





## 1. Опис дисципліни

<b>Анотація дисципліни (Призначення навчальної дисципліни)</b>	Вивченням навчальної дисципліни є технології об'єктно-орієнтованого програмування, об'єктно-випадкового моделювання систем, що необхідні для вирішення поставлених задач фахового спрямування
<b>Мета вивчення</b>	Вивчення теоретичних відомостей та опанування практичних навичок з об'єктно-орієнтованого підходу (ООП) до програмування.
<b>Завдання навчальної дисципліни</b>	Ознайомлення з теоретичними основами ООП; отримання навичок використання базових механізмів наслідування, інкапсуляції та поліморфізму; вивчення інструментарію мови програмування С++ при підході ООП
<b>Пререквізити</b>	Вивчення дисципліни "Об'єктно-орієнтоване програмування" базується на знаннях, отриманих з курсу «Основи програмування» та «Алгоритми та структури даних», під час опанування дисципліни вона взаємодіє з такими дисциплінами, як «Бази даних», а знання, набуті в процесі цього вивчення, використовуються під час опанування наступних дисциплін: «.Net програмування», «Web-програмування», «Мова програмування JAVA», та впродовж написання курсових проєктів та кваліфікаційної роботи
<b>Результати навчання</b>	ПР1 - Аналізувати, цілеспрямовано шукати і вибирати необхідні для вирішення професійних завдань інформаційно-довідникові ресурси і знання з урахуванням сучасних досягнень науки і техніки. ПР4 - Знати і застосовувати професійні стандарти і інші нормативно- правові документи в галузі інженерії програмного забезпечення. ПР5 - Знати і застосовувати відповідні математичні поняття, методи доменного, системного і об'єктно-орієнтованого аналізу та математичного моделювання для розробки програмного забезпечення. ПР7 – Знати і застосовувати на практиці фундаментальні концепції, парадигми і основні принципи функціонування мовних, інструментальних і обчислювальних засобів інженерії програмного забезпечення. ПР10 - Проводити передпроектне обстеження предметної області, системний аналіз об'єкта проектування. ПР11 - Вибирати вихідні дані для проектування, керуючись формальними методами опису вимог та моделювання. ПР12 - Застосовувати на практиці ефективні підходи щодо проектування програмного забезпечення. ПР13 - Знати і застосовувати методи розробки алгоритмів, конструювання програмного забезпечення та структур даних і знань. ПР23 – Вміти документувати та презентувати результати розробки програмного забезпечення.

## 2. Обсяг та ознаки навчальної дисципліни

Найменування показників	Галузь знань, спеціальність, рівень вищої освіти	Характеристика навчальної дисципліни	
		денна форма навчання	заочна та дистанційна форма навчання
Кількість кредитів – 7	<b>Галузь знань:</b> 12 «Інформаційні технології»  <b>Спеціальність:</b> 121 «Інженерія програмного забезпечення»;  <b>Рівень вищої освіти:</b> перший (бакалаврський)	<i>Обов'язкова</i>	
Модулів – 2		Рік підготовки	
Змістових модулів – 8		2-й	2-й
Індивідуальне науково-дослідне завдання :		Семестр	
Курсовий проект – 1 кредит		3,4-й	3,4-й
Загальна кількість годин – 240 (210+30)		Лекції	
		20 год.	10 год.
		Практичні, семінарські	
		50 год.	12 год.
		Лабораторні	
	0 год.	0 год.	
	Самостійна робота		
	170 год.	218 год.	
	В т.ч. індивідуальні завдання:		
	30 год.		
Вид контролю:			
<b>Залік:</b> 3-й сем.	<b>Залік:</b> 3-й сем.		
<b>Екзамен:</b> 4-й сем.	<b>Екзамен:</b> 4-й сем.		
<b>Курс. проект:</b> 4-й сем	<b>Курс. проект:</b> 4-й сем		

### 3. Дидактична карта дисципліни

Назви змістових модулів і тем	Кількість годин												
	денна форма						заочна форма						
	усього	у тому числі					усього	у тому числі					
		о	л	лаб	п	інд		с.р.	о	л	лаб	п	інд
1	2	3	4	5	6	7	8	9	10	11	12	13	
<b>Модуль 2</b>													
<b>Змістовий модуль 1 Введення до ООП</b>													
Тема 1 Підходи до програмування	15	2		2		11	15	1		1		14	
Тема 2 Особливості ООП	15	2		4		9	15					14	
<b>Разом за ЗМ 1</b>	<b>30</b>	<b>4</b>		<b>6</b>		<b>20</b>	<b>30</b>	<b>1</b>		<b>1</b>		<b>28</b>	
<b>Змістовий модуль 2 Інкапсуляція та керування доступом до даних</b>													
Тема 3 Інкапсуляція	15	2		4		10	15	2		1		14	
Тема 4 Класи та методи доступу	15			4		10	15						13
<b>Разом за ЗМ 2</b>	<b>30</b>	<b>4</b>		<b>8</b>		<b>20</b>	<b>30</b>	<b>2</b>		<b>1</b>		<b>27</b>	
<b>Змістовий модуль 3 Поведінка та реалізація класу</b>													
Тема 5 Конструктори та деструктори	30	2		4		24	30	1		2		27	
<b>Змістовний модуль 4 Ієрархія класів</b>													
Тема 6 Класи-підкласи	15	2		6		8	15	2		2		13	
Тема 7 Дружні функції	15			6		8	15						13
<b>Разом за ЗМ4</b>	<b>30</b>	<b>2</b>		<b>12</b>		<b>16</b>	<b>30</b>	<b>2</b>		<b>2</b>		<b>26</b>	
<b>Разом за М1</b>	<b>120</b>	<b>10</b>		<b>30</b>		<b>80</b>	<b>120</b>	<b>6</b>		<b>6</b>		<b>108</b>	
<b>Модуль 3</b>													
<b>Змістовий модуль 5 Успадкування та поліморфізм</b>													
Тема 8 Наслідування та перевантаження	15	2		2		12	15	1		2		14	
Тема 9 Абстракція та віртуальні функції	15			4		10	15						13
<b>Разом за ЗМ5</b>	<b>30</b>	<b>2</b>		<b>6</b>		<b>22</b>	<b>30</b>	<b>1</b>		<b>2</b>		<b>27</b>	
<b>Змістовий модуль 6 UML-діаграми</b>													
Тема 10 Інфорграфічне моделювання	15	2		2		12	15	1		2		14	
Тема 11 Уніфікована мова програмування	15			2		12	15						13
<b>Разом за ЗМ6</b>	<b>30</b>	<b>2</b>		<b>4</b>		<b>24</b>	<b>30</b>	<b>1</b>		<b>2</b>		<b>27</b>	
<b>Змістовий модуль 7 Класи колекцій та виняткові ситуації</b>													
Тема 12 Система введення-виведення потокової інформації	10	2		4		4	10	2		2		9	
Тема 13 Шаблони та колекції	10	2		4		4	10						9
Тема 14 Обробка винятків	10	2		2		6	10						8
<b>Разом за ЗМ7</b>	<b>30</b>	<b>6</b>		<b>10</b>		<b>14</b>	<b>30</b>	<b>2</b>		<b>2</b>		<b>26</b>	
<b>Разом за М3</b>	<b>90</b>	<b>10</b>		<b>20</b>		<b>60</b>	<b>90</b>	<b>4</b>		<b>6</b>		<b>80</b>	
<b>Модуль 4</b>													
<b>Змістовий модуль 8 Курсове проектування</b>													
Курсове проектування	30					30	30					30	

Усього годин	240	20		50	30	140	240	10		12	30	188
--------------	-----	----	--	----	----	-----	-----	----	--	----	----	-----

### Самостійна робота

№ з/п	Назва теми	Вид роботи	Кіль. годин	
			Денна	Заочна
1.	Підходи до програмування	Опрацювання конспекту з теми	2	3
2.		Створення текстового документу для звіту	1	1
3.		Доопрацювання та оформлення лабораторної роботи № 1	8	10
4.	Особливості ООП	Опрацювання конспекту з теми	2	3
5.		Доопрацювання та оформлення лабораторної роботи № 2	7	11
6.	Інкапсуляція	Опрацювання конспекту з теми	2	3
7.		Доопрацювання та оформлення лабораторної роботи №3	8	11
8.	Класи та методи доступу	Опрацювання конспекту з теми	2	3
9.		Доопрацювання та оформлення лабораторної роботи №3	8	10
10.	Конструктори та деструктори	Опрацювання конспекту з теми	2	3
11.		Доопрацювання та оформлення лабораторної роботи № 4	22	24
12.	Класи-підкласи	Опрацювання конспекту з теми	2	3
13.		Доопрацювання та оформлення лабораторної роботи № 4	6	10
14.	Дружні функції	Опрацювання конспекту з теми	2	3
15.		Доопрацювання та оформлення лабораторної роботи № 4	6	10
16.	Наслідування та перевантаження	Опрацювання конспекту з теми	2	3
17.		Доопрацювання та оформлення лабораторної роботи № 5	10	11
18.	Абстракція та віртуальні функції	Опрацювання конспекту з теми	2	3
19.		Доопрацювання та оформлення лабораторної роботи № 6	8	10
20.	Інфографічне моделювання	Опрацювання конспекту з теми	2	3
21.		Доопрацювання та оформлення лабораторної роботи № 7	10	11
22.	Уніфікована мова програмування	Опрацювання конспекту з теми	2	3
23.		Доопрацювання та оформлення лабораторної роботи № 7	10	10
24.	Система введення-виведення потокової інформації	Опрацювання конспекту з теми	2	3
25.		Доопрацювання та оформлення лабораторної роботи № 8	2	6
26.	Шаблони та колекції	Опрацювання конспекту з теми	2	3
27.		Доопрацювання та оформлення лабораторної роботи № 9	2	6
28.	Обробка винятків	Опрацювання конспекту з теми	2	3
29.		Доопрацювання та оформлення лабораторної роботи № 10	4	5
<b>Разом</b>			<b>140</b>	<b>188</b>

#### 4. Індивідуальні завдання

Виконання індивідуального завдання у вигляді курсового проектування здійснюється згідно за відповідними методичними рекомендаціями впродовж четвертого семестру. Тему курсового проекту необхідно обрати з вказаного у рекомендаціях переліку або власну та узгодити з викладачем.

Формою контролю виконання студентом курсового проекту є публічний захист екзаменаційній комісії. Склад комісії та розклад проведення затверджується відповідним наказом ректора за поданням завідувача кафедри та навчального відділу.

Оцінювання результатів роботи проводить екзаменаційна комісія у день захисту згідно до шкали оцінювання

#### 5. Методи навчання

Дидактична карта даної дисципліни передбачає проведення лекційних та практичних занять, самостійної роботи.

Основна дидактична мета *лекційних занять* – формування орієнтувальної теоретичної основи з основ програмування для подальшого засвоєння студентами навчального матеріалу дисципліни. Лекційні заняття з даної дисципліни будуються на три ланковій структурі (вступ, основна частина, висновки) зі застосуванням графічно – символічних методів. Лекційний курс проводиться згідно до плану-конспекту у відповідності до даної робочої програми навчальної дисципліни (пункт 3) з метою розкриття загальних теоретичних і практичних основ і для набування знань і умінь у відповідності до освітньо-кваліфікаційної характеристики та освітньої програми підготовки майбутніх фахівців.

*Практичні заняття* проводяться у виді *Лабораторного практикуму* у добре підготовлених, спеціально обладнаних комп'ютерних аудиторіях обчислювального центру ЗІЕІТ. Основна мета лабораторних робіт – набування студентами відповідних практичних навичок, та систематизація і узагальнення здобутих під час лекційних занять теоретичних основ.

Лабораторні роботи виконуються згідно до методичних вказівок з урахуванням індивідуальних варіантів завдань, що регламентуються викладачем. Результати виконаних завдань формуються до індивідуального звіту про виконану роботу. Кожен студент повинен самостійно виконати всі лабораторні роботи й оформити їх результати.

*Самостійна робота студента* – виконання студентом за завданням та під методичним керівництвом і контролем викладача без його прямої участі наступних видів учіння:

- слухання та відпрацювання лекцій, виконання лабораторних робіт;
- доопрацювання у разі необхідності лабораторних робіт та відповідне їх оформлення до звіту;
- підготовка до підсумкового контролю.

Кожен студент має доступ до самостійної роботи у комп'ютерних аудиторіях інформаційно-обчислювального центру.

*Консультації з дисципліни* проводяться у відповідності до робочого навчального плану та згідно з розкладом проведення консультацій, наданого кафедрою та затвердженого навчальним відділом. Консультації проводяться на ІОЦ або у іншій аудиторії (закріпленої за кафедрою або викладачем). Консультаційна робота спрямована на допомогу в опануванні студентами методології певної теми чи розділу, методами самостійної навчальної роботи.

## 6. Система контролю та оцінювання.

Система складається з поточного, модульного та підсумкового контролю:

Поточний контроль здійснюється у кожній академічній групі, і полягає у тому, що студенти виконують практичні роботи з кожного модулю у відповідні аудиторні часи та у часи самостійної роботи. Виконана практична робота комплексно оцінюється викладачем, враховуючи такі критерії:

1. правильність одержаних результатів практичної роботи;
2. застосування раціонального методу рішення задач;
3. наявність звіту з лабораторної роботи. в якому наводиться наступна інформація:
  - опис теми лабораторної роботи,
  - стислі теоретичні відомості по темі,
  - хід виконання роботи з відповідною додатковою інформацією,
  - результати виконання роботи (вхідні і вихідні дані, тексти програм, алгоритми тощо),
  - висновки.

Кожний студент повинен узгодити з викладачем номер власного варіанту індивідуального завдання практикуму.

Практична частина, як форма поточного контролю, складається з 10 практичних робіт. Результат виконання і захисту студентом кожної роботи оцінюється окремо.

Для кожного з модулів кількість робіт та максимальна оцінка кожної наведена у наступній таблиці.

Модуль	Назва практичної роботи	Бали	
		Денна	Заочна
1	Створення об'єктів за допомогою структур	5	15
	Реалізація об'єктів за допомогою функцій	10	
	Вступ у класи та об'єкти. Модульність та обмеження доступу	15	15
	Класова ієрархія та механізм успадкування	10	20
	Віртуальність та поліморфізм	10	
Разом		50	50
2	Перевантаження операторів	5	10
	Віртуальність та абстрактні класи	5	
	Побудова UML-діаграм	10	10
	Використання об'єктів потоків	10	30
	Шаблони функцій та шаблони класів.	10	
	Параметризовані контейнерні класи.	10	
Обробка виняткових ситуацій	10		
Разом		50	50

У разі несвоєчасного захисту лабораторної роботи з неповажних причин викладач має право знизити максимальну оцінку за роботу.

Формою результатів виконання самостійної роботи з даної дисципліни є звіт, створення та оформлення якого наведено у методичних вказівках з самостійної роботи. Звіт перевіряє та оцінює викладач упродовж тижня самостійної роботи поточного модуля згідно за розкладом. Кількість балів за даний вид роботи не перевищує



	<p>десяти.</p> <p><i>Модульний контроль</i> проводиться у відповідності до наказу ректора «Про проведення поточного та семестрового контролю успішності студентів інституту» згідно за розкладом, встановленим навчальним відділом. Формою проведення модульного контролю є виконання тестової контрольної роботи, оформленої письмово або за допомогою спеціалізованого тестувального програмного забезпечення. Тривалість складання студентом модульної контрольної роботи не перевищує двох академічних годин. Максимальний рейтинговий бал при цьому не перевищує 40 балів. Модульний контроль здійснюється одноразово та не може бути перескладеним.</p> <p><i>Підсумковим контролем</i> засвоєння дисципліни є залік наприкінці другого модуля та екзамен наприкінці третього.</p> <p>У відповідності до стандарту підприємства підсумковий контроль автоматично проставляється як сума рейтингових балів поточного та модульного контролю. Студент може покращити свої результати у разі недостатньої кількості балів (&lt;60).</p> <p>У разі заліку студент має змогу здати цей вид підсумкового контролю згідно за розкладом, встановленим навчальним відділом. Студент може покращити результати поточного контролю шляхом перездачі або доздачі лабораторного практикуму.</p> <p>У разі екзамену при наявності мінімально необхідної кількості балів поточного контролю (від 20 балів) студент має змогу здати цей вид підсумкового контролю у письмовій формі за екзаменаційними білетами, що розроблені викладачем, згідно за розкладом, встановленим навчальним відділом. Письмову роботу кодують та надають викладачу на перевірку. Кількість максимальних рейтингових балів становить 40. Ці бали замінюють рейтингові бали модульного контролю, та у разі достатньої кількості з поточним контролем ставиться відповідна оцінка. Кількість спроб складання іспиту регламентована деканатом інституту.</p> <p>До захисту курсового проекту студент допускається викладачем, який є керівником курсового проектування. Необхідними умовами допуску є:</p> <ul style="list-style-type: none"> <li>- наявність відповідного діючого програмного продукту;</li> <li>- наявність пояснювальної записки, оформленої згідно з ДСТУ;</li> <li>- наявність презентації доповіді.</li> </ul> <p>Студент під час доповіді повинен викласти основні результати курсової роботи. Орієнтовний час виступу - 5 хвилин. Після доповіді студент повинен відповісти на поставлені запитання. Якщо з відповідей на запитання з'ясується, що студент не є єдиним автором курсової роботи, вона відхиляється, а студент отримує нове завдання</p> <p>Максимальний рейтингова оцінка із захисту курсового проектування становить 100 балів та поділяється на три базові категорії:</p> <ul style="list-style-type: none"> <li>- проект – 60 балів;</li> <li>- оформлення пояснювальної записки проекту – 20 балів;</li> <li>- презентація проекту – 20 балів.</li> </ul>
<p><b>7. Технічне й програмне забезпечення/обладнання</b></p>	<p><i>Лабораторний практикум</i> з дисципліни «Об'єктно-орієнтоване програмування» проводиться у добре підготовлених, спеціально обладнаних комп'ютерних аудиторіях обчислювального центру ЗІЕІТ (ОЦ ЗІЕІТ). З метою якісної підготовки фахівців кожному студенту на базі ОЦ ЗІЕІТ надається спеціально сформований та схвалений</p>

	<p>рішенням кафедри програмно-апаратний комплекс навчально-методичного забезпечення, що містить наступне:</p> <ul style="list-style-type: none"> <li>- обліковий запис (профіль користувача) на сервері обчислювального центру, що забезпечує санкціонований вхід до комплексу, регламентує певні права та правила користування, надає можливість контролю над відвідуванням;</li> <li>- персональний комп'ютер з мережним постійним запам'ятовуючим пристроєм, що підключається;</li> <li>- повний комплект системного та прикладного програмного забезпечення (вільного користування або з наявністю відповідної ліцензії);</li> <li>- комплект навчально-методичного матеріалу, що містить у повному обсязі методичне забезпечення, базову та допоміжну літературу;</li> <li>- доступ до Центру дистанційної освіти ЗІЕІТ та глобальної мережі Internet.</li> </ul> <p>Самостійна робота може виконуватись як у ОЦ ЗІЕІТ, так і у інших зручних місцях для студента, у час, вільний від основного навчання та за умови наявності у нього персонального комп'ютера з відповідним програмним забезпеченням.</p> <p>Перелік необхідного програмного забезпечення:</p> <ul style="list-style-type: none"> <li>- операційна система (MS Windows XP та вище, або LinuxUbuntu 14 або вище);</li> <li>- середовище розробки (MS VisualStudioCode 1.60.х або MS VisualStudio 2015 або ін.);</li> <li>- система UML-проекування (Umbrella 2.32.х, або UMLDesigner 9.0.х, або StarUML3.х або ін.);</li> <li>- текстовий процесор (OpenOffice4.х Writer або MS Word);</li> </ul> <p>Все програмне забезпечення має бути вільного користування або з відповідною ліцензією чи умовами (наприклад учнівська, тимчасова та ін.)</p>
<p><b>8. Політика дисципліни.</b></p>	<ul style="list-style-type: none"> <li>• Курс передбачає роботу в колективі.</li> <li>• Середовище в аудиторії є дружнім, творчим, відкритим до конструктивної критики.</li> <li>• Відвідування занять є обов'язковим компонентом оцінювання, за яке нараховуються бали. За об'єктивних причин (наприклад, хвороба, міжнародне стажування) навчання може відбуватись в дистанційному режимі за погодженням із керівником курсу та презентувати виконані завдання під час консультації викладача.</li> <li>• Самостійна робота включає в себе теоретичне вивчення питань, що стосуються тем лекційних занять, які не ввійшли в теоретичний курс, або ж були розглянуті коротко, їх поглиблена проробка за рекомендованою літературою, а також виконання завдань з метою закріплення теоретичного матеріалу.</li> <li>• Усі завдання, передбачені програмою, мають бути виконані у встановлений термін.</li> <li>• Студент, який спізнився, вважається таким, що пропустив заняття з неповажної причини з виставленням 0 балів за заняття, і при цьому має право бути присутнім на занятті.</li> <li>• За використання телефонів і комп'ютерних засобів без дозволу викладача, порушення дисципліни студент отримує за заняття 0 балів і зобов'язаний відпрацювати таке заняття.</li> <li>• Курсова робота повинна бути захищена не пізніше, ніж за тиждень до початку екзаменаційної сесії.</li> </ul>

	<ul style="list-style-type: none"><li>• Ліквідація заборгованості відбувається протягом 1 тижня після встановленого терміну. При цьому оцінка знижується на 10 %.</li><li>• Здобувачі вищої освіти мають право скласти екзамен автоматично, у випадку, якщо впродовж семестру такі здобувачі набрали 90-100 балів.</li><li>• Здобувачам вищої освіти після аудиторних занять надається право підвищувати свій рейтинг лише під час складання іспитів (підсумкового оцінювання) за графіком екзаменаційної сесії.</li><li>• Під час роботи над завданнями не допустимо порушення академічної доброчесності: при використанні Інтернет ресурсів та інших джерел інформації студент повинен вказати джерело, використане під час виконання завдання.</li><li>• Усі письмові роботи перевіряються на наявність плагіату і допускаються до захисту із коректними текстовими запозиченнями не більше 20%.</li><li>• У разі виявлення факту плагіату студент отримує за завдання 0 балів і повинен повторно виконати завдання, які передбачені у силабусі.</li><li>• Списування під час контрольних робіт та екзаменів заборонені (в т.ч. із використанням мобільних пристроїв). Мобільні пристрої дозволяється використовувати лише під час он-лайн тестування.</li></ul>
--	---

## 9. Розподіл балів, які отримують студенти

Семестр 3	Модуль 2	Тема 1	5	
		Тема 2	10	
		Тема 3	5	
		Тема 4	10	
		Тема 5	5	
		Тема 6	10	
		Тема 7	5	
		Самостійна робота	10	
		Модульний контроль	40	
		<b>Залік</b>	<b>100</b>	
Семестр 4	Модуль 3	Тема 8	5	
		Тема 9	5	
		Тема 10	5	
		Тема 11	5	
		Тема 12	10	
		Тема 13	10	
		Тема 14	10	
		Самостійна робота	10	
		Модульний контроль	40	
			<b>Екзамен</b>	<b>100</b>
	<b>Курсовий проект</b>	Робота (проект)	60	
		Оформлення	20	
		Доповідь	20	
<b>Разом</b>		<b>100</b>		

### Шкала оцінювання: національна та ЄКТС

Сума балів за всі види навчальної діяльності	Оцінка за національною шкалою	
	для екзамену, курсового проекту (роботи), практики	для заліку
90 – 100	відмінно	зараховано
82-89	добре	
74-81		
64-73	задовільно	
60-63		
35-59	незадовільно з можливістю повторного складання	не зараховано з можливістю повторного складання
0-34	незадовільно з обов'язковим повторним вивченням дисципліни	не зараховано з обов'язковим повторним вивченням дисципліни

## 10. Питання до заліку чи екзамену:

Теоретичні питання курсу:

1. Основи об'єктно-орієнтованого підходу.
2. Поняття об'єкту та порівняння зі структурами та алгоритмами.
3. Поняття класу.
4. Вбудовані inline функції. Перевантаження імен функцій.
5. Аргументи за замовчуванням. Область дії ідентифікаторів.

6. Особливості використання struct, enum, union.
7. Поняття інкапсуляції. Поняття про захист внутрішніх даних об'єкту.
8. Метод як канал доступу до внутрішніх даних.
9. Модель об'єкта як чорного ящика.
10. Клас та екземпляр.
11. Синтаксис оголошення класу.
12. Специфікатори доступу public, protected та private.
13. Означення методів класу за межами класу.
14. Конструктори, їх роль та призначення.
15. Конструктори по замовчуванню та з параметрами.
16. Особливості ініціалізації членів-даних, винесення перед тілом конструктора.
17. Динамічне виділення пам'яті для об'єкту: виклик конструктора з оператору new.
18. Деструктори.
19. Відношення клас-підклас та його зв'язок з відношенням абстрактне-конкретне.
20. Сумісність типів знизу вгору.
21. Дружні функції, окремі класи та класи в цілому, обхід механізмів захисту членів класу.
22. Переваги та недоліки використання механізму дружності в програмах з об'єктно-орієнтованою композицією.
23. Наслідування. Механізм наслідування членів-даних та методів.
24. Перевантаження операторів функціями та методами.
25. Поліморфізм та віртуальні функції.
26. Поняття оголошеного та фактичного типу.
27. Механізм виклику віртуальної функції.
28. Абстрактні функції та класи. Поняття інтерфейсу.
29. Мова UML.
30. Види та типи UML діаграм.
31. Діаграми прецедентів. Діаграми класів.
32. Діаграма активності. Діаграма розгортання.
33. Основні поняття. Файлові та вбудовані потоки.
34. Класи потоків. Перевизначення операторів.
35. Використання дружніх функцій.
36. Форматування даних. Використання маніпуляторів.
37. Створення власних маніпуляторів.
38. Файлове введення-виведення.
39. Класи потоків введення-виведення.
40. Управління форматом, модифікатори.
41. Шаблони функцій та класів.
42. Класи колекцій стандартної бібліотеки.
43. Поняття про виняткову ситуацію. Обґрунтування вимог до механізму обробки винятків.
44. Оператори try, throw і catch.
45. Порядок генерації та перехоплення виняткової ситуації.

## 11. Рекомендована література:

### Базова:

1. Жеребцов О.А. Об'єктно-орієнтоване програмування. Методичні вказівки з лабораторного практикуму для студентів ЗІЕІТ спеціальностей 121 «Інженерія програмного забезпечення», 122 «Комп'ютерні науки», 123 «Комп'ютерна інженерія», 2019 – 83 с.
2. Жеребцов О.А. Об'єктно-орієнтоване програмування. Методичні вказівки з курсового проектування для студентів ЗІЕІТ спеціальностей 121 «Інженерія програмного забезпечення», 122 «Комп'ютерні науки», 123 «Комп'ютерна інженерія», 2019 – 50 с.
3. Жеребцов О.А. Об'єктно-орієнтоване програмування. Методичні вказівки до самостійної роботи для студентів ЗІЕІТ спеціальностей 121 «Інженерія програмного забезпечення», 122

«Комп'ютерні науки», 123 «Комп'ютерна інженерія», 2019 – 15 с.

4. Ю.І. Грицюк. Об'єктно-орієнтоване програмування мовою С++. Навчальний посібник / Грицюк Ю.І., Рак / Львів: Вид-во Львівського ДУ БЖД, 2011. – 404 с. ISBN 978-966-3466-86-3
5. Дацун Н.М. Об'єктно-орієнтоване програмування: навчальний посібник для студентів спеціальності «Програмна інженерія». – Донецьк: ДонНТУ, 2014. – 205 с.
6. Жуковський С.С., Вакалюк Т.А. Об'єктно-орієнтоване програмування мовою С++. Навчально-методичний посібник для студентів напряму 6.040302 Інформатика. – Житомир: Вид-во ЖДУ, 2016. – 100 с.
7. Бублик В.В. Б90 Об'єктно-орієнтоване програмування: [Підручник] / В.В. Бублик. – К.: ІТкнига, 2015. – 624 с.: іл.

#### **Допоміжна:**

8. ВиртНиклаус. Алгоритмы и структуры данных : пер. с англ. / Н. Вирт. — 2-е изд., испр. — СПб. : НевскийДиалект, 2001. — 351 с.
9. Stroustrup, Bjarne. The C++ programming language / Bjarne Stroustrup.—Fourth edition. Addison-Wesley publishing. 2013 – 1366 p.
10. ГербертШилдт. С++ A Beginner's Guide: Step by Step. – М.: ЭКОМ Паблицерз., 2013. – 640 с. ISBN - 978-5-9790-0127-2, 0-07-219467-7.
11. С++. Теорія та практика : Навч. посібник / [О. Г. Трофименко, Ю. В. Прокоп, І. Г. Швайко, Л. М. Буката та ін.]; за ред. О. Г. Трофименко. – Одеса: ОНАЗ ім. О.С. Попова, 2011. – 588 с.
12. ЛиппманСтенли Б., ЛажойеЖози, Му Барбара Э. Язык программирования С++. Базовый курс. М.: Вильямс, 2014. — 1120 с. — 5-е изд. — ISBN: 9785845918390, 9780321714114.
13. Основи програмування : навчальний посібник для вузів / Броніслав Генріхович Пекарський . – Київ : Кондор, 2008 . – 363 с. : іл. - Бібліогр.: с.362-363(17 назв) . – На укр. яз. - ISBN 966-351-075-0 .

#### **Інформаційні ресурси:**

14. #SimpleCode. Основи С++. Программирование для начинающих. Youtube- канал для перегляду навчальних відеороликів. [Режим доступу] - <https://www.youtube.com/playlist?list=PLQOaTSbfxUtCrKs0nicOg2npJQYSPGO9r>.
15. Microsoft DeveloperNetwork. Інформаційний портал для розробників. [Режим доступу] - <https://msdn.microsoft.com/ru-ru/>.
16. Microsoft VisualStudio. Портал для завантаження безкоштовного програмного забезпечення для розробників. [Режим доступу] - <https://visualstudio.microsoft.com/ru/free-developer-offers/>.