – українська

Ужгород – 2025

Робоча програма навчальної дисципліни Системне програмне забезпечення для здобувачів спеціальності 123 – «Комп'ютерна інженерія» освітньої програми «Комп'ютерні системи та мережі» – 18 с.

Розробник:

Пойда В.Ю., канд. фіз.-мат. наук, доцент кафедри комп'ютерних систем та мереж,

Робочу програму розглянуто та затверджено на засіданні кафедри комп'ютерних систем та мереж

протокол № 13 від «25» червня _____ 2025 р.

Завідувач кафедри _____ доц. Горват П.П.
(підпис) (прізвище та ініціали)

Схвалено науково-методичною комісією інженерно-технічного факультету

протокол № 6 від «27» червня _____ 2025 р.

Голова науково-методичної комісії _____ Володимир ЦИГИКА
(підпис) (прізвище та ініціали)

1. ОПИС НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

Найменування показників	Розподіл годин за навчальним планом
	денна форма навчання
Кількість кредитів ЄКТС – 7,0	Рік підготовки:
Загальна кількість годин – 210	2,3-й
Кількість модулів – 4	Семестри
	4,5-й
Тижневих годин для денної форми навчання: аудиторних – 2,6/2,6 години самостійної роботи студента – 2,8/4,3 години	Лекції
	24+24 год
	Практичні (семінарські)
	-
Вид підсумкового контролю: екзамен/залік	Лабораторні
	20+20 год
Форма підсумкового контролю: усна	Самостійна робота
	46+46 год

2. МЕТА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

Мета вивчення навчальної дисципліни “Системне програмне забезпечення” - отримання студентами поглиблених знань з програмування. Отримані знання з даної дисципліни дозволять застосовувати класичні та сучасні методології програмування в практичній роботі. Якісно планувати та виконувати роботи по створенню програмного забезпечення. Майбутній спеціаліст може застосовувати знання, які отримав при вивченні цієї дисципліни, як при подальшому навчанні, так і після отримання освітньо-кваліфікаційного рівня – бакалавр у своїй професійній діяльності.

Завдання дисципліни – формувати теоретичні знання та практичні навички у майбутніх фахівців відповідно до поставленої мети.

В результаті вивчення дисципліни студент повинен:

- Засвоїти основні поняття з системного програмного забезпечення. Вміти застосовувати набуті знання при розробці ПЗ.
- Уміти планувати виконання робіт по створенню програмного забезпечення.

Відповідно до освітньої програми, вивчення дисципліни сприяє формуванню у здобувачів вищої освіти таких компетентностей:

- інтегральна (здатність розв’язувати складні задачі і проблеми у певній галузі професійної діяльності або у процесі навчання що передбачає проведення досліджень та/або здійснення інновацій та характеризується невизначеністю умов і вимог);
- загальні ((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; (ЗК10) здатність зберігати та примножувати моральні, культурні, наукові цінності і досягнення суспільства на основі розуміння історії та закономірностей розвитку предметної області, її місця у загальній системі знань про природу і суспільство та у розвитку суспільства, техніки і технологій, використовувати різні види та форми рухової активності для активного відпочинку та ведення здорового способу життя)
- фахові ((ФК2) здатність використовувати сучасні методи і мови програмування для розроблення алгоритмічного та програмного забезпечення; (ФК3) здатність створювати системне та прикладне програмне забезпечення комп’ютерних систем та мереж; (ФК11) здатність оформляти отримані робочі результати у вигляді презентацій, науково-технічних звітів; (ФК15) здатність аргументувати вибір методів розв’язування спеціалізованих задач, критично оцінювати отримані результати, обґрунтовувати та захищати прийняті рішення);

3. ПЕРЕДУМОВИ ДЛЯ ВИВЧЕННЯ НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

Передумовами вивчення навчальної дисципліни “ Системне програмне забезпечення” є опанування курсів: (ОК14) “Програмування”, (ОК16) “Структури даних та алгоритми”, (ОК19) “Об’єктно-орієнтоване програмування” та інших нормативних дисциплін спеціальності.

4. ОЧІКУВАНІ РЕЗУЛЬТАТИ НАВЧАННЯ

Відповідно до освітньої програми «Комп’ютерні системи та мережі», вивчення навчальної дисципліни повинно забезпечити досягнення здобувачами вищої освіти таких програмних результатів навчання (ПРН):

Програмні результати навчання	Шифр ПРН
Знати і розуміти наукові положення, що лежать в основі функціонування комп’ютерних засобів, систем та мереж.	ПРН1
Вміти системно мислити та застосовувати творчі здібності до формування нових ідей.	ПРН8
Вміти здійснювати пошук інформації в різних джерелах для розв’язання задач комп’ютерної інженерії.	ПРН11
Вміти поєднувати теорію і практику, а також приймати рішення та виробляти стратегію діяльності для вирішення завдань спеціальності з урахуванням загальнолюдських цінностей, суспільних, державних та виробничих інтересів.	ПРН14
Вміти оцінювати отримані результати та аргументовано захищати прийняті рішення.	ПРН16

Очікувані результати навчання, які повинні бути досягнуті здобувачами освіти після опанування навчальної дисципліни „Системне програмне забезпечення”:

Очікувані результати навчання з дисципліни	Шифр ПРН
Мати чітке представлення про основні наукові положення, що лежать в основі функціонування комп’ютерних систем.	ПРН1
Вміння системно мислити та застосовувати творчі здібності до формування нових ідей.	ПРН8
Вміння здійснювати пошук інформації в різних джерелах для розв’язання поставлених задач.	ПРН11
Вміння поєднувати теорію і практику, а також приймати рішення та виробляти стратегію діяльності для вирішення завдань спеціальності з урахуванням загальнолюдських цінностей,	ПРН14

суспільних, державних та виробничих інтересів.	
Вміння оцінювати отримані результати та аргументовано захищати прийняті рішення.	ПРН16

5. ЗАСОБИ ДІАГНОСТИКИ ТА КРИТЕРІЇ ОЦІНЮВАННЯ РЕЗУЛЬТАТІВ НАВЧАННЯ

Робоча програма з дисципліни “Системне програмне забезпечення”, що читається на другому і третьому курсах бакалаврату ІТФ спеціальності комп’ютерна інженерія, містить чотири модулі, які складаються із вісьмох змістовних модулів. Використовуються методи усного та письмового контролю. Поточний контроль передбачає: опитування студентів під час захисту лабораторних робіт та опитування на лекціях. Програмою передбачена самостійна робота студентів та контроль за нею у формі публічного захисту. Приводиться список основної та допоміжної літератури, яка рекомендується для вивчення цієї дисципліни. Підсумковий контроль передбачає екзамени у кінці першого, та залік у кінці другого семестрів.

Для контролю знань розроблено: перелік теоретичних питань та типових завдань, (наведено в додатку); завдання для самостійної роботи, зі змістом яких студенти ознайомлюються на початку семестру.

Оцінка ECTS, яку студент отримує після вивчення кредитного модуля дисципліни, визначається відповідно до рейтингу студента. Рейтинг студента складається з балів, що він отримує протягом кожного із семестрів за такі види робіт:

1. Модульні контрольні роботи (МКР) тривалістю по 2 акад. години. Максимальна кількість балів за дві МКР (за семестр) – 40 (по 20 балів кожна).
2. Виконання лабораторних робіт. Протягом вивчення дисципліни студенти виконують 9 лабораторних робіт (4 за перший семестр, та 5 – за другий). Максимальна кількість балів за семестрові лабораторні роботи – 20.
3. Бали із індивідуальної та самостійної роботи студентів нараховуються за дослідницьку роботу, підготовку рефератів та презентацій, за творчий підхід до виконання завдань лабораторних робіт: 0-20 балів за семестр.
4. Активна робота студента на заняттях оцінюється в діапазоні 0-20 балів за семестр.

Модулі в сумі за семестр оцінюються **максимальним** балом 100.

Необхідною умовою допуску до семестрового екзамену є відсутність заборгованостей з лабораторних робіт та зарахування МКР, з сумарним поточним балом не менше 35.

Розподіл балів, які студенти отримують за модулі наведені в таблицях.

Розподіл балів, які отримують здобувачі вищої освіти (модуль 1)

Поточне тестування (опитування)						Самостійна робота	Письмова контрольн а робота	Сума
Змістовний модуль 1				Змістовний модуль 2		10	20	50
T1	T2	T3	T4	T1	T2			
3	3	3	3	4	4			

Розподіл балів, які отримують здобувачі вищої освіти (модуль 2)

Поточне тестування (опитування)						Самостійна робота	Письмова контрольна робота	Сума
Змістовний модуль 3				Змістовний модуль 4		10	20	50
T1		T2		T1	T2			
5		5		5	5			

Розподіл балів, які отримують здобувачі вищої освіти (модуль 3)

Поточне тестування (опитування)						Самостійна робота	Письмова контрольн а робота	Сума
Змістовний модуль 1				Змістовний модуль 2		10	20	50
T1	T2	T3		T1	T2			
4	4	4		4	4			

Розподіл балів, які отримують здобувачі вищої освіти (модуль 4)

Поточне тестування (опитування)						Самостійна робота	Письмова контрольн а робота	Сума
Змістовний модуль 1				Змістовний модуль 2		10	20	50
T1	T2	T3		T1	T2			
4	4	4		4	4			

Оцінювання окремих видів навчальної роботи з дисципліни

Вид діяльності здобувача вищої освіти	Модулі 1-2	
	Кількість	Максимальна кількість балів (сумарна)
Практичні (семінарські) заняття	-	-
Захист самостійної роботи	1	20
Лабораторні заняття (допуск, виконання та захист)	2	20
Комп'ютерне тестування при тематичному оцінюванні	-	-
Письмове тестування при тематичному оцінюванні	4	20
Модульні контрольні роботи за семестр	2	40
Разом за семестр		100

Критерії оцінювання модульних контрольних робіт

Модульна контрольна робота 1 (МКР1), містить 10 тестових питань. Кожне питання оцінюється у 2 бали. Сумарний бал за контрольну роботу рівний 20.

Модульна контрольна робота 2 (МКР2), складається з 5 задач, які охоплюють усі теми модуля. Контрольну роботу студенти виконують на комп'ютерах. Кожна задача оцінюється у 5 балів. Сумарний бал за контрольну роботу рівний 20. Таким чином бали нараховуються з надлишковістю.

Модульна контрольна робота 3 (МКР3), містить 5 питань. Усі питання короткі і потребують короткі, але змістовні відповіді. Кожне питання оцінюється у 4 бали. За оригінальні відповіді студент може отримати ще 2 бали. Сумарний бал за контрольну роботу рівний 20.

Модульна контрольна робота 4 (МКР4), складається з 5 питань. Кожне питання оцінюється у 4 бали. Сумарний бал за контрольну роботу рівний 20.

Критерії оцінювання підсумкового семестрового контролю

До складання екзамену допускаються лише студенти, які мають рейтинговий бал не менше 35. Екзамен з навчальної дисципліни студент

може не складати, якщо він склав усі модулі, з рейтинговим балом не менше 60, та його влаштовує рейтингова оцінка. Студенти, які мають рейтинговий бал від 35 до 59 екзамен складають обов'язково. Студент може підвищити на екзамені оцінку, при цьому рейтингова оцінка не може бути зменшена.

За результатами виконання студентом навчальної програми впродовж семестру рекомендується виставляти заліки та екзамени без додаткового опитування за такою шкалою:

Сумарні бали	Оцінка ECTS	Екзамен (диф. залік)	Залік
90 – 100	A	Відмінно	Зараховано
82 – 89	B	Добре	
74 – 81	C		
64 – 73	D	Задовільно	
60 – 63	E		
35 – 59	FX	Незадовільно з можливістю повторного складання	Незараховано з можливістю повторного складання
1 – 34	F	Незадовільно з обов’язковим повторним вивченням дисципліни	Незараховано з обов’язковим повторним вивченням дисципліни

6. ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

6.1. Зміст навчальної дисципліни

Змістовний модуль 1. Послідовні контейнери.

Тема 1. Вступ в системне програмне забезпечення. Місце дисципліни серед інших навчальних дисциплін. Бібліотека STL. Послідовні контейнери.

Тема 2. Вектори. Опис та наповнення вектора. Операції над векторами. Функції визначення розміру вектора. Сортування вектора.

Тема 3. Списки. Опис та наповнення списку. Операції над списками. Сортування списку.

Тема 4. Черги. Опис та наповнення черги. Операції над чергами. Функції визначення розміру черги.

Змістовний модуль 2. Немодифікуючі та модифікуючі алгоритми.

Тема 1. Немодифікуючі алгоритми. Знаходження інформації про послідовності. Лямбда-функції у немодифікуючих алгоритмах. Часові дослідження немодифікуючих алгоритмів.

Тема 2. Модифікуючі алгоритми. Базові модифікуючі алгоритми. Алгоритми перетворення однієї послідовності. Алгоритми копіювання. Алгоритми перетворення двох послідовностей.

Змістовний модуль 3. Алгоритми сортування.

Тема 1. Алгоритми сортування. Повне та часткове сортування контейнерів. Лямбда-функції у алгоритмах сортування. Стійке сортування. Часові дослідження алгоритмів сортування.

Тема 2. Алгоритми, зв'язані із сортування. Алгоритми пошуку. Лексикографічні алгоритми. Алгоритми злиття.

Змістовний модуль 4. Множини, піраміди, асоціативні контейнери.

Тема 1. Алгоритми роботи з множинами та пірамідами. Алгоритми виконання операцій над множинами. Перетворення послідовності у піраміду. Додавання та вилучення елементів з піраміди. Сортування піраміди.

Тема 2. Асоціативні контейнери. Словники та словники з дублікатами. Множини та множини з дублікатами. Бітові множини.

Змістовний модуль 5. Патерни в задачах проектування.

Тема 1. Вступ в патерни. Історія патернів. Класифікація патернів.

Тема 2. Патерни в задачах проектування. Ступінь деталізації. Нотації від ОМТ.

Тема 3. Проектування з врахуванням майбутніх змін. Типові причини перепроєктування.

Змістовний модуль 6. Породжуючі патерни.

Тема 1. Породжуючі патерни. Теми актуальні для породжуючи патернів.

Тема 2. Патерн будівельник. Призначення та мотивація патерна. Застосовність та структура патерна. Діаграма взаємодій.

Змістовний модуль 7. Структурні патерни.

Тема 1. Структурні патерни. Патерни рівня класу. Патерни рівня об'єкту.

Тема 2. Патерн адаптер. Призначення та мотивація патерна. Застосовність та структура патерна. Діаграма взаємодій.

Тема 3. Патерн компоновщик. Призначення та мотивація патерна. Застосовність та структура патерна. Діаграма взаємодій.

Змістовний модуль 8. Поведінкові патерни.

Тема 1. Поведінкові патерни.

Тема 2. Патерн ланцюг обов'язків. Призначення та мотивація патерна. Застосовність та структура патерна. Діаграма взаємодій.

6.2. СТРУКТУРА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

Назви змістовних модулів і тем	Кількість годин					
	Денна форма					
	Усього	у тому числі				
		лекції	практичні	лабораторні	індивідуальна робота	самостійна робота
1	2	3	4	5	6	7
Модуль 1						
Змістовний модуль 1. Послідовні контейнери						
Тема 1. Вступ в системне програмне забезпечення	4	2				2
Тема 2. Вектори	4	2				2
Тема 3. Списки	4	2				2
Тема 4. Черги	4	2				2
Разом за змістовним модулем 1	16	8				8
Змістовний модуль 2. Немодифікуючі та модифікуючі алгоритми						
Тема 1. Немодифікуючі алгоритми	14	2		4		8
Тема 2. Модифікуючі алгоритми	14	4		4		6
Разом за змістовним модулем 2	28	6		8		14
Усього за модуль 1	44	14		8		22
Модуль 2						
Змістовний модуль 3. Алгоритми сортування						
Тема 1. Алгоритми сортування	10	2		3		5
Тема 2. Алгоритми, зв'язані із сортування	12	2		3		7
Разом за змістовним модулем 3	22	4		6		12
Змістовний модуль 4. Множини, піраміди, асоціативні контейнери						
Тема 1. Алгоритми роботи з множинами та пірамідами	12	4		3		5
Тема 2. Асоціативні контейнери	12	2		3		7
Разом за змістовним модулем 4	24	6		6		12
Усього за модуль 2	46	10		12		24
Усього за семестр 1	90	24		20		46
Модуль 3						
Змістовний модуль 5. Патерни в задачах проектування						
Тема 1. Вступ в патерни	4	2				2
Тема 2. Патерни в задачах проектування	8	2				6
Тема 3. Проектування з врахуванням	8	2				6

майбутніх змін						
Разом за змістовним модулем 5	20	6				14
Змістовний модуль 6. Породжуючі патерни						
Тема 1. Породжуючі патерни	20	4		4	5	7
Тема 2. Патерн будівельник	20	2		4	5	9
Разом за змістовним модулем 6	40	6		8	10	16
Усього за модуль 3	60	12		8	10	30
Модуль 4						
Змістовний модуль 7. Структурні патерни						
Тема 1. Структурні патерни	14	2		2	4	6
Тема 2. Патерн адаптер	8	2		2	3	1
Тема 3. Патерн компоновщик	8	2		2	3	1
Разом за змістовним модулем 7	30	6		6	10	8
Змістовний модуль 8. Поведінкові патерни						
Тема 1. Поведінкові патерни	15	4		2	5	4
Тема 2. Патерн ланцюг обов'язків	15	2		4	5	4
Разом за змістовним модулем 8	30	6		6	10	8
Усього за модуль 4	60	12		12	20	16
Усього за семестр 2	120	24		20	30	46
Усього за курс	210	48		40	30	92

6.3. Темати лабораторних занять

№ з/п	Назва теми	Кількість годин
	Модуль 1	
1	Немодифікуючі алгоритми в послідовних контейнерах	4
2	Модифікуючі алгоритми в послідовних контейнерах	4
	Разом за модуль 1	8
	Модуль 2	
3	Алгоритми сортування	6
4	Множинні та пірамідальні алгоритми	6
	Разом за модуль 2	12
	Разом за семестр 1	20
	Модуль 3	
1	Патерн Абстрактна фабрика	4
2	Патерн Фабричний метод	4
	Разом за модуль 3	8
	Модуль 4	
3	Патерн Міст	4
4	Патерн Пристосуванець	4
5	Патерн Ланцюг обов'язків	4
	Разом за модуль 4	12
	Разом за семестр 2	20
	Усього за курс	40

6.4. Самостійна робота

№ з/п	Назва теми	Кількість годин
1	Вступ в системне програмне забезпечення	2
2	Вектори	2
3	Списки	2
4	Черги	2
5	Немодифікуючі алгоритми	8
6	Модифікуючі алгоритми	6
7	Алгоритми сортування	12
8	Алгоритми роботи з множинами та пірамідами	5
9	Асоціативні контейнери	7
	Разом за 1-й семестр	46
10	Патерни в задачах проектування	8
11	Проектування з врахуванням майбутніх змін	6
12	Породжуючі патерни	7
13	Патерн будівельник	9
14	Структурні патерни	6
15	Патерн адаптер	1
16	Патерн компоновщик	1
17	Поведінкові патерни	4
18	Патерн ланцюг обов'язків	4
	Разом за 2-й семестр	46
	Разом	92

6.5. Індивідуальні завдання

Підготовка рефератів та презентацій на задані теми (в тому числі і винесені на самостійне вивчення) та виконання контрольних і лабораторних робіт. У другому семестрі студенти працюють над курсовим проектом. Для виконання курсового проекту відводиться 30 годин індивідуальної роботи. Теми проектів студенти пропонують самі і погоджують з викладачем.

7. ІНСТРУМЕНТИ, ОБЛАДНАННЯ ТА ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ, ВИКОРИСТАННЯ ЯКИХ ПЕРЕДБАЧАЄ НАВЧАЛЬНА ДИСЦИПЛІНА

При виконанні студентами лабораторних та самостійних робіт використовуються звичайні персональні комп'ютери з операційною системою Windows-XP і вище та Microsoft Office.

При виконанні лабораторних робіт використовується інтегроване середовище розробки програмного забезпечення Microsoft Visual Studio.

При організації дистанційного навчання використовуються системи Moodle та Google Meet. Лектор, також, використовує графічний планшет.

8. РЕКОМЕНДОВАНІ ДЖЕРЕЛА ІНФОРМАЦІЇ

Основна література

1. **Galowicz J.** C++17 STL Cookbook: Discover the latest enhancements to functional programming and lambda expressions. – 20107. – 532 p.
2. **Gamma E., Helm R., Johnson R., Vlissides J.** Design Patterns: Elements of Reusable Object-Oriented Softwar. Addison-Wesley Professional. – 1995. – 396 p.
3. **Будай Андрій.** Дизайн-патерни – просто, як двері. 2012. - 93с.

Додаткова

1. **Herbert Schildt.** Teach Yourself C++. Osborne McGraw-Hill.
2. **Stephen Prata.** C++ Primer Plus. Indianapolis, Indiana, 46240 USA. – 792 p.
3. **Robert Lafore.** Object-Oriented Programming in C++. Sams Publishing. – 2001. – 1040 p.
4. **Фримен Е., Робсон Е., Бейтс Б.** Патерни проєктування. Вид. Фабула. 2020. - 672 с.

Додаток 1.

Варіант модульної контрольної роботи 1

1. Функція додає елемент в кінець вектора.

А	Б	В	Г	Д
<code>push_back</code>	<code>pop_back</code>	<code>push_end</code>	<code>erase</code>	<code>insert</code>

2. `vector<int> v;`
`for (int i=0; i < 10;i++) v.push_back(1);`. Далі розмір вектора **v** визначений функціями **size** та **capacity**

А	Б	В	Г	Д
10 та 10	10 та 11	10 та 12	10 та 13	10 та 14

3. `vector<int> v(101,2);`. Сума елементів вектора **v** рівна

А	Б	В	Г	Д
0	2	101	202	103

4. Функція знаходження першого входження значення в послідовність

А	Б	В	Г	Д
<code>find</code>	<code>adjacent_find</code>	<code>find_first_of</code>	<code>find_if</code>	<code>find_end</code>

5. Для послідовності **v: 2 3 2 3 2 3 1 3 2 3**, функція **count(v.begin()+3,v.end(),3)** поверне

А	Б	В	Г	Д
0	1	2	3	4

6. Для послідовності **v: 3 3 5 5 2 2 1 1 4 4**, функція ***adjacent_find(v.begin(),v.end())** поверне

А	Б	В	Г	Д
1	2	3	4	5

7. Для послідовності **v: 3 7 5 8 2 9 1 0 4 6**, вираз **find(v.begin(),v.end(),5)-v.begin()** поверне

А	Б	В	Г	Д
1	2	3	4	5

8. Для послідовності **v: 3 7 5 8 2 9 1 0 4 6**, вираз **find_if(v.begin(),v.end(),[](int a){return a>5 && a<7;})-v.begin()** поверне

А	Б	В	Г	Д
1	3	5	7	9

9. Яким з алгоритмів можна перетворити послідовність **3 7 5 8 -> 4 8 6 7**

А	Б	В	Г	Д
for_each	count_if	search	mismatch	find_if

10. Для послідовностей **v: 3 7 5 8**, **t: 3 7 15 8**, вираз **mismatch(v.begin(),v.end(),t.begin()).first-v.begin()** поверне

А	Б	В	Г	Д
0	1	2	3	4

Варіант модульної контрольної роботи 2

Задача А1. Замінити елементи першої чверті черги на куб їхнього індексу.

Використати алгоритм **generate**

Тест: 3 4 7 5 8 6 7 9 -> 0 1 7 5 8 6 7 9

Задача В1. Відсортувати елементи вектора. Окремо першу третину у порядку зростання, другу – у порядку спадання, та третю – у порядку зростання.

Використати алгоритм **sort**

Тест: 3 7 5 1 4 6 8 2 0 -> 3 5 7 6 4 1 0 2 8

Задача C1. Злити три відсортовані масиви, з елементів типу **char**, у один масив. Використати два рази алгоритм **merge**

Тест: a c e, b d f, b d g -> a b b c d d e f g

Задача D1. Знайти перетин трьох векторів, з елементів типу **char**. Використати два рази алгоритм **set_intersection**

Тест: d g a b, b f d g, d b g h -> b d g

Задача E1. Наповнити словник **map** елементами типу **char int**, Вивести словник на екран. Словник **m**: a 8, c 6, d 5, e 4, f 3, g 2, h 1, i 0. Вилучити з словника функцією **erase** початковий, та кінцевий елементи. Знову вивести словник на екран.

Тест: a 8, c 6, d 5, e 4, f 3, g 2, h 1, i 0
-> c 6, d 5, e 4, f 3, g 2, h 1

Теоретичні питання до модуля 3

1. Історія патернів.
2. Класифікація патернів.
3. Каталог патернів проектування.
4. Патерни в задачах проектування.
5. Нотації від ОМТ (Буч).
6. Відношення між класами.
7. Проектування з врахуванням майбутніх змін.
8. Типові причини перепроєктування.
9. Каркаси додатків.
10. Три суттєві різниці між каркасами і патернами.
- 11.Породжуючі патерни проектування.**
- 12.Абстрактна фабрика.
- 13.Структура абстрактної фабрики.
- 14.Учасники абстрактної фабрики.
- 15.Патерн будівельник.

- 16.Доцільність застосування патерна будівельник.
- 17.Структура патерна будівельник.
- 18.Учасники патерна будівельник.
- 19.Плюси і мінуси патерна будівельник.

Теоретичні питання до модуля 4

- 1. Структурні патерни проектування.**
2. Структурні патерни рівня класу.
3. Структурні патерни рівня об'єкта.
4. *Структурний патерн адаптер.*
5. Мотивації патерна адаптер.
6. Структура патерна адаптер.
7. Учасники патерна адаптер.
8. *Структурний патерн компоновщик.*
9. Мотивації патерна компоновщик.
10. Структура патерна компоновщик.
11. Учасники патерна компоновщик.
12. Результати патерна компоновщик.
13. *Структурний патерн міст.*
14. *Структурний патерн декоратор.*
15. *Структурний патерн фасад.*
16. *Структурний патерн легковик.*
- 17. Поведінкові патерни проектування.**
18. Поведінкові патерни рівня об'єктів.
19. *Патерн ланцюг обов'язків.*
20. Мотивації патерна ланцюг обов'язків.
21. Структура патерна ланцюг обов'язків.
22. Учасники патерна ланцюг обов'язків.
23. *Поведінковий патерн команда.*
24. *Поведінковий патерн ітератор.*
25. *Поведінковий патерн стан.*

Додаток 2.

Приклади тем курсових проектів

1. Програмна реалізація каталогу взуття на основі шаблонів проектування.
2. Програмна реалізація каталогу одягу на основі шаблонів проектування.
3. Розробка *frontend* частини для сервісу по замовленню мінеральної води, на основі шаблонів проектування.

4. Реалізація прототипу інтернет-магазину на основі шаблонів проектування.
5. Реалізація веб-додатку “клавіатурний тренажер для веб-розробників”, на основі шаблонів проектування.
6. Розробка моделі магазину меблів на основі шаблонів проектування.
7. Реалізація програмного забезпечення для ведення обліку автотранспорту компанії на основі шаблонів проектування.