

**Національний університет кораблебудування  
імені адмірала Макарова  
Херсонський навчально-науковий інститут**

Кафедра інформаційних технологій  
та фізико-математичних дисциплін

T7112

**ЗАТВЕРДЖЕНО**  
Заступник директора з  
навчальної роботи



*[Signature]* к.т.н., проф. Дудченко О.М.

**РОБОЧА ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ**

**Program of the Discipline**

**ОСНОВИ ПРОГРАМНОЇ ІНЖЕНЕРІЇ**

**FUNDAMENTALS OF SOFTWARE ENGINEERING**

рівень вищої освіти *перший (бакалаврський)*

тип дисципліни *обов'язкова*

мова викладання *українська*

**Херсон – 2022 рік**

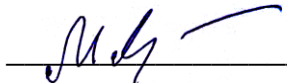
Робоча програма навчальної дисципліни «Основи програмної інженерії» є однією із складових комплексної підготовки фахівців галузі знань 12 «Інформаційні технології» спеціальності 121 «Інженерія програмного забезпечення» освітня програма «Інженерія програмного забезпечення»

«27»серпня 2022 р. – 29 с.

Розробники: Дудченко О.М. професор НУК кафедри інформаційних технологій та фізико-математичних дисциплін, канд. техн. наук; Карпова С.О., ст. викладач кафедри інформаційних технологій та фізико-математичних дисциплін.

Проект робочої програми навчальної дисципліни «Основи програмної інженерії» узгоджено з гарантом освітньої програми


Гарант освітньої програми «Інженерія програмного забезпечення»

д.пед.н., к.ф.-м.н., проф.  М.Б. Літвінова

Проект робочої програми навчальної дисципліни «Основи програмної інженерії» розглянуто на засіданні кафедри інформаційних технологій та фізико-математичних дисциплін

Протокол № 08 від «27»серпня 2022 р.

Завідувач кафедри

 П.Й. Гучек

Робоча програма навчальної дисципліни «Основи програмної інженерії» затверджена методичною радою ХННІ НУК.

Протокол №01 від «29» серпня 2022 р.

Голова МР ХННІ НУК

 О.М. Дудченко

## ЗМІСТ

Вступ.....	4
1. Опис навчальної дисципліни.....	5
2. Мета навчальної дисципліни.....	6
3. Передумови для вивчення дисципліни .....	6
4. Очікувані результати навчання.....	6
5. Програма навчальної дисципліни.....	8
6. Методи навчання, засоби діагностики результатів навчання та методи їх демонстрування .....	18
7. Форми поточного та підсумкового контролю.....	18
8. Критерії оцінювання результатів навчання.....	21
9. Засоби навчання.....	22
10. Рекомендовані джерела інформації.....	23
Додаток.....	26

## Вступ

### Анотація

*Освітньою програмою «Основи програмної інженерії» підготовки бакалаврів передбачено набуття студентами знань про побудову комп'ютерних програм, відображає закономірності їх розвитку, узагальнює досвід програмування у вигляді комплексу загальних знань і правил регламентації інженерної діяльності розробників програмного забезпечення (ПЗ).*

Ця інженерна дисципліна надає всю необхідну інформацію та стандарти для вибору найбільш відповідного методу проектування практичних завдань. Не виключається й творчий неформальний підхід до створення програмного забезпечення.

*Дисципліна «Основи програмної інженерії» охоплює всі аспекти створення ПЗ, починаючи від формування вимог до створення, супроводу та зняття з експлуатації ПЗ, а також включає інженерні методи оцінки трудовитрат, вартості, продуктивності й якості.*

*Дисципліна «Основи програмної інженерії» носить міждисциплінарний характер, вона забезпечує підготовку студентів до вивчення навчальних дисциплін «Проектний практикум», «Аналіз вимог до програмного забезпечення», «Моделювання програмного забезпечення», «Архітектура та проектування програмного забезпечення», «Професійна практика програмної інженерії», «Якість програмного забезпечення та тестування» а також розробку відповідних розділів атестаційної випускної роботи бакалавра.*

**Ключові слова:** *аналіз, життєвий цикл, вимоги, інженерія програмного забезпечення, програмування, проектування, тестування, управління.*

### Annotation

*The educational program «Fundamentals of Software Engineering» for bachelors provides students with knowledge about computer programs' structure, explains the patterns of their development, summarizes the experience of programming in the form of a set of general knowledge and rules for regulating engineering software developers' activities.*

This engineering discipline provides all the necessary information and standards to select the most appropriate method of designing practical tasks. A creative informal approach to software development is not excluded.

*The discipline «Fundamentals of Software Engineering» covers all aspects of software development - from requirements to the development, maintenance and decommissioning of software, and also includes engineering methods for estimating capacity, costs, efficiency and quality.*

*The discipline «Fundamentals of Software Engineering» is interdisciplinary, it prepares students to study disciplines «Project Workshop», «Analysis of software requirements», «Software Modeling», «Software Architecture and Design», «Professional practice of software engineering», «Software quality and testing» as well as the development of relevant sections of the bachelor's thesis.*

**Key words:** *analysis, life cycle, requirements, software engineering, programming, design, testing, management.*

## 1. Опис навчальної дисципліни

Найменування показників	Галузь знань, спеціальність (освітня програма), освітній рівень	Характеристика навчальної дисципліни	
		денна форма навчання	заочна форма навчання
Кількість кредитів – 6	Галузь знань 12 – «Інформаційні технології»	Обов'язкова	
Модулів - 6		<b>Рік підготовки</b>	
Змістових модулів - 6		1-й	1-й
Електронний адрес на сайті ХННІ НУК: <a href="http://kb.nuos.edu.ua/Licensing%20and%20accreditation%20specialties/b-software-engineering.html">http://kb.nuos.edu.ua/Licensing%20and%20accreditation%20specialties/b-software-engineering.html</a>	Спеціальність 121 – «Інженерія програмного забезпечення»  Освітня програма «Інженерія програмного забезпечення»	<b>Семестри</b>	
		1, 2-й	1, 2-й
		<b>Лекції</b>	
		1-й семестр – 15 год. 2-й семестр – 15 год.	6 год. 8 год.
		<b>Лабораторні</b>	
Індивідуальне науково-дослідне завдання нема		1-й семестр – 15 год. 2-й семестр – 15 год.	6 год. 6 год.
Загальна кількість годин - 180			
Тижневих годин для денної форми навчання: аудиторних: 1-й семестр – 2 2-й семестр – 2 самостійної роботи студента: 1-й семестр – 2 2-й семестр – 2	Освітній рівень: <b>перший (бакалаврський)</b>	<b>Самостійна робота</b>	
		1-й семестр – 60 год. 2-й семестр – 60 год.	78 год. 78 год.
		<b>Індивідуальні завдання: год.</b>	
		-	-
		<b>Види контролю:</b> 1-й семестр – залік 2-й семестр – екзамен	
		<b>Форма контролю:</b> комбінована (письмовий контроль, тестовий контроль)	

## **2. Мета навчальної дисципліни**

2.1 Метою вивчення навчальної дисципліни «Основи програмної інженерії» є формування у студентів згідно зі Стандартом вищої освіти України, затвердженим Наказом Міністерства освіти і науки України від 29.10.2018 №1166 таких компетентностей:

Інтегральна компетентність

– здатність розв'язувати складні спеціалізовані завдання або практичні проблеми інженерії програмного забезпечення, що характеризуються комплексністю та невизначеністю умов, із застосуванням теорій та методів інформаційних технологій;

Загальні компетентності

K01. Здатність до абстрактного мислення, аналізу та синтезу.

K02. Здатність застосовувати знання у практичних ситуаціях.

K05. Здатність вчитися і оволодівати сучасними знаннями.

K06. Здатність до пошуку, оброблення та аналізу інформації з різних джерел.

K13. Здатність ідентифікувати, класифікувати та формулювати вимоги до програмного забезпечення.

K15. Здатність розробляти архітектури, модулі та компоненти програмних систем.

K25. Здатність обґрунтовано обирати та освоювати інструментарій з розробки та супроводження програмного забезпечення.

K26. Здатність до алгоритмічного та логічного мислення.

## **3. Передумови для вивчення дисципліни**

Передумовами для вивчення даної дисципліни є дисципліни: математика, фізика та основи інформатики в курсі середньої школи.

## **4. Очікувані результати навчання**

Вивчення навчальної дисципліни передбачає формування та розвиток у студентів таких результатів навчання:

ПР01. Аналізувати, цілеспрямовано шукати і вибирати необхідні для вирішення професійних завдань інформаційно-довідникові ресурси і знання з урахуванням сучасних досягнень науки і техніки.

ПР02. Знати кодекс професійної етики, розуміти соціальну значимість та культурні аспекти інженерії програмного забезпечення і дотримуватись їх в професійній діяльності.

ПР03. Знати основні процеси, фази та ітерації життєвого циклу програмного забезпечення.

ПР04. Знати і застосовувати професійні стандарти і інші нормативно-правові документи в галузі інженерії програмного забезпечення.

ПР06. Уміння вибирати та використовувати відповідну задачі методологію створення програмного забезпечення.

ПР07. Знати і застосовувати на практиці фундаментальні концепції, парадигми і основні принципи функціонування мовних, інструментальних і обчислювальних засобів інженерії програмного забезпечення.

ПР14. Застосовувати на практиці інструментальні програмні засоби доменного аналізу, проектування, тестування, візуалізації, вимірювань та документування програмного забезпечення.

ПР23. Вміти документувати та презентувати результати розробки програмного забезпечення.

## **5. Програма навчальної дисципліни**

### **1-й семестр**

#### **Модуль 1.**

##### **Змістовний модуль 1. Інженерні основи програмного забезпечення**

Тема 1. Введення в програмну інженерію. Питання і відповіді про інженерію програмного забезпечення. Професійні і етичні вимоги до фахівців з програмного забезпечення.

Джерела інформації: [7] – стор. 17-32.

Тема 2. Системотехніка обчислювальних систем. Інтеграційні властивості систем. Система і її оточення. Моделювання систем. Процес створення систем. Придбання систем.

Джерела інформації: [7] – стор. 35-51.

Тема 3. Процес створення програмного забезпечення. Моделі процесу створення ПЗ. Ітераційні моделі розробки ПЗ. Специфікація програмного забезпечення. Проектування і реалізація ПЗ. Атестація програмних систем. Еволюція програмних систем. Автоматизовані засоби розробки ПЗ.

Джерела інформації: [1] – стор. 126-129; [7] – стор. 53-79.

Тема 4. Технології виробництва програмного забезпечення

Джерела інформації: [6] – стор. 22-45; [9] – стор. 8- 60; [12]; [15].

#### **Модуль 2.**

##### **Змістовний модуль 2. Вимоги до програмного забезпечення**

Тема 5. Вимоги до програмного забезпечення. Функціональні і нефункціональні вимоги. Призначені для користувача вимоги. Системні вимоги. Документування системних вимог.

Джерела інформації: [1] – стор. 184-238; [7] – стор. 105-125.

Тема 6. Розробка вимог. Аналіз здійсненності. Формування і аналіз вимог. Атестація вимог. Управління вимогами.

Джерела інформації: [3] – стор. 68-79; [7] – стор. 127-147; [6] – стор. 104-121; [8] – стор. 180-183.

Тема 7. Матриця вимог. Розробка матриці вимог.

Джерела інформації: [3] – стор. 67-68.



## Модуль 3.

### Змістовний модуль 3. Моделювання програмного забезпечення

Тема 8. Архітектурне проектування. Структуризація системи. Моделі управління. Модульна декомпозиція. Проблемно-залежна архітектура.

Джерела інформації: [1] – стор. 333-355; [3] – стор. 247-255; [7] – стор. 205-223.

Тема 9. Архітектура розподілених систем. Багатопроцесорна архітектура. Архітектура клієнт/сервер. Архітектура розподілених об'єктів. CORBA.

Джерела інформації: [7] – стор. 225-241.

Тема 10. Об'єктно-орієнтоване проектування. Об'єкти і класи об'єктів. Процес об'єктно-орієнтованого проектування. Модифікація системної архітектури.

Джерела інформації: [7] – стор. 243-263.

Тема 11. Проектування систем реального часу. Проектування систем. Програми, що управляють. Системи спостереження і управління. Системи збору даних.

Джерела інформації: [7] – стор. 265-281.

Тема 12. Проектування з повторним використанням компонентів. Покомпонентна розробка. Сімейства додатків. Проектні патерни.

Джерела інформації: [1] – стор. 389-390; [7] – стор. 283-301.

Тема 13. Проектування інтерфейсу користувача. Принципи проектування інтерфейсів користувача. Взаємодія з користувачем. Представлення інформації. Засоби підтримки користувача. Оцінювання інтерфейсу.

Джерела інформації: [1] – стор. 547-550; [7] – стор. 303-322.

## 2-й семестр

### Модуль 1.

#### Змістовний модуль 1. Технології розробки програмного забезпечення

Тема 14. Життєвий цикл програмного забезпечення: моделі та їх особливості.

Джерела інформації: [3] – стор. 49-58; [5] – стор. 16-59; [7] – стор. 35-55; [13].

Тема 15. Методологічні основи технологій розробки програмного забезпечення.

Джерела інформації: [2] – стор. 8-19.

Тема 16. Методи структурного аналізу і проектування ПЗ.

Джерела інформації: [2] – стор. 102-122.

Тема 17. Методи об'єктно-орієнтованого аналізу та проектування ПЗ. Мова моделювання UML.

Джерела інформації: [1] – стор. 323-330; [2] – стор. 142-161; [4] – стор. 33-43, 63-205.

## **Модуль 2.**

### **Змістовний модуль 2. Письмова комунікація. Документування проекту програмного забезпечення**

Тема 18. Документування етапів розробки ПЗ

Джерела інформації: [1] – стор. 554-556; [7] – стор. 122-125; [2] – стор. 228-233.

Тема 19. Планування проекту.

Джерела інформації: [1] – стор. 358-359; [7] – стор. 85-88.

Тема 20. Верифікація й атестація ПЗ. Тестування дефектів. Тестування збірки. Тестування об'єктно-орієнтованих систем. Інструментальні засоби тестування.

Джерела інформації: [3] – стор. 167-168; [7] – стор. 385-425.

Тема 21. Тестування програмного забезпечення

Джерела інформації: [1] – стор. 487-517; [3] – стор. 169-172; [6] – стор. 441-503

## **Модуль 3.**

### **Змістовний модуль 3. Управління проектом програмного забезпечення**

Тема 22. Управління проектами. Процеси управління. Планування проекту. Графік робіт. Управління ризиками.

Джерела інформації: [1] – стор. 107-111, 121-126; [3] – стор. 230-232; [7] – стор. 81-100.

Тема 23. Управління персоналом. Межі мислення. Групова робота. Підбір і збереження персоналу. Модель оцінки рівня розвитку персоналу.

Джерела інформації: [1] – стор. 111-120; [7] – стор. 447-466.

Тема 24. Оцінка вартості програмного продукту. Продуктивність. Методи оцінювання. Алгоритмічне моделювання вартості. Тривалість проекту і наймання персоналу.

Джерела інформації: [7] – стор. 469-491.

Тема 25. Управління якістю. Забезпечення якості і стандарти. Планування якості. Контроль якості. Вимірювання показників ПЗ.

Джерела інформації: [3] – стор. 206-219; [7] – стор. 493-511.

Тема 26. Надійність ПЗ. Забезпечення надійності ПЗ. Атестація безвідмовності. Гарантії безпеки. Оцінювання захищеності ПЗ.

Джерела інформації: [3] – стор. 219-221; [7] – стор. 327-341.

Тема 27. Вдосконалення виробництва ПЗ. Якість продукту і виробництва. Аналіз і моделювання виробництва. Вимірювання виробничого процесу. Модель оцінки рівня розвитку. Класифікація процесів вдосконалення.

Джерела інформації: [7] – стор. 513-529.

## 5.1 Тематичний план навчальної дисципліни

Назви змістових модулів і тем	Кількість годин									
	денна форма					заочна форма				
	усього	у тому числі				усього	у тому числі			
		л	п	лаб	с.р.		л	п	лаб	с.р.
1	2	3	4	5	6	7	8	9	10	11
<b>1-й семестр</b>										
<b>Модуль 1</b>										
<b>Змістовний модуль 1. Інженерні основи програмного забезпечення</b>										
Тема 1. Введення в програмну інженерію	5	1	-	1	5	2	2	-	2	6
Тема 2. Системотехніка обчислювальних систем	5	1	-	1	5					6
Тема 3. Процес створення програмного забезпечення	10	1	-	2	6					7
Тема 4. Технології виробництва програмного забезпечення	10	1	-		6					7
<b>Разом за змістовим модулем 1</b>	<b>30</b>	<b>4</b>	<b>-</b>	<b>4</b>	<b>22</b>	<b>30</b>	<b>2</b>	<b>-</b>	<b>2</b>	<b>26</b>
<b>Модуль 2</b>										
<b>Змістовний модуль 2. Вимоги до програмного забезпечення</b>										
Тема 5. Вимоги до програмного забезпечення	10	1	-	-	7	2	2	-	2	8
Тема 6. Розробка вимог	10	2	-	3	7					9
Тема 7. Матриця вимог	10	2	-		8					9
<b>Разом за змістовим модулем 2</b>	<b>30</b>	<b>5</b>	<b>-</b>	<b>3</b>	<b>22</b>	<b>30</b>	<b>2</b>	<b>-</b>	<b>2</b>	<b>26</b>
<b>Модуль 3</b>										
<b>Змістовний модуль 3. 3. Моделювання програмного забезпечення</b>										
Тема 8. Архітектурне проектування	5	1	-	3	3	2	2	-	2	4
Тема 9. Архітектура розподілених систем	5	1	-		3					4
Тема 10. Об'єктно-орієнтоване проектування	5	1	-	2	3					4
Тема 11. Проектування систем реального часу	5	1	-	-	2					4
Тема 12. Проектування з повторним використанням компонентів	5	1	-	3	2					5
Тема 13. Проектування інтерфейсу користувача	5	1	-		3	5				
<b>Разом за змістовим модулем 3</b>	<b>30</b>	<b>6</b>	<b>-</b>	<b>8</b>	<b>16</b>	<b>30</b>	<b>2</b>	<b>-</b>	<b>2</b>	<b>26</b>
<b>Разом за 1-й семестр</b>	<b>90</b>	<b>15</b>	<b>-</b>	<b>15</b>	<b>60</b>	<b>90</b>	<b>6</b>	<b>-</b>	<b>6</b>	<b>78</b>

<b>2-й семестр</b>										
<b>Модуль 1</b>										
<b>Змістовний модуль 1. Технології розробки програмного забезпечення</b>										
Тема 14.Життєвий цикл програмного забезпечення: моделі та їх особливості	7	1	-	4	6	30	2	-	2	6
Тема 15.Методологічні основи технологій розробки програмного забезпечення	7	1	-		6					6
Тема 16.Методи структурного аналізу і проектування ПЗ	8	1	-		5					7
Тема 17.Методи об'єктно-орієнтованого аналізу та проектування ПЗ. Мова моделювання UML	8	1	-		5					7
<b>Разом за змістовим модулем 1</b>	<b>30</b>	<b>4</b>	<b>-</b>	<b>4</b>	<b>22</b>	<b>30</b>	<b>2</b>	<b>-</b>	<b>2</b>	<b>26</b>
<b>Модуль 2</b>										
<b>Змістовний модуль 2.Письмова комунікація. Документування проекту програмного забезпечення</b>										
Тема 18.Документування етапів розробки ПЗ	7	1	-	-	5	30	2	-	2	6
Тема 19.Планування проекту ПЗ	7	1	-	-	5					6
Тема 20.Верифікація й атестація ПЗ	8	1	-	5	5					7
Тема 21.Тестування програмного забезпечення	8	2	-	-	5					7
<b>Разом за змістовим модулем 2</b>	<b>30</b>	<b>5</b>	<b>-</b>	<b>5</b>	<b>20</b>	<b>30</b>	<b>2</b>	<b>-</b>	<b>2</b>	<b>26</b>
<b>Модуль 3</b>										
<b>Змістовний модуль 3.Управління проектом програмного забезпечення</b>										
Тема 22. Управління проектами	5	1	-	3	3	30	2	-	2	4
Тема 23.Управління персоналом	5	1	-	-	3					4
Тема 24.Оцінка вартості програмного продукту	5	1	-	3	3					4
Тема 25.Управління якістю	5	1	-	-	3					4
Тема 26. Надійність ПЗ	5	1	-	-	3					5
Тема 27.Вдосконалення виробництва ПЗ	5	1	-	-	3					5
<b>Разом за змістовим модулем 3</b>	<b>30</b>	<b>6</b>	<b>-</b>	<b>6</b>	<b>18</b>	<b>30</b>	<b>2</b>	<b>-</b>	<b>2</b>	<b>26</b>
<b>Разом за 2-й семестр</b>	<b>90</b>	<b>15</b>	<b>-</b>	<b>15</b>	<b>60</b>	<b>90</b>	<b>6</b>	<b>-</b>	<b>6</b>	<b>78</b>

**Примітка.** Для студентів заочної форми навчання викладаються оглядові лекції за темами змістових модулів в обсягах відповідно до таблиці.

## 5.2 Теми лабораторних занять

№ з/п	Назва теми	Кількість годин	
		Денна форма	Заочна форма
<b>1-й семестр</b>			
1	Побудова карти пам'яті (діаграма-зв'язку). Ознайомлення з програмою MindManager. Джерела інформації: [11], робота 1.	2	0,5
2	Побудова карти пам'яті (діаграма-зв'язку) для заданого процесу в програмі MindManager. Джерела інформації: [11,16], робота 2.	2	0,5
3	Створення презентацій. Джерела інформації: [17-21], робота 3.	2	1
4	Розробка матриці вимог. Джерела інформації: [3,14], робота 4.	2	1
5	Схематична розробка моделі потоків даних. Джерела інформації: [8,9], робота 5.	2	1
6	Схематична розробка об'єктних моделей. Джерела інформації: [10], робота 6.	2	1
7	Створення елементів графічного інтерфейсу користувача. Джерела інформації: [13], робота 7.	3	1
<b>Разом за 1-й семестр</b>		<b>15</b>	<b>6</b>
<b>2-й семестр</b>			
1	Розробка вимог до програмного забезпечення. Джерела інформації: [1, 2, 6-8], робота 1.	2	1
2	Розробка інтерфейсу та виконання розрахунку в середовищі VisualFortran для заданого процесу. Огляд методів розробки критичних систем. Джерела інформації: [1,7], робота 2.	5	1
3	Побудова тимчасової діаграми етапів робіт для заданого процесу. Джерела інформації: [1,3,7], робота 3.	2	1
4	Тестування програмного забезпечення. Джерела інформації: [3,7], робота 4.	4	1
5	Оцінка вартості програмного продукту. Джерела інформації: [7], робота 5.	2	2
<b>Разом за 2-й семестр</b>		<b>15</b>	<b>6</b>
<b>Разом</b>		<b>30</b>	<b>12</b>

### 5.3 Самостійна робота

№ з/п	Назва теми	Кількість годин	
		Денна форма	Заочна форма
<b>1-й семестр</b>			
1	Проектування систем. Розробка підсистем. Збірка системи. Інсталяція системи.	0,25	1
2	Структурована мова специфікацій. Створення специфікацій за допомогою PDL.	0,25	1
3	Постійні і змінні вимоги. Планування управління вимогами. Управління змінами вимог.	0,5	1
4	Моделі систем. Моделі системного оточення. Поведінкові моделі. Моделі даних. Об'єктні моделі. Інструментальні CASE-засоби.	0,5	1
5	Моделі потоків даних. Моделі кінцевих автоматів. Моделі спадкоємства.	0,5	2
6	Використання прототипів програмних систем. Використання прототипів в процесі розробки програмного забезпечення. Технології швидких прототипів. Використання прототипів призначених для користувача інтерфейсів.	0,5	2
7	Програмування баз даних. Збірка додатків з повторним використанням компонентів.	1	2
8	Формальні специфікації в процесі розробки програмного забезпечення. Специфікація інтерфейсів. Специфікація поведінки систем.	1	2
9	Проектування систем. Розробка підсистем. Збірка системи. Інсталяція системи.	1	2
10	Оточення системи і моделі її використання. Проектування архітектури. Визначення об'єктів. Моделі архітектури. Специфіцирування інтерфейсів об'єктів.	1	2
11	Моделювання систем реального часу. Програмування систем реального часу.	1	2
<b>Разом</b>		<b>7,5</b>	<b>18</b>
<b>2-й семестр</b>			
1	Формальна розробка систем. Розробка програмного забезпечення на основі раніше створених компонентів. Модель покрокової розробки. Спіральна модель розробки.	0,25	2
2	Планування верифікації та атестації. Інспектування програмних систем. Автоматичний статичний аналіз програм. Метод "чиста кімната".	0,25	2
3	Тестування методом чорного ящика. Области еквівалентності. Структурне тестування. Тестування гілок. Низхідне і висхідне тестування. Тестування	1	2

	інтерфейсів. Тестування з навантаженням. Тестування класів об'єктів. Інтеграція об'єктів.		
4	План проекту. Контрольні позначки етапів робіт.	1	2
5	Тимчасові і мереживі діаграми. Визначення ризиків. Аналіз ризиків. Планування ризиків. Моніторинг ризиків.	1	2
6	Організація людської пам'яті. Вирішення завдань. Мотивація. Створення команди. Згуртованість команди. Спілкування в групі. Організація групи. Робоча середа.	0,5	2
7	Модель СОСОМО. Алгоритмічні моделі вартості у плануванні проекту.	0,5	2
8	Стандарти на технічну документацію. Якість процесу створення програмного забезпечення і якість програмного продукту. Перевірка якості. Процес вимірювання. Показники програмного продукту.	1	1
9	Системи, критичні щодо забезпечення безпеки. Специфікація критичних систем. Розробка критичних систем.	1	1
10	Винятки в процесі створення програмного забезпечення. Оцінювання рівня розвитку	0,5	1
11	Еволюція програмного забезпечення. Успадковані системи. Модернізація програмного забезпечення. Реінженірінг програмного забезпечення. Управління конфігураціями.	0,5	1
<b>Разом</b>		<b>7,5</b>	<b>18</b>



## 5.4 Розподіл годин самостійної роботи

№ з/п	Вид роботи	Кількість годин	
		денна форма	заочна форма
<b>1-й семестр</b>			
1	Підготовка до лекцій	7,5	15
2	Підготовка до лабораторних робіт	15	15
3	Підготовка до поточного модульного контролю	15	-
4	Виконання контрольної роботи	-	15
5	Самостійне вивчення тем, що не входять до лекційного курсу	7,5	18
6	Підготовка до підсумкового контролю	15	15
<b>Разом за 1-й семестр</b>		<b>60</b>	<b>78</b>
<b>2-й семестр</b>			
1	Підготовка до лекцій	7,5	15
2	Підготовка до лабораторних робіт	15	15
3	Підготовка до поточного модульного контролю	15	-
4	Виконання контрольної роботи	-	15
5	Самостійне вивчення тем, що не входять до лекційного курсу	7,5	18
6	Підготовка до екзамену	15	15
<b>Разом за 2-й семестр</b>		<b>60</b>	<b>78</b>
<b>Разом</b>		<b>120</b>	<b>156</b>

## 6. Методи навчання, засоби діагностики результатів навчання та методи їх демонстрування

Методи навчання:

для всіх видів занять:

- робота з літературою - опрацювання різних видів джерел, спрямоване на формування нових знань, їх закріплення, вироблення вмінь і навичок та реалізацію контрольної-корекційної функції в умовах формальної освіти;

для лекційних занять:

- лекція - усний виклад навчального матеріалу, який характеризується великим обсягом, складністю логічних побудов, сконцентрованою розумових образів, доведень і узагальнень;

- відеометод - використання відеоматеріалів для активізації наочно-чуттєвого сприймання; забезпечує більш легке і міцне засвоєння знань в їх образно-понятійній цілісності та емоційній забарвленості;

для лабораторних занять:

- лабораторна робота - метод поглиблення і закріплення теоретичних знань шляхом створення програм і отримання результатів роботи програми з використанням комп'ютерів;

- інструктаж - ознайомлення зі способами виконання завдань, інструментами, матеріалами, технікою безпеки та організацію робочого місця.

Засобами оцінювання та методами демонстрування результатів навчання є:

- звіти з виконання лабораторної роботи та презентації результатів виконаних лабораторних робіт на комп'ютері (або письмовий контроль результатів);

- усні відповіді на лабораторних заняттях;

- поточні модульні контрольні роботи у формі тестування (тестовий контроль);

- залік, екзамен.

## 7. Форми поточного та підсумкового контролю

Досягнення студента оцінюються за 100-бальною системою Університету.

Підсумкова оцінка навчального курсу включає в себе оцінки з поточного контролю і оцінки заключного іспиту.

Питома вага заключного іспиту в загальній системі оцінок – **40 балів**. Право здавати заключний іспит дається студенту, якій з урахуванням максимальних балів проміжних оцінок і заключного іспиту набирає не менше **60 балів**. Підсумкова оцінка навчального курсу є сумою проміжних оцінок і оцінки іспиту.

Поточний контроль проводиться на кожному лабораторному занятті та за результатами виконання завдань самостійної роботи. Він передбачає оцінювання теоретичної підготовки здобувачів вищої освіти із зазначеної теми (у тому числі, самостійно опрацьованого матеріалу) під час виконання завдань лабораторних робіт.

Зарахування кредитів навчального курсу можливо тільки після досягнення результатів, запланованих РПНД, що виражається в одній з позитивних оцінок, передбачених чинним законодавством.

## 7.1 Форми контролю результатів навчальної діяльності студентів та їх оцінювання

### Критерії оцінювання лабораторних робіт

Бал*	Критерії оцінювання
5/10	Робота виконана у встановлений термін. Виконана самостійно, чітко сформульовані цілі, завдання та гіпотеза досліджень. Застосовувалися коректні методи обробки отриманих результатів. У висновках проведена коректна інтерпретація результатів.
4/8	Робота виконана у встановлений термін. Студент виконує лабораторну роботу згідно з інструкцією, іноді після консультації викладача; описує спостереження; в цілому правильно складає звіт та робить висновки.
3/6	Робота виконана з порушенням встановлених термінів. Студент виконує лабораторну роботу згідно з інструкцією, іноді після консультації викладача; описує спостереження; складає звіт, що містить неточності у висновках та помилки.
2/4	Робота виконана з порушенням встановлених термінів. Студент виконує лабораторну згідно з інструкцією; складений звіт містить неточності у висновках та помилки.
1/2	Робота виконана з порушенням встановлених термінів. Студент виконує лабораторну під керівництвом викладача; складений звіт містить неточності у висновках та помилки.
0	Робота не виконувалася

\* - Бал 1-го семестру/бал 2-го семестру

### Критерії оцінювання поточного модульного контролю знань у формі тестування

Правильних відповідей, %	100	90	80	70	60	50	40	30	20	10
<b>1-й семестр</b>										
<b>Бал</b>	20	18	16	14	12	10	8	6	4	2
<b>2-й семестр</b>										
<b>Бал</b>	15	13	11	9	8	7	6	5	4	2

## Критерії оцінювання контрольної роботи (для заочної форми)

Бал*	Критерії оцінювання
20/15	Робота виконана у встановлений термін. Матеріал викладено у достатньому обсязі, аргументовано і у правильній послідовності. Під час захисту роботи студент вільно орієнтується в матеріалах.
15/10	Робота виконана у встановлений термін. Матеріал викладено у достатньому обсязі, але частка програм наведена без результатів розрахунків. Під час захисту роботи студент вільно орієнтується в матеріалах.
10/5	Робота виконана з порушенням встановлених термінів. Матеріал викладено у правильній послідовності, але недостатньо повно, більша частка програм наведена без результатів розрахунків. Під час захисту роботи студент слабо орієнтується в матеріалах.
0	Роботу не виконано.

\* - Бал 1-го семестру/бал 2-го семестру

## Критерії оцінювання підсумкового контролю та екзамену

Бал	Критерії оцінювання
40	Студент склав програму самостійно без помилок та відповідає на теоретичні питання без помилок
30	Студент склав програму самостійно без помилок, але відповіді на теоретичні питання не повні
20	Студент розуміє алгоритм, але склав програму, яка працює не правильно, проте відповідає на теоретичні питання без помилок
10	Студент не розуміє алгоритм, не склав програму, але відповідає на теоретичні питання без помилок
0	Студент не розуміє алгоритм, не склав програму і не відповідає на теоретичні питання без помилок

## Узагальнюючі результати поточного контролю знань

Форма контролю	Максимальна кількість балів	
	Денна форма	Заочна форма
<b>1-й семестр</b>		
Виконання лабораторних робіт	7 роб. × 5 балів = 35 балів	7 роб. × 5 балів = 35 балів
Поточний модульний контроль	1 МКР × 25 балів = 25 балів	-
Виконання контрольних робіт	-	1 роб. × 25 балів = 25 балів
<b>Всього</b>	<b>60</b>	<b>60</b>

<b>2-й семестр</b>		
Виконання лабораторних робіт	5 роб. × 10 балів = 50 балів	5 роб. × 10 балів = 50 балів
Поточний модульний контроль	1 МКР × 10 балів = 10 балів	-
Виконання контрольних робіт	-	1 роб. × 10 балів = 10 балів
<b>Всього</b>	<b>60</b>	<b>60</b>

### 8. Критерії оцінювання результатів навчання

Змістовий модуль	Тема	Денна форма		Заочна форма	
		Вид роботи	Бали	Вид роботи	Бали
1	2	3	4	5	6
<b>1-й семестр</b>					
ЗМ 1	T1	Лабораторна робота № 1	5	Лабораторна робота № 1	5
	T2	Лабораторна робота № 2	5	Лабораторна робота № 2	5
	T3 –T4	Лабораторна робота № 3	5	Лабораторна робота № 3	5
ЗМ 2	T6 –T7	Лабораторна робота № 4	5	Лабораторна робота № 4	5
	T1-T7	Поточний модульний контроль	25	Поточний модульний контроль	-
ЗМ 3	T8-T9	Лабораторна робота № 5	5	Лабораторна робота № 5	5
	T10	Лабораторна робота № 6	5	Лабораторна робота № 6	5
	T12-T13	Лабораторна робота № 7	5	Лабораторна робота № 7	5
	T1-T13	-	-	Контрольна робота	25
Підсумковий контроль		Тест	40	Тест	40
Сума			<b>100</b>		<b>100</b>
<b>2-й семестр</b>					
ЗМ 4	T5, T18	Лабораторна робота № 1	10	Лабораторна робота № 1	10
	T13-T17	Лабораторна робота № 2	10	Лабораторна робота № 2	10

1	2	3	4	5	6
	T5, T13-T18	Поточний модульний контроль	10		-
ЗМ 5	T19, T22	Лабораторна робота № 3	10	Лабораторна робота № 3	10
	T20	Лабораторна робота № 4	10	Лабораторна робота № 4	10
ЗМ 6	T24	Лабораторна робота № 5	10	Лабораторна робота № 5	10
				Контрольна робота -	10
Підсумковий контроль		Екзамен	40	Екзамен	40
Сума			<b>100</b>		<b>100</b>

### 9. Засоби навчання

Технічні засоби навчання: мультимедійний проектор, персональні комп'ютери з підключенням до мережі Інтернет.

При проведенні занять за дистанційною формою навчання (у період карантину) використовуються дистанційні платформи й інформаційно-комунікаційні технології (Moodle, Google Classroom, DingTalk, ZOOM Cloud Meetings, Skype, Viber, WeChat, Telegram, соціальні мережі тощо).

## 10. Рекомендовані джерела інформації

### Основна література

1. Брауде Эрик Дж.. Технология разработки программного обеспечения. Пер.с англ. - СПб.: Питер, 2004. – 655с.:ил.
2. Иванова Г.С. Технология программирования: Учебник, 3-е изд., стер. — Москва : КНОРУС, 2018. — 336 с.
3. Лаврищева Е.М., Петрухин В.А. Методы и средства инженерии программного обеспечения: Учебник. - М.: 2006. – 304с.
4. Ларман К. Применение UML и шаблонов проектирования, 2-е издание.: Пер. с англ. – М.: Издательский дом "Вильямс", 2004. – 620с.
5. Липаев, В.В. Программная инженерия. Методологические основы: Учеб. / В. В. Липаев; Гос. ун-т - Высшая школа экономики. -М.: ТЕИС, 2006. - 608 с.
6. Орлов С.А. Программная инженерия: Учебник для вузов. 5-е издание обновленное и дополненное. Стандарт третьего поколения.-Питер, 2020.- 640 с.:ил.
7. Соммервилл Иан. Инженерия программного обеспечения, 6-е издание.: Пер. с англ. – М.: Издательский дом "Вильямс", 2002. – 624 с.: ил.
8. Халл Э., Джексон К. Разработка и управление требованиями, GrayPublishing, USA (2-е изд. - 2005г.). -229 с. пер. Корнипаев Илья. Практическое руководство пользователя.
9. Якунин Ю.Ю. Технологии разработки программного обеспечения. Красноярск: ИПК СФУ, 2008.-225с.
- 10.<http://www.intuit.ru/department/se/inprogeng/1/>
- 11.<http://www.ixbt.com/soft/mind-manager.shtml>
- 12.[http://tehprog.ru/index.php\\_page=lecture12.html](http://tehprog.ru/index.php_page=lecture12.html)
- 13.[http://it-claim.ru/Education/Course/ISDevelopment/Lecture\\_5.pdf](http://it-claim.ru/Education/Course/ISDevelopment/Lecture_5.pdf)
- 14.<http://edu.reqcenter.pro/?p=173>
- 15.[http://www.interface.ru/iarticle/files/35798\\_52840953.pdf](http://www.interface.ru/iarticle/files/35798_52840953.pdf)
- 16.<http://www.mindjet.com/mindmanager/>
- 17.<http://antonkozlov.ru/kak-sdelat/kak-sdelat-prezentaciyu-na-kompyutere.html>
- 18.[http://www.lessons-tva.info/edu/e-inf2/m2t3\\_1.html](http://www.lessons-tva.info/edu/e-inf2/m2t3_1.html)
- 19.<http://pcpro100.info/kak-sdelat-prezentatsiyu/>
- 20.<http://presportal.ru/power-point/kak-sdelat-prezentaciyu-v-power-point-pervye-shagi/>
- 21.<https://support.office.com/>

## Допоміжна література

1. Бабенко Л.П., Лавріщева К.М. Основи програмної інженерії. Навч. посіб. – К.: т-во "Знання", 2001.
2. Бартенев О.В. Visual Fortran: новые возможности. – М.: «Диалог-МИФИ», 1999. - 304с.
3. Вигерс К.И. Разработка требований к программному обеспечению. Пер. с англ. - М.: Русская редакция, 2004.
4. Винниченко И.. Автоматизация процессов тестирования. - СПб: «Питер», 2005. - 203 с.
5. Геци К., Джазайери М., Мандриоли Д. Основы инженерии программного обеспечения. Пер. с англ. - СПб.: БХВ-Петербург, 2005.
6. Калашян А.Н., Калянов Г.Н. - Структурные модели бизнеса: DFD-технологии. - М.: Финансы и статистика, 2003
7. Кознов Д.В. Введение в программную инженерию. Часть I. Изд-во Санкт-Петербургского ун-та, 2005 г., 43 с.
8. Конноли Т., Бегг К. - Базы данных: проектирование, реализация и сопровождение. Теория и практика. 3-е издание.: Пер. с англ. - М.: Вильямс, 2003
9. Котляров В.П. Основы тестирования программного обеспечения: Учебное пособие / В.П. Котляров, Т.В. Коликова. - М.: Интернет-Университет Информационных Технологий; БИНОМ. Лаборатория знаний, 2006. -285с: ил. (Серия «Основы информационных технологий»).
10. Леффингуэлл Д., Уидриг Д. Принципы работы с требованиями к программному обеспечению. Унифицированный подход. Пер. с англ. -М.: Вильямс, 2002.
11. Липаев В.В. Системное проектирование сложных программных средств для информационных систем. Изд. второе, переработанное и дополненное. - М.: СИНТЕГ, 2002.
12. Липаев В.В. Функциональная безопасность программных средств. -М.: СИНТЕГ, 2004.
13. Липаев В.В. Анализ и сокращение рисков проектов сложных программных средств. - М.: СИНТЕГ, 2004.
14. Липаев В.В. Документирование сложных программных средств. -М.: СИНТЕГ, 2005.
15. Марка Д.А., МакГоуэн К. Методология структурного анализа и проектирования. - М.: МетаТехнология, 1993
16. Мацяшек Л. А. Анализ требований и разработка информационных систем с использованием UML. - М.: Издат. дом "Вильямс", 2002.
17. Орлов С. Технологии разработки программного обеспечения: Учебник-СПб.: Питер, 2002. - 464 с.: ил.
18. Савин Р. Тестирование Dot Com, или Пособие по жестокому обращению с багами в интернет-стартапах. - М.: Дело, 2007. - 312 с.
19. Фатрелл Р. Т., Шафер Д. Ф., Шафер Л. И. Управление программными проектами: достижение оптимального качества при минимальных затратах. Пер. с англ. - М.: Вильямс, 2003.



20. Черемных С.В. , Семенов И.О. , Ручкин В.С.- Структурный анализ систем: IDEF-технологии. - М.: Финансы и статистика, 2001

21. Якобсон А., Буч Г., Рамбо Дж. Унифицированный процесс разработки программного обеспечения. Пер. с англ. — СПб.: Питер, 2002.

22. Boehm V.W. et al. Software cost estimation with COCOMO II. PrenticeHall PTR. New Jersey, 2000.

23. Encyclopedia of Software Engineering. Vol.1 A-N; Vol.2 0-Z. Editor -In - Chief John J. Marciniak. John Wiley & Sons. Inc. 2001.

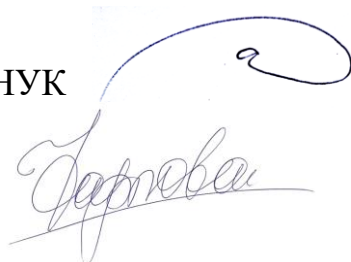
24. Titus Winters Tom Manshreck Hyrum Wright Software Engineering at Google: Lessons Learned from Programming Over Time. - O'Reilly Media, 2020.

### Інформаційні ресурси в Інтернет

1. Сайт ХННІ НУК: <http://kb.nuos.edu.ua>
2. [https://courses.prometheus.org.ua/courses/course-v1:LITS+115+2017\\_T4/about](https://courses.prometheus.org.ua/courses/course-v1:LITS+115+2017_T4/about)
3. <https://eduhub.in.ua/courses>
4. <https://ru.coursera.org/courses?query=software%20engineering>

Розробники:  
к.т.н., професор НУК

ст. викладач



Дудченко О.М.

Карпова С.О.

**Питання для модульного контролю****Контрольні питання до 1-го модуля**

1. Що таке програмне забезпечення?
2. Що таке програмна інженерія?
3. У чому відмінність між інженерією програмного забезпечення і комп'ютерною наукою?
4. У чому відмінність програмної інженерії від системотехніки?
5. Що таке процес створення програмного забезпечення?
6. Що таке модель процесу? Які моделі процесу програмного забезпечення?
7. Що таке узагальнені моделі створення програмного забезпечення?
8. Що таке ітераційні моделі розробки програмного забезпечення?
9. Автоматизовані засоби розробки програмного забезпечення (CASE-засоби)?
10. Основні відмінності між процесом створення систем і процесом розробки програмного забезпечення?
11. Що таке технологія виробництва програмного забезпечення?

**Контрольні питання до 2-го модуля**

1. Що таке вимога до програмного забезпечення?
2. Які вимоги призначені для користувача?
3. Що таке системні вимоги?
4. Що таке функціональні і нефункціональні вимоги?
5. Що таке документування системних вимог?
6. Методи формування і аналізу вимог?
7. Що таке атестація вимог?
8. Що таке огляд і управління вимогами?
9. Що таке матриця вимог і які види матриць існують?
10. Що таке моделі робочого оточення систем?
11. Що таке поведінкові моделі, назвіть їх види?
12. Що таке об'єктні моделі, назвіть їх види?
13. Що таке концепція моделювання поведінки, даних і об'єктів систем?
14. Які інструментальні CASE-засоби, використовуються при моделюванні систем?
15. Роль прототипів в процесі розробки програмного забезпечення?
16. Відмінність між еволюційним і експериментальним використанням прототипів?
17. Які методи розробки прототипів існують?
18. Які методи формальної специфікації існують?

## Контрольні питання до 3-го модуля

1. Що таке архітектурне проектування програмного забезпечення?
2. Які моделі, використовують при документуванні системної архітектури?
3. Які типи архітектури програмного забезпечення існують??
4. Які існують моделі управління?
5. Що таке модульна декомпозиція?
6. Що таке моделі проблемно-залежної архітектури?
7. Що таке структуризація системи і назвіть її моделі?
8. Які основні переваги і недоліки розподілених систем?
9. Що таке багатопроекторна архітектура?
10. Які підходи використовують при розробці архітектури клієнт/сервер?
11. Які відмінності між архітектурою клієнт/сервер і архітектурою розподілених об'єктів?
12. Яка концепція брокера запитів до об'єктів і які принципи, реалізовані в стандартах CORBA?
13. Що таке стандарти CORBA?
14. Які основні етапи процесу об'єктно-орієнтованого проектування?
15. Що таке об'єкти і класи об'єктів?
16. Які моделі використовуються при документуванні об'єктно-орієнтованої структури?
17. Які основні концепції систем реального часу і, чому ці системи зазвичай реалізовані у вигляді паралельних процесів?
18. Які основні етапи процесу проектування систем реального часу?
19. Яке призначення програми системи реального часу?
20. Що таке архітектурні процеси систем спостереження і управління, а також систем збору даних?
21. Які основні переваги повторного використання компонентів програмного забезпечення і які проблеми, можуть виникнути при цьому?
22. Які існують типи повторно використовуваних компонентів і які основні етапи процесу їх проектування?
23. Що таке сімейство додатків і чому вони служать ефективним засобом повторного використання програмного забезпечення?
24. Що таке проектні патерни?
25. Основні принципи проектування інтерфейсу користувача?
26. Які п'ять основних стилів взаємодії користувача з програмними системами?
27. Які стилі представлення інформації і в яких випадках доцільне графічне представлення даних?
28. Які основні правила проектування засобів підтримки користувача, вбудованих в програмне забезпечення?
29. Які основні показники зручності використання систем?

## 2-й семестр

### Контрольні питання до 1-го модуля

1. Що таке життєвий цикл програмного забезпечення? Які існують моделі життєвого циклу, назвіть їх особливості?
2. Що таке методологічна основи технологій розробки програмного забезпечення?
3. Які існують методи структурного аналізу і проектування програмного забезпечення?
4. Які існують методи об'єктно-орієнтованого аналізу та проектування програмного забезпечення?
5. Що таке мова моделювання UML? Які основні діаграми використовуються в UML?

### Контрольні питання до 2-го модуля

1. Які етапи документування розробки програмного забезпечення?
2. Що таке планування проекту?
3. Чи є інспекція програм, ефективним методом виявлення помилок в програмах?
4. Чому статичний аналіз програм є одним з основних методів верифікації?
5. Розробка програм за допомогою методу "чиста кімната" і чому цей метод є ефективним?
6. Які методи тестування використовуються для виявлення програмних помилок?
7. Які основні принципи тестування інтерфейсів?
8. Які особливості покомпонентного тестування і тестування процесу збірки об'єктно-орієнтованих систем?
9. Які CASE-засоби вживані для тестування?
10. Що таке атестація програмного забезпечення?

### Контрольні питання до 3-го модуля

1. Що таке процес управління проектом?
2. Що означає планування проекту і графік робіт?
3. Що таке управління ризиками?
4. Які основні моделі організації людської пам'яті, моделі вирішення проблем і мотивації, а також механізми їх застосування в практичній роботі керівників проектами по створенню програмного забезпечення?
5. Які основні проблеми командної роботи?
6. Які проблеми, пов'язані з підбором і збереженням персоналу в організації, яка займається розробкою і супроводом програмного забезпечення?
7. Що таке модель Р-СММ?
8. Як оцінюється собівартість і призначається ціна програмного продукту?
9. Що таке системи оцінювання виробництва програмного продукту?

10. Що потрібно для правильної оцінки вартості програмного забезпечення і для створення графіка робіт?
11. Що таке алгоритмічні моделі визначення вартості?
12. Які принципи роботи моделі СОСОМО 2 для алгоритмічної оцінки вартості?
13. Які основні процеси управління якістю і основні складові процесу управління якістю?
14. Які застосовуються стандарти в процесі управління якістю?
15. Які метричні показники програмного забезпечення і які відмінності між прогнозованими і контролюючими показниками?
16. Яка необхідність систем вимірювання при оцінці показників якості і обмеження в процесі вимірювання показників програмного забезпечення?
17. Які основні принципи вдосконалення виробництва програмного забезпечення?
18. Які чинники, впливають на якість програмного продукту і продуктивність розробників програмного забезпечення?
19. Що таке модель оцінки рівня розвитку виробництва? Чому вдосконалення на основі моделі оцінки рівня розвитку виробництва може бути застосоване до виробництва не всіх типів програмного забезпечення?
20. Що таке складові функціональної надійності?
21. Що таке критичні системи?
22. Що таке працездатність і безвідмовність системи?
23. Що таке безпека системи?
24. Що таке захищеність системи?
25. Які основні засоби досягнення функціональної надійності системи?
26. Які вимоги безвідмовності?
27. Що таке специфікація вимог безпеки?
28. Що таке специфікація вимог захищеності?
29. Які методи розробки програмного забезпечення, допомагають уникнути збоїв в програмній системі?
30. Що таке мінімізація помилок і збоїв?
31. Що таке стійкість до збоїв?
32. Що таке відмовностійка архітектура?
33. Навіщо проектування безпечних систем?
34. Що таке N-варіантне програмування і блоки відновлення?
35. Які переваги і недоліки використання формальних методів верифікації критичних систем?