

**ВИЩИЙ НАВЧАЛЬНИЙ ЗАКЛАД
УНІВЕРСИТЕТ ЕКОНОМІКИ ТА ПРАВА "КРОК"
Коледж економіки, права та інформаційних технологій**

ДОБРИШИН Ю.Є., ЧЕРНОЗУБКІН І.О.

**МЕТОДИЧНІ РЕКОМЕНДАЦІЇ ЩОДО ОРГАНІЗАЦІЇ ТА
ПРОВЕДЕННЯ КОМПЛЕКСНОГО ЕКЗАМЕНУ
(спеціальність 121 «Інженерія програмного забезпечення»)**

Ю.Є. Добришин, Чернозубкін І.О.

Методичні рекомендації щодо організації та проведення комплексного екзамену зі спеціальності для студентів Коледжу економіки, права та інформаційних технологій спеціальності: 121 «Інженерія програмного забезпечення» - К.: Університет економіки та права "КРОК". 2019. –48 с.

Методичні рекомендації щодо організації та проведення державних екзаменів містять вимоги до комплексного екзамену зі спеціальності, процедуру та правила проведення екзамену, програму дисциплін та перелік питань щодо підготовки до комплексного екзамену зі спеціальності.

РОЗГЛЯНУТО І СХВАЛЕНО

Педагогічною радою Коледжу економіки, права та інформаційних технологій

Протокол № 1 від 31 серпня 2019 р.

© Добришин Ю.Є., 2019

© Чернозубкін І.О., 2019

© Університет економіки та права "КРОК", 2019

© Коледж економіки, права та інформаційних технологій, 2019

ЗМІСТ

ВСТУП.....	4
ВИМОГИ ДО КОМПЛЕКСНОГО ЕКЗАМЕНУ ЗІ СПЕЦІАЛЬНОСТІ.....	2
ПРОЦЕДУРА ПРОВЕДЕННЯ КОМПЛЕКСНОГО ЕКЗАМЕНУ.....	2
ПРАВИЛА ПРОВЕДЕННЯ КОМПЛЕКСНОГО ЕКЗАМЕНУ ЗІ СПЕЦІАЛЬНОСТІ.....	3
ПРОГРАМА ДИСЦИПЛІН ТА ПЕРЕЛІК ПИТАНЬ ДЛЯ ПІДГОТОВКИ ДО КОМПЛЕКСНОГО ЕКЗАМЕНУ ЗІ СПЕЦІАЛЬНОСТІ.....	4
1. ОСНОВИ ПРОГРАМУВАННЯ ТА АЛГОРИТМІЧНІ МОВИ.....	4
1.1. Основний зміст дисципліни.....	4
1.2. Програма навчальної дисципліни.....	4
1.3. Перелік питань з дисципліни «Основи програмування та алгоритмічні мови».....	7
2. ОБ'ЄКТНО-ОРІЄНТОВАНЕ ПРОГРАМУВАННЯ.....	8
2.1. Основний зміст дисципліни.....	8
2.2. Програма навчальної дисципліни.....	9
2.3. Перелік питань з дисципліни «Об'єктно-орієнтоване програмування»	14
3. ОСНОВИ ПРОГРАМНОЇ ІНЖЕНЕРІЇ.....	17
3.1. Основний зміст дисципліни.....	17
3.3. Перелік питань з дисципліни «Основи програмної інженерії».....	19
4. ТЕХНОЛОГІЯ СТВОРЕННЯ ПРОГРАМНИХ ПРОДУКТІВ.....	20
4.1. Основний зміст дисципліни.....	20
4.3. Перелік питань з дисципліни «Технологія створення програмних продуктів».....	22
5. АДМІНІСТРУВАННЯ ПРОГРАМНИХ СИСТЕМ І КОМПЛЕКСІВ.....	24
5.1 Основний зміст дисципліни.....	24
5.2. Програма навчальної дисципліни.....	26
5.3. Перелік питань з дисципліни «Адміністрування програмних систем і комплексів».....	28
6. ТЕХНОЛОГІЇ ЗАХИСТУ ІНФОРМАЦІЇ.....	31
6.1. Основний зміст дисципліни.....	31
6.2. Програма навчальної дисципліни.....	32
6.3. Перелік питань з дисципліни «Технології захисту інформації».....	34
ОЦІНЮВАННЯ РЕЗУЛЬТАТІВ КОМПЛЕКСНОГО ЕКЗАМЕНУ ЗІ СПЕЦІАЛЬНОСТІ.....	35
СПИСОК ЛІТЕРАТУРИ.....	40

ВСТУП

Комплексний екзамен зі спеціальності проводиться після завершення студентами навчання у Коледжі економіки, права та інформаційних технологій для визначення освітньо-кваліфікаційного рівня – молодший спеціаліст.

Комплексний екзамен зі спеціальності є формою кваліфікаційних випробувань, які об'єктивно і надійно визначають рівень освітньої та професійної підготовки випускників зі спеціальності 121 «Інженерія програмного забезпечення»

Мета комплексного екзамену зі спеціальності – виявлення рівня засвоєння теоретичних знань та практичних навичок студентів, їх відповідності вимогам освітньо-кваліфікаційної характеристики.

До складання комплексного екзамену зі спеціальності допускаються студенти, які виконали вимоги навчального плану.

Комплексний екзамен зі спеціальності проводиться як комплексна перевірка знань студентів *з таких навчальних дисциплін:*

1. Об'єктно-орієнтоване програмування.
2. Основи програмування та алгоритмічної мови.
3. Основи програмної інженерії.
4. Технологія створення програмних продуктів.
5. Адміністрування програмних систем та комплексів.
6. Технологія захисту інформації.

ВИМОГИ ДО КОМПЛЕКСНОГО ЕКЗАМЕНУ ЗІ СПЕЦІАЛЬНОСТІ

Комплексний екзамен зі спеціальності передбачає виконання комплексу кваліфікаційних завдань і є адекватною формою щодо об'єктивного визначення рівня якості освітньої та професійної підготовки випускників Коледжу економіки, права та інформаційних технологій.

Інформаційною базою формування засобів об'єктивного контролю ступеня досягнення кінцевих цілей освітньо-професійної підготовки є змістові модулі програм навчальних дисциплін, що формують систему професійних компетенцій молодшого спеціаліста.

Під час комплексного екзамену зі спеціальності студент повинен продемонструвати знання та виявити вміння, що складають комплекс професійних компетенцій молодшого спеціаліста, визначених стандартами щодо спеціальності 121 «Інженерія програмного забезпечення».

ПРОЦЕДУРА ПРОВЕДЕННЯ КОМПЛЕКСНОГО ЕКЗАМЕНУ

Комплексний екзамен зі спеціальності освітньо-кваліфікаційного рівня «молодший спеціаліст» складається з двох етапів. Перший етап включає письмові відповіді на питання, що зазначаються у екзаменаційних білетах (30 хвилин), другий передбачає усні відповіді на питання білету членам екзаменаційної комісії, яка складається з трьох осіб (провідних фахівців-викладачів), призначених наказом по Університету «КРОК»

На першому етапі студенти отримують індивідуальні стандартизовані бланки білетів та бланки для оцінювання відповідей. Перед початком екзамену члени екзаменаційної комісії пояснюють порядок заповнення бланку оцінювання, на якому студенти зазначають групу, прізвище, ініціали, номер білету, дату проведення екзамену.

Екзаменаційні білети складаються з чотирьох питань, кожне з яких відповідає змісту питань навчальних дисциплін, з яких здійснюється перевірка знань студентів.

Після закінчення першого етапу, студент відповідає усно членам

екзаменаційної комісії на питання екзаменаційного білету. Тривалість відповіді повинна складати до 15 хвилин.

Після перевірки знань студентів, екзаменаційна комісія на закритому засіданні відкритим голосуванням приймає рішення про оцінювання знань, виявлених студентами при складанні комплексного екзамену зі спеціальності.

Результати складання комплексного екзамену зі спеціальності визначають оцінками «відмінно», «добре», «задовільно», «незадовільно» і оголошують у день їх проведення після оформлення протоколу засідання екзаменаційної комісії.

Студенту, який успішно склав екзамен, рішенням екзаменаційної комісії надається право отримання диплому кваліфікаційного рівня – молодший спеціаліст. У разі якщо студент отримав незадовільну оцінку, під час проведення комплексного екзамену зі спеціальності, він рішенням екзаменаційної комісії подається на відрахування з Коледжу економіки, права та інформаційних технологій.

ПРАВИЛА ПРОВЕДЕННЯ КОМПЛЕКСНОГО ЕКЗАМЕНУ ЗІ СПЕЦІАЛЬНОСТІ

1. Реєстрація учасників комплексного екзамену зі спеціальності починається за 30 хвилин до його початку, запізнення є неприпустимими.

2. Студенти до початку комплексного екзамену зі спеціальності мають надати секретарю комісії залікову книжку.

3. Під час екзамену забороняється:

- використовувати будь-яку літературу, записи, мобільні телефони;
- підказувати, говорити, надавати консультації студентами один одному;
- консультуватися з членами екзаменаційної комісії;
- виходити з аудиторії з будь-якими робочими аркушами.

У разі недотримання перелічених вище вимог студент буде видалений з аудиторії без попередження й права продовження складання екзамену.

Неприпустимим є продовження роботи над завданням білету після оголошення про її закінчення.

ПРОГРАМА ДИСЦИПЛІН ТА ПЕРЕЛІК ПИТАНЬ ДЛЯ ПІДГОТОВКИ ДО КОМПЛЕКСНОГО ЕКЗАМЕНУ ЗІ СПЕЦІАЛЬНОСТІ

1. ОСНОВИ ПРОГРАМУВАННЯ ТА АЛГОРИТМІЧНІ МОВИ

1.1. Основний зміст дисципліни

Метою дисципліни є формування в студентів знань структур даних мов програмування, алгоритмів розв'язування типових задач та умінь і практичних навичок, необхідних для використання сучасних засобів розробки програм різними мовами та в різних середовищах програмування.

Завданням дисципліни є оволодіння студентами теоретичними знаннями і набуття ними практичних навичок з розробки програм з використанням різних мов та середовищ програмування.

Предметом дисципліни є базові поняття програмування: структури даних, алгоритми, основні алгоритмічні структури, семантика та синтаксис високорівневих мов програмування, функціональні можливості найбільш поширених середовищ розробки програм.

Після вивчення дисципліни студенти мають знати основні принципи розробки, написання і налагодження простих програм з використанням сучасних інструментальних засобів.

Студенти повинні вміти застосовувати отримані навички для знаходження розв'язку сформульованої задачі: розробити алгоритм, написати програму мовою програмування високого рівня, налагодити програму й одержати результат в заданому інструментальному середовищі, систематизувати сучасні засоби розробки програм, аналізувати тенденції розвитку засобів і систем проектування програм.

1.2. Програма навчальної дисципліни

Вступ. Предмет дисципліни. Базові поняття програмування.

Модуль 1. Організація програм. Прості типи даних. Керування потоком обчислень

Тема 1. Поняття алгоритму. Способи опису алгоритму. Основні алгоритмічні структури

Інформація в пам'яті комп'ютера. Зображення чисел у комп'ютері. Поняття алгоритму. Вимоги, яким має задовольняти алгоритм. Алгоритмічна структура розгалуження/ Алгоритмічна структура повторення.

Тема 2. Засоби створення програм. Класифікація мов програмування

Історія розвитку мов програмування. Машинна мова, асемблери, мови програмування високого рівня. Переклад високорівневих програм у машинні коди. Компілятори та інтерпретатори.

Технологія створення програми. Розробка, налагодження, тестування, експлуатація.

Інтегроване середовище розробки Microsoft Visual Studio 2017 та його компоненти: редактор коду, компілятор, компоновальник, налагоджувач, довідкова система.

Створення, налагодження та виконання найпростішої програми.

Тема 3. Прості типи даних. Константи, змінні, вирази. Оператори

Цілочислові типи. Дійсні типи. Символьні типи, мінні. Ім'я (ідентифікатор), тип, значення, область видимості.

Оператори присвоєння, арифметичні оператори.

Тестування і налагодження програм.

Принципи тестування. Засоби тестування і налагодження системи програмування.

Тема 4. Керування обчислювальним процесом

Алгоритмічний вибір альтернатив із двох альтернатив. Умовний оператор if. Вкладеність конструкцій вибору. Поліваріантний вибір. Конструкція case ... switch.

Алгоритмічні конструкції повторення. Цикли із передумовою, цикли із постумовою, цикли з параметрами.

Типові алгоритмічні задачі з використанням циклів.

Модуль 2. Складені типи даних. Функції користувача

Тема 5. Масиви. Вказівники і масиви

Визначення масиву. Адреси, вказівники, посилання і масиви. Адресна арифметика.

Одновимірні та багатовимірні масиви. Динамічні масиви, оператори new, delete, запобігання втрати пам'яті. Ініціалізація масивів.

Тип даних vector.

Найпростіші алгоритми пошуку та впорядкування для масивів.

Тема 6. Рядки і масиви символів

Визначення рядка. Масиви символів. Вказівники і рядки.

Види рядків в мові C++.

Стандартні функції обробки рядків бібліотеки <string>.

Тема 5. Функції користувача в мові C++

Визначення функції. Прототип функції. Виклик функції. Тіло функції. Типи значень, що повертаються функцією. Перевизначення функцій.

Формальні параметри. Фактичні параметри. Передача параметрів за значенням, адресою та посиланням.

Локальні і глобальні змінні. Рекурсивні виклики.

Тема 9. Типи даних користувача. Структури

Оголошення структури. Використання елементів структури. Структури і функції. Функції, що змінюють елементи структури.

Модуль 3. Файли та динамічні структури даних

Тема 9. Робота з файлами. Бінарні та текстові файли

Виведення у файловий потік. Читання із вхідного файлу.

Бінарні файли, текстові файли. Визначення кінця файлу.

Обробка помилок при виконанні файлових операцій.

Тема 10. Алгоритми впорядкування в динамічних структурах даних.

Побудова списків різних типів: стеки, черги, лінійні списки, двонаправлені списки, кільцеві списки.

Організація впорядкування в динамічних структурах даних.

Способи впорядкування на різних відстанях і в різних напрямках: швидке

впорядкування. Порозрядне впорядкування.

Тема 11. Алгоритми пошуку. Розв'язок оптимізаційних задач

Задача пошуку у великих масивах даних.

Послідовний, прискорений пошук, метод транспозиції.

Тема 12. Рекурсивні алгоритми і структури даних

Рекурсивні структури даних. Деревоподібні структури. Бінарне дерево.

Використання рекурсії для побудови дерева. Приклад створення об'єкта-дерево.

Рекурсивні алгоритми пошуку.

1.3. Перелік питань з дисципліни «Основи програмування та алгоритмічні мови»

1. Представлення даних та команд в пам'яті комп'ютера. Поняття алгоритму. Основні алгоритмічні структури. Класифікація мов програмування. Засоби та технологія створення програм.

2. Прості типи даних. Константи, змінні, вирази. Оператори. Тестування і налагодження програм. Перетворення типів даних.

3. Керування обчислювальним процесом. Типові алгоритмічні задачі з використанням циклів.

4. Складені типи даних. Масиви. Вказівники і масиви. Алгоритми пошуку та впорядкування масивів.

5. Рядки, масиви символів. Стандартні функції обробки рядків.

6. Функції користувача в мові C++. Перевизначення функцій. Формальні і фактичні параметри. Передача параметрів за значенням, адресою та посиланням.

7. Локальні і глобальні змінні. Рекурсивні виклики. Приклади використання рекурсії.

8. Типи даних користувача. Структури, масиви структур. Структури і функції.

9. Робота з файлами. Бінарні та текстові файли. Виведення у файловий потік. Читання із вхідного потоку. Обробка помилок при виконанні файлових операцій.

10. Побудова списків різних типів: стеки, черги, лінійні списки, двонаправлені списки, кільцеві списки. Організація впорядкування в динамічних структурах даних.

11. Задача пошуку у великих масивах даних. Послідовний, прискорений пошук, метод транспозиції.

12. Рекурсивні структури даних. Деревоподібні структури. Бінарне дерево. Використання рекурсії для побудови дерева. Рекурсивні алгоритми пошуку.

2. ОБ'ЄКТНО-ОРІЄНТОВАНЕ ПРОГРАМУВАННЯ

2.1. Основний зміст дисципліни

Мета: одержання студентами теоретично-прикладних знань про розробку програм з використанням методів та технологій об'єктно-орієнтованого програмування на платформі .NET.

Завдання:

- навчити проектуванню компонент програмного забезпечення: пояснити основи об'єктно-орієнтованого аналізу завдань, закріпити на прикладах прийоми опису структури і логіки рішення завдань засобами мови C#;

- навчити проектуванню людино-машинного інтерфейсу інформаційних систем: показати використання засобів .NET Framework, в об'єктно-орієнтованих Desktop-застосуваннях, для розробки елементів керування, для введення-виведення інформації та роботи з графікою.

- доповнити отримані у попередніх курсах прийоми реалізації прототипів архітектури програмного забезпечення методами програмування ієрархії класів, забезпечення взаємодії об'єктів через події, використання шаблонних класів тощо, веб-сервісів, тощо;

- показати особливості інсталяції, налаштування та обслуговування програмного забезпечення застосувань, побудованих за принципами ООП.

У результаті вивчення навчальної дисципліни студент повинен **знати:**

- основи об'єктно-орієнтованого аналізу, парадигми об'єктно-орієнтованого програмування, синтаксис мови C# та вбудовані в Microsoft

Visual Studio базові бібліотеки класів .NET Framework, прийоми розробки об'єктно-орієнтованих програм для виконання обчислень та моделювання процесів.

У результаті вивчення навчальної дисципліни студент повинен **вміти:**

аналізувати завдання з програмування і розробляти постановку задачі з використанням методів об'єктно-орієнтованого аналізу;

- використовувати вбудовані бібліотеки .NET Framework, підручники, довідники і ресурси Інтернет, знаходити типові класи при розробці програм;

- проектувати і програмувати власні системи класів мовою C#;

- розробляти графічні інтерфейси користувача засобами C# .NET Framework, з використанням власних класів;

- виконувати редагування тексту програми та її налагодження засобами Microsoft Visual Studio C#;

тестувати розроблені програми.

2.2. Програма навчальної дисципліни

Розділ 1. Основи об'єктно-орієнтованого програмування мовою C# на платформі .Net

Тема 1. Вступ. Мета та завдання дисципліни „Об'єктно-орієнтоване програмування”.

Мета та завдання дисципліни. Структура дисципліни. Місце ООП та технологій .Net серед інших технологій програмування.

Тема 2. Архітектура .NET Framework. Мова програмування C#.

Основні характеристики платформи MS .NET Framework. Стек технологій .Net Framework.

Загальна специфікація мов програмування CLS. Загальномовне середовище виконання CLR. Керований та некерований код. Структура бібліотеки класів FCL. Проміжна мова платформи Microsoft .NET – MSIL (IL).

Збірка як компонент в контексті середовища .NET Framework. Об'єднання керованих модулів у збірку (Assembly). Маніфест збірки (Manifest).

Створення та виконання програми у середовищі .NET Framework. Двохетапна компіляція (Just In Time, JIT) модулів програми. Переваги і недоліки платформи .NET. Структура консольного застосування C#.

Тема 3. Модель типів даних .Net. Стандартні бібліотеки

Базова модель та ієрархія класів .NET. Універсальна система типів CTS (Common Type System) в .NET. Змінні, імена, константи, зона видимості змінних, масиви. Тип-значення, тип-посилання. Вбудовані типи значень C#. Цілочисельний тип даних, типи з плаваючою комою, десятковий тип, логічний тип, символічний тип. Масиви, структури, перерахування у мові C#. Вбудовані типи-посилань, рядки. Особливості операції присвоювання boxing і unboxing. Перетворення типів.

Стандартні бібліотеки, графіка. Класи потоків введення та виведення.

Тема 4. Керування виконанням програми. Обробка виняткових ситуацій.

Оператор присвоєвання. Блок операторів. Оператори розгалуження, вибору, циклів. Оператор foreach. Оператори переходу goto, break, continue.

Обробка виняткових ситуацій.

Тема 5. Об'єкти Windows Forms та сервіси, що ними надаються.

Основні компоненти Windows Forms-застосування – форма, керуючі елементи, властивості, події, методи форм, модальні і немодальні форми.

Управління та події життєвого циклу форми. Обробка подій. Ієрархія класів елементів керування (Controls). Елементи: кнопка (клас Button), прапорець (клас CheckBox), перемикач (клас RadioButton), списки (класи ListBox, ComboBox, CheckedListBox), меню, панелі інструментів. Клавіші швидкого доступу (ShortcutKeys), впливаючі підказки.

Тема 6. Парадигма об'єктно-орієнтованого програмування. Класи, об'єкти, методи

Основні поняття та принципи ООП. Оголошення класу, модифікатори доступу, об'єкти, поля, властивості, методи. Конструктор та деструктор класу. Ключове слово this. Простори імен та директива using. Клас System.Object як

основа ієрархії класів. Перетворення та сумісність типів, тип `object`. Явне і неявне перетворення типів. Операція `is` та `as`.

Способи передачі параметрів в методи: за посиланням, за значенням, змінна кількість параметрів. Модифікатори типів параметрів `in`, `out`, `ref`. `params`.

Перевантаження операцій та методів. Інкапсуляція та властивості.

Тема 7. Успадкування та поліморфізм

Поняття успадкування та реалізація ієрархії класів в мові C#. Успадкування членів батьківських класів, конструкторів. Виклик конструктора базового класу. Послідовність виклику конструкторів. Обмеження успадкування. Сумісність типів при успадкуванні. Закриті класи та методи (`sealed`).

Поліморфізм. Віртуальні методи. Переоголошення, перевизначення та перевантаження функцій батьківських класів. Таблиця віртуальних методів. Раннє та пізнє зв'язування. Перевизначення методу в похідному класі. Механізм виклику віртуального методу.

Клас `System.Object` та тип `object`. Відкриті методи класу `System.Object`. Перевизначення методу `ToString()`, `Equals()` та `GetHashCode()`.

Поняття універсальної функції і універсального класу (`generics`) та їх реалізація у мові C#. Використання та успадкування універсальних класів. Обмеження універсальності.

Взаємодія класів. Успадкування: спеціалізація, специфікація, конструювання, розширення, узагальнення, обмеження. Вкладення: композиція, агрегація. Модель включення-делегування. Ініціалізація вбудованих об'єктів у мові C#. Доступ до членів вбудованих об'єктів.

Тема 8. Статичні класи та методи. Абстрактні класи

Оголошення та використання статичних методів, членів, властивостей, конструктора. Патерн `Singleton` та можливість його використання в багатопоточних процесах. Статичні класи. Члени-константи, тільки для читання.

Абстрактні класи. Застосування абстрактних класів. Поліморфні методи.

Тема 9. Інтерфейси. Множинне успадкування

Загальні відомості про інтерфейси. Інтерфейси та успадкування. Відмінності інтерфейсу від абстрактного класу. Множинне успадкування в C#.

Реалізація інтерфейсів. Звернення до методу через об'єкт типу інтерфейс. Явне задання імені інтерфейсу. Приклад перевизначення методу.

Стандартні інтерфейси .NET. Порівняння об'єктів, інтерфейс IComparer. Ітератори і інтерфейс перерахування IEnumerable. Клонування об'єктів. Поверхнєве і глибоке.

Тема 10. Колекції та узагальнені колекції об'єктів. Ітератор

Поняття колекції та узагальненої колекції об'єктів. Колекції List, ArrayList, Hashtable, Queue, Stack, SortedList, Dictionaries, BitArrays, їх властивості та методи.

Узагальнені колекції List<T>, ArrayList<T>, Hashtable<T>, Queue<T>, Stack<T>, SortedList<T>, Dictionaries<T>, BitArrays<T> їх властивості та методи.

Реалізація інтерфейсів IList, ICollection, IComparer, IEnumerable, IList<T>, ICollection<T>, IComparer<T>, IEnumerable <T>.

Тема 11 . Делегати та зворотній виклик методу класу

Визначення делегата як вказівника на функцію. Клас Delegate його методи і властивості. Створення екземпляру делегату. Виклик делегата. Делегати як параметри методів. Коваріація та контраваріація. Анонімні методи. Узагальнені делегати.

Комбінування делегатів, список викликів. Клас MulticastDelegate. Операції на делегатами. Додавання видалення методів в делегат.

Зворотній виклик (callback). Приклад використання зворотнього виклику. Приклади застосування делегатів.

Тема 12. Події та обробники подій

Узагальнення поняття події. Механізм обміну даними через події. Делегати і події. Оголошення подій. Клас Sender. Клас EventArgs і його нащадки. Вхідні і вихідні аргументи події. Клас Receiver.

Обробник події. Зв'язування обробника з подією. Відключення обробника. Взаємодія об'єктів sender і receiver. Динамічне зв'язування подій з їх обробниками. Програмування подій користувача.

Тема 13. Поняття об'єктно-орієнтованого аналізу, проектування та програмування. Мова UML

Об'єктно-орієнтований аналіз. Предметна область як система об'єктів, які взаємодіють. Поняття об'єктної моделі. Узагальнення і абстрагування. Покрокова конкретизація властивостей об'єктів. Виділення рівнів абстрагування з врахуванням окремих поглядів на модель.

Графічна нотація і метамоделі. Основи процесу розробки засобами UML. Зображення класів мовою UML. Клас-клієнт і клас-постачальник. Ієрархія класів. Батьківський клас і похідні класи, успадкування методів і властивостей.

Діаграми варіантів використання. Діаграми класів, асоціації, атрибути, операції, узагальнення, обмеження, агрегація і композиція. Діаграми послідовності і кооперації, стану, діяльності.

Розділ 2. Базові технології об'єктно-орієнтованого програмування на платформі .Net.

Тема 14. Регулярні вирази

Регулярні вирази. Простір RegularExpressions. Методи і властивості класу Regex. Типові задачі застосування регулярних виразів. Розбір текстів і пошук за зразком. Приклади застосування регулярних виразів.

Тема 15 . LINQ та лямбда-вирази.

Вступ до LINQ. Методи розширення LINQ. Фільтрація, впорядкування, Групування. Запит до масиву, колекції об'єктів. Створення нових об'єктів в запитах: проекція, з'єднання. LINQ за типами джерел даних.

Лямбда-вирази та анонімні методи. Методи розширення лямбда-виразів. Призначення інтерфейсів IEnumerable<T> та IQueryable<T>. Відмінності в реалізації методів-розширень цих інтерфейсів.

Відкладене та негайне виконання запитів LINQ. «Ліниве» (lazy) завантаження.

Тема 16. Серіалізація об'єктів. Обробка виняткових ситуацій

Поняття потоку введення та виведення у мові C#. Робота з бінарними та текстовими файлами у мові C#. Виняткові ситуації. Обробка виняткових ситуацій. Серіалізація об'єктів: бінарна XML, SOAP, JSON.

Тема 17. Web-сервіси WSDL

Використання Веб сервісів для розв'язку проблем функціональної сумісності, обходу брандмауера, складність розробки для віддаленого виклику процедур. Основні поняття технології виклику віддалених процедур за допомогою Веб-сервісів: XML, SOAP, UDDI (каталог сервісів), WSDL. Принципи роботи компоненти архітектури Веб-сервісів.

Розробка клієнтського застосування, яке використовує зовнішні WEB-сервіси. Розробка WEB-сервісу користувача.

Тема 18. Багатопоточність

Багатопоточність у програмах. Клас **System.Threading**. Створення потоку на основі делегатів **ThreadStart**, **ParameterizedThreadStart**. Активні та фонові потоки. Пул потоків. Використання класу **Timer**.

Ресурси, які спільно використовуються. Синхронізація. Оператор **lock**

2.3. Перелік питань з дисципліни «Об'єктно-орієнтоване програмування»

1. Характеристики платформи .NET Framework. Загальна специфікація мов програмування CLS. Загальномовне середовище виконання CLR. Поняття збірки. Компіляція та виконання програми у середовищі .NET Framework.

2. Базова модель та ієрархія класів .NET. Універсальна система типів в .NET. Змінні, імена, константи, зона видимості змінних. Цілочисельні типи даних, типи з плаваючою комою, десятковий, логічний, символний.

3. Масиви, структури, перерахування, рядки у мові C#. Тип-значення, тип-посилання. Перетворення типів, boxing і unboxing.

4. Керування виконанням програми. Обробка виняткових ситуацій.

5. Основні компоненти Windows Forms-застосування. Ієрархія класів елементів керування. Обробка подій.

6. Парадигма об'єктно-орієнтованого програмування. Класи, об'єкти, методи. Конструктор та деструктор класу. Способи передачі параметрів в методи. Перевантаження операцій та методів. Інкапсуляція та властивості.

7. Успадкування та реалізація ієрархії класів в мові C#. Успадкування конструкторів. Сумісність типів при успадкуванні. Закриті класи та методи (sealed).

8. Поліморфізм. Віртуальні методи. Перевизначення та перевантаження функцій. Раннє та пізнє зв'язування. Механізм виклику віртуального методу.

9. Клас System.Object та тип object. Відкриті методи класу System.Object. Перевизначення методу ToString(), Equals() та GetHashCode().

10. Універсальні функції та класи (generics). Використання та успадкування універсальних класів. Обмеження універсальності.

11. Взаємодія класів. Успадкування: спеціалізація, специфікація, конструювання, розширення, узагальнення, обмеження. Вкладення: композиція, агрегація. Модель включення-делегування. Ініціалізація та доступ до членів вбудованих об'єктів у мові C#.

12. Статичні класи, методи, члени, властивості, конструктори. Патерн Singleton.

13. Абстрактні класи. Застосування абстрактних класів. Поліморфні методи.

14. Інтерфейси та успадкування. Стандартні інтерфейси .NET: IComparer, IEnumerable. Клонування об'єктів.

15. Колекції об'єктів. Колекції List, ArrayList, HashTable, Queue, Stack, SortedList, Dictionaries, BitArrays, їх властивості та методи. Реалізація інтерфейсів IList, ICollection, IComparer, IEnumerable.

16. Узагальнені колекції. Колекції List<T>, ArrayList<T>, HashTable<T>, Queue<T>, Stack<T>, SortedList<T>, Dictionaries<T>.

BitArrays<T> їх властивості та методи. Реалізація інтерфейсів IList<T>, ICollection<T>, IComparer<T>, IEnumerable <T>.

17. Визначення, виклик делегата. Делегати як параметри методів. Коваріація та контраваріація. Анонімні методи. Узагальнені делегати. Комбінування делегатів. Операції на делегатами.

18. Схема зворотнього виклику методу (callback), як приклад застосування делегатів.

19. Події і делегати. Механізм обміну даними через події. Класи Sender, EventArgs, Receiver. Зв'язування обробника з подією. Взаємодія об'єктів sender і receiver. Програмування подій користувача.

20. Об'єктно-орієнтований аналіз. Поняття об'єктної моделі. Узагальнення і абстрагування. Покрокова конкретизація властивостей об'єктів. Виділення рівнів абстрагування з врахуванням окремих поглядів на модель.

21. Проектування застосування засобами UML. Графічна нотація і метамоделі мови. Зображення класів мовою UML. Класи клієнт і постачальник. Ієрархія класів, успадкування методів і властивостей.

22. Діаграми варіантів використання. Діаграми класів, асоціації, атрибути, операції, узагальнення, обмеження, агрегація і композиція. Діаграми послідовності і кооперації, стану, діяльності.

23. Регулярні вирази. Простір імен RegularExpressions. Методи і властивості класу Regex. Типові задачі застосування регулярних виразів.

24. LINQ. Методи розширення, фільтрація, впорядкування, групування. Запит до масиву, колекції об'єктів. Створення нових об'єктів в запитах: проекція, з'єднання. LINQ за типами джерел даних. Відкладене та негайне виконання запитів LINQ. «Ліниве» (lazy) завантаження.

25. Лябда-вирази та анонімні методи. Методи розширення лябда-виразів. Призначення інтерфейсів IEnumerable<T> та IQueryable<T>. Відмінності в реалізації методів-розширень цих інтерфейсів.

26. Потоки введення та виведення у мові С#. Робота з бінарними та текстовими файлами у мові С#. Обробка виняткових ситуацій. Серіалізація об'єктів: бінарна, XML, SOAP, JSON.

27. Принципи роботи застосувань та основні поняття технології WSDL Веб-сервісів. Розробка WEB-сервісу користувача.

28. Багатопоточність. Клас System.Threading. Створення потоку на основі делегатів ThreadStart, ParameterizedThreadStart. Активні та фонові потоки. Пул потоків. Використання класу Timer. Ресурси, які спільно використовуються. Синхронізація.

3. ОСНОВИ ПРОГРАМНОЇ ІНЖЕНЕРІЇ

3.1. Основний зміст дисципліни

Мета викладання дисципліни – вивчення фундаментальних теоретичних основ професійної діяльності в галузі розробки програмного забезпечення; отримання практичних навичок ведення професійної діяльності в умовах наближених до реальних.

Завдання вивчення дисципліни – практичне засвоєння методів та засобів програмної інженерії в систематизованому вигляді для їх застосування на процесах проектування, тестування та оцінки якості програмних систем.

В результаті вивчення дисципліни студенти повинні **знати**:

- основні концепції, що лежать в основі процесу створення програмного забезпечення;
- мати уявлення про основні моделі процесу створення програмного забезпечення;
- архітектуру програмного забезпечення та концепції архітектури його проектування;
- мати уявлення про основні етапи процесу об'єктно-орієнтованого проектування;
- основні аспекти проектування інтерфейсу користувача;

- мати загальне уявлення про верифікацію та атестацію програмного забезпечення та знати методи статичної верифікації;
- знати основні задачі, що стоять перед керівником програмного проекту;
- принципи інженерії програмного забезпечення: вимоги, проектування та тестування;
- принципи об'єктної орієнтації;
- об'єктно-орієнтований аналіз з використанням UML;
- засоби та прикладні інтерфейси програмування;
- клієнт-серверну архітектуру програмного забезпечення.

В результаті вивчення дисципліни студенти повинні *вміти*:

- розробляти WEB-додатки;
- розробляти зрозумілі, стислі, повністю формальні вимоги для розширення існуючих систем, засновані на реальних потребах користувачів та інших зацікавлених осіб;
- використовувати принципи проектування та шаблони при проектуванні;
- створювати UML-діаграми класів, що моделюють аспекти домену та архітектури програмного забезпечення;
- створювати UML-діаграми послідовностей та автомати станів, коректно моделюючих поведінку системи;
- реалізовувати простий графічний користувацький інтерфейс системи;
- використовувати прості методи вимірювання програмного забезпечення;
 - демонстрування розуміння ролі інженерії програмного забезпечення.

3.2. Програма навчальної дисципліни

Вступ. Програмна інженерія як фах.

Розділ 1. Проектування програмного забезпечення

Тема 1.1. Основи проектування.

Тема 1.2. Ключові питання проектування.

Тема 1.3. Структура і архітектура програмного забезпечення.

Тема 1.4. Аналіз якості та оцінка програмного дизайну.

Тема 1.5. Нотації проектування.

Тема 1.6. Стратегії та методи проектування програмного забезпечення.

Розділ 2. Конструювання програмного забезпечення

Тема 2.1. Основи конструювання.

Тема 2.2. Управління конструюванням.

Розділ 3. Тестування програмного забезпечення

Тема 3.1. Основи тестування.

Тема 3.2. Рівні тестування.

Тема 3.3. Техніки тестування.

Тема 3.4. Вимірювання результатів тестування.

Тема 3.5. Процес тестування.

3.3. Перелік питань з дисципліни «Основи програмної інженерії»

1. Життєвий цикл програмного забезпечення.
2. Моделі життєвого циклу програмного забезпечення.
3. Гнучка розробка програмного забезпечення (Agile software development)
4. Методологія SCRUM та її особливості.
5. Методологія Kanban та її особливості.
6. Методологія DevOps (development + operations) та її особливості.
7. Структурна схема програмної системи / програмного комплексу.
8. Функціональна схема.
9. Забезпечення якості (Quality Assurance).
10. Контроль якості (Quality Control).
11. Метрики щодо забезпечення якості.
12. Тестування програмного забезпечення (Software Testing)
13. Верифікація та валідація.

14. Якість програмного забезпечення.
15. Характеристики якості програмного забезпечення
16. Життєвий цикл дефекту.
17. Тестовий випадок (Test Case).
18. Баг(дефект) репорт (Bug Report)
19. Серйозність (Severity) дефекту та її градація.
20. Пріоритет (Priority) дефекту та його градація
21. План Тестування (Test Plan) та його види.
22. Покриття тестами (Test Coverage)
23. Тест Дизайн (Test Design)
24. Умови для проведення тестування програмного забезпечення.
25. Основні поняття автоматизованого тестування програмного забезпечення (скрипт тесту (Test Script), тестовий набір(Test Suite), тести для запуску (Test Run), інструмент для автоматизованого тестування(Automation Test Tool)).
26. Призначення та зміст SWEBOOK (Software Engineering Body of Knowledge)
27. Призначення та зміст PMBOOK (Project Management Body of Knowledge)
28. Призначення та зміст BABOK (Business Analysis Body of Knowledge)

4.ТЕХНОЛОГІЯ СТВОРЕННЯ ПРОГРАМНИХ ПРОДУКТІВ

4.1. Основний зміст дисципліни

Мета навчальної дисципліни: формування у майбутніх техніків-програмістів сучасного рівня інформаційної та програмістської культури, оволодіння основними принципами створення програмного забезпечення; набуття практичних навичок самостійного написання якісного коду для розв'язання різноманітних задач у практичній діяльності.

Програма дисципліни передбачає навчання в формі лекцій та практичних робіт. Для практичного засвоєння основних тем дисципліни практичні роботи

проводяться з застосуванням комп'ютерів, локальних мереж та мережі Internet у комп'ютерних класах.

Завданням вивчення дисципліни є формування теоретичних знань та практичних навичок у відповідності з поставленою метою.

У результаті вивчення навчальної дисципліни:

Студент повинен знати:

- мету конструювання програмного забезпечення;
- місце конструювання в програмній інженерії;
- необхідність попередніх умов при конструюванні програмного забезпечення;
- основні концепції та методики проектування при конструюванні;
- ефективне використання класів;
- характеристики якості програмного забезпечення;
- стратегій інтеграції програмного забезпечення;
- поняття рефакторінгу;
- оцінки продуктивності коду;
- інструменти програмування;

Студенти повинні вміти:

- ефективно планувати та проектувати конструювання якісного програмного забезпечення;
- розробляти якісні інтерфейси та методи класів;
- створювати власний механізми тверджень і виключень;
- використовувати відлагоджувальні засоби;
- оцінювати продуктивність програмного забезпечення.

4.2. Програма навчальної дисципліни

Розділ 1. Проектування програмного забезпечення

Тема 1.1. Важливість виконання попередніх умов. Визначення типу програмного забезпечення. Попередні умові, пов'язані з визначенням проблеми. Попередні умови, пов'язані з розробкою умов. Попередні умови,

пов'язані з розробкою архітектури. Час, який слід присвятити виконанню попередніх умов. Основи моделювання при конструюванні.

Тема 1.2. Проблеми, пов'язані з проектуванням ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ. Основні концепції проектування. Компоненти проектування: евристичні принципи. Методики проектування. Коментарі з приводу популярних методологій.

Тема 1.3. Основи класів: абстрактні типи даних. Якісні інтерфейси класів. Питання проектування та реалізації. Причини створення класів. Аспекти, специфічні для мов. Наступний рівень: пакети класів.

Розділ 2. Якість та оптимізація програмного коду

Тема 2.1. Характеристики якості програмного забезпечення. Методики підвищення якості програмного забезпечення. Важливість вибору підходу до інтеграції. Частота інтеграції — поетапна чи інкрементна? Стратегії інкрементної інтеграції. Щоденна збірка та димові тести.

Тема 2.2. Поняття рефакторінгу, окремі види рефакторінга, безпечний рефакторінг, стратегії рефакторінгу стратегії рефакторінгу.

Тема 2.3. Поняття оптимізації коду, оцінка продуктивності коду. Логіка, цикли, зміни типів даних, вирази та методи, переробка коду на низькому рівні.

Тема 2.4. Інструменти для проектування, інструменти для роботи з вихідним кодом, інструменти для роботи з виконавчим кодом, інструменти та середовища.

4.3. Перелік питань з дисципліни «Технологія створення програмних продуктів»

1. Функціональні та експлуатаційні вимоги до програмного продукту.
2. Основні розділи технічного завдання на розробку програмного забезпечення.
3. Ескізний проект на розробку програмного забезпечення.
4. Для чого призначені і яким чином будуються діаграми потоків даних (DFD – Data Flow Diagrams),

5. Для чого призначені і яким чином будуються діаграми «сутність-зв'язок» (ERD – Entity-Relationship Diagrams).
6. Для чого призначені і яким чином будуються діаграм переходів станів (STD – State Transition Diagrams).
7. Технічний проект на розробку програмного забезпечення..
8. Техніки структурних карт (Константайна, Джексона)
9. Для чого призначена і яким чином будується діаграма використання.
10. Для чого призначена і яким чином будується діаграма взаємодії.
11. Для чого призначена і яким чином будується діаграма послідовності.
12. Для чого призначена і яким чином будується кооперативна діаграма.
13. Для чого призначена і яким чином будується діаграма станів.
14. Функціональні види тестування програмного забезпечення.
15. Нефункціональні види тестування програмного забезпечення.
16. Пов'язані зі змінами види тестування програмного забезпечення.
17. Функціональне тестування (Functional testing) програмного забезпечення.
18. Тестування безпеки (Security and Access Control Testing) програмного забезпечення.
19. Тестування взаємодії (Interoperability Testing) програмного забезпечення.
20. Тестування встановлення (Installation testing) програмного забезпечення.
21. Тестування зручності користування (Usability Testing) програмного забезпечення.
22. Тестування на відмову і відновлення (Failover and Recovery Testing) програмного забезпечення.

23. Конфігураційне тестування (Configuration Testing) програмного забезпечення.
24. Тестування продуктивності програмного забезпечення.
25. Тестування навантаженням (Performance and Load Testing) програмного забезпечення .
26. Стресове тестування (Stress Testing) програмного забезпечення.
27. Тестування стабільності або надійності (Stability / Reliability Testing) програмного забезпечення.
28. Об'ємне тестування (Volume Testing) програмного забезпечення.
29. Димове тестування (Smoke Testing) програмного забезпечення.
30. Регресійне тестування (Regression Testing) програмного забезпечення.
31. Тестування збірки (Build Verification Test) програмного забезпечення.
32. Санітарне тестування або перевірка узгодженості / справності (Sanity Testing) програмного забезпечення.

5. АДМІНІСТРУВАННЯ ПРОГРАМНИХ СИСТЕМ І КОМПЛЕКСІВ

5.1 Основний зміст дисципліни

Мета навчальної дисципліни: формування у слухачів сучасного рівня інформаційної та програмістської культури, отримання знань та практичних навиків з виконання операцій з адміністрування програмних систем та комплексів, оволодіння методиками та правилами планування заходів щодо здійснення основних операції з адміністрування операційних систем та баз даних, програмних додатків та мережевих компонентів, розташованих на базі сучасного серверного обладнання та персональних комп'ютерів.

Програма дисципліни передбачає проведення навчання із застосуванням таких видів занять, як лекції та лабораторні роботи. Для засвоєння основних тем дисципліни лабораторні роботи проводяться з застосуванням комп'ютерів, локальних мереж та мережі Internet у комп'ютерних класах.

Завданням дисципліни є формування теоретичних знань та практичних навичок у відповідності з поставленою метою. За результатами вивчення навчальної дисципліни **студенти повинні знати:**

1. Мету та завдання заходів, що здійснюється під час адміністрування автоматизованих інформаційних систем та комплексів.

2. Основні нормативні документи, що визначають порядок технічного обслуговування та супроводження програмного забезпечення автоматизованих інформаційних систем та комплексів.

3. Типові операції, що застосовуються для виконання технічного обслуговування, адміністрування та супроводження програмних систем та комплексів.

4. Методику виконання технологічних операцій з адміністрування програмних компонентів операційних систем та систем керування базою даних, адміністрування мережного обладнання.

5. Технологію відновлення працездатності програмного забезпечення, програмні засоби, що призначені для виконання операцій з відновлення роботи програмних систем та комплексів.

Студенти повинні вміти:

1. Планувати супроводження, здійснювати управління та розробляти перелік технологічних операцій з технічного обслуговування та супроводження автоматизованих програмних систем та комплексів, здійснювати заходи з оцінки якості програмного забезпечення.

2. Виконувати операції з встановлення, налаштування та адміністрування системного та загальносистемного програмного забезпечення сучасних операційної системи.

3. Встановлювати, налагоджувати програмне забезпечення на основі клієнт серверної архітектури.

4. Виконувати практичні роботи з ліквідації збоїв в роботі програмного забезпечення.

5. Здійснювати операції з адміністрування програмного забезпечення сучасних системи керування базою даних (СКБД) SQL Server.

6. Проводити моніторинг продуктивності роботи та підтримувати працездатність програмних систем і комплексів в процесі їх супроводження та системотехнічного обслуговування.

7. Виконувати роботи з адміністрування та супроводження мережного програмного забезпечення автоматизованих систем та комплексів, здійснювати операції з налагодження роботи DNS та DHCP сервера, програмних компонентів локальної мережі.

5.2. Програма навчальної дисципліни

Тема 1. Завдання та мета системного адміністрування.

Мета та перелік заходів, що здійснюється під час адміністрування програмного забезпечення інформаційних систем та комплексів. Поняття системного адміністрування. Перелік нормативних документів, які визначають заходи, що здійснюються під час проведення технічного обслуговування програмного забезпечення та технічного обладнання. Види та вміст комплексу послуг, які належать до робіт з виконання технологічних операцій з адміністрування програмного забезпечення. Перелік осіб та їх посадові обов'язки щодо виконання операцій з адміністрування програмного забезпечення інформаційних систем та комплексів.

Тема 2. Адміністрування програмного забезпечення MS Windows Server

Вимоги щодо встановлення та конфігурування програмного забезпечення MS Windows Server 2012/2016. Порядок встановлення операційної системи Windows Server 2012/2016. Перевірка результатів. Особливості встановлення та конфігурування операційної системи Windows 10 для роботи на робочих станціях. Конфігурування програмного забезпечення операційної системи MS Windows 10 щодо роботи у режимі робочої групи та домену.

Огляд серверних операційних системи Windows. Порядок встановлення та конфігурування активного каталогу операційної системи Windows. Поняття домену, встановлення домену та його налаштування. Засоби централізованого адміністрування, служба каталогу Active Directory (AD), планування та розгортання AD. Створення та адміністрування організаційних підрозділів та облікових записів користувачів.

Тема 3. Моніторинг та відновлення роботи серверного програмного забезпечення ОС Windows.

Аналіз системних журналів операційної системи на наявність помилок. Виявлення причин їх появи та здійснення заходів з їх усунення. Перевірка стану фрагментації дискових масивів, здійснення операцій дефрагментації дискової пам'яті ОС. Аналіз стану завантаженості оперативної пам'яті ОС, пошук вузьких місць мережі, підвищення ефективності роботи мережі. Тестування роботи жорстких дисків системи, визначення збійних блоків та їх відновлення засобами ОС. Виконання резервного копіювання структурних компонентів ОС та їх відновлення в разі пошкодження.

Тема 4. Адміністрування DNS та DHCP Server операційної системи Windows

Поняття DNS та DHCP сервера. Встановлення та конфігурування програмного забезпечення DNS Server MS Windows. Призначення зон DNS Server та робота з ними. Практичне введення робочої станції у домен та з домену. Типові помилки, що виникають під роботи з DNS Server. Служби DHCP. Встановлення та налаштування мережних протоколів. Технологія призначення динамічних адрес користувачів в домен локальної мережі, можливі типові помилки та методики їх виправлення. Перевірка роботи DHCP Server.

Тема.5. Маршрутизація та віддалений доступ, сервер терміналів

Поняття маршрутизації та віддаленого доступу. Налаштування та адміністрування служби маршрутизації та віддаленого доступу. Адміністрування сервера терміналів. Служба ліцензування. Типові помилки, що виникають під час роботи робочої станції у термінальному режимі

Тема.6. Адміністрування екземпляра СКБД MS SQL Server

Вимоги щодо встановлення MS SQL Server. Версії СКБД та особливості їх застосування Фізична та логічна структура СКБД MS SQL Server. Журнальні файли транзакцій, фізичне розміщення файлів СКБД на дисках сервера. Використання архівних журналів та журналів діагностики бази.

Тема.7. Адміністрування компонентів бази даних MS SQL Server

Перевірка розмірів файлів даних та журналів транзакцій. Порядок їх зміни. Перевірка стану індексів бази даних. Виконання практичних робіт щодо їх створення, перебудови та реорганізації. Побудова плану запиту, перевірка вартості виконання запиту. Моніторинг роботи СКБД, основні програмні засоби , що виконують функції моніторингу роботи MS SQL Server. Виконання операцій експорту та імпорту інформації. Резервне копіювання бази даних.

Тема.8. Створення користувачів бази даних MS SQL Server, призначення відповідних прав

Існуючі користувачі бази даних. Способи автентифікації під час роботи з базою даних. Способи створення та призначення користувачів бази даних. Забезпечення зв'язку користувача з базою даних. Призначення прав та ролей користувачів. Вбудовані ролі бази даних та їх використання. Виконання практичних завдань щодо створення користувачів бази та надання їм доступу до роботи з компонентами бази даних.

5.3. Перелік питань з дисципліни «Адміністрування програмних

систем і комплексів»

1. Поняття системного адміністрування.
2. Категорії системних адміністраторів, короткий зміст їх обов'язків
3. Нормативні документи з обслуговування програмного забезпечення та технічного обладнання.
4. Послуги, які належать до робіт з виконання технологічних операцій з адміністрування програмного забезпечення.
5. Супровід програмного забезпечення. Основні операції.
6. Технічні вимоги щодо встановлення програмного забезпечення MS Windows Server. Версії програмного забезпечення MS Windows Server та їх характеристика.
7. Методи інсталяції програмного забезпечення MS Windows Server. Особливості інсталяції та конфігурування.
8. Основні роли MS Windows Server. Порядок їх призначення. Перелік та характеристики програмного забезпечення сервера щодо управління та конфігурування його компонентів.
9. Вимоги до безпеки під час розгортання мережної інфраструктури сервера. Служба каталогів Active Directory, призначення та основні характеристики.
10. Поняття автентифікації та авторизації. Поняття облікової записи, типи облікових записів.
11. Внутрішня структура Active Directory. Порядок призначення імен об'єктів в Active Directory.
12. Поняття робочої групи та доменної моделі. Особливості їх застосування. Переваги та недоліки.
13. Поняття контролеру домену. Особливості призначення імен контролеру домена. Основний та резервний контролер домену.
14. Основні технологічні операції з моніторингу операційної системи Windows. Перевірка стану служб операційного середовища Windows. Основні команди.

15. Моніторинг завантаженості операційної системи Windows. Контроль за станом пам'яті сервера. Перевірка розміру файлу підкачки. Моніторинг продуктивності роботи операційної системи.

16. Контроль за станом завантаженості процесора на серверному обладнанні. Перевірка програм за допомогою програмних утиліт.

17. Перегляд журналів подій та системного журналу безпеки операційної системи Windows. Класифікація та налаштування системних журналів операційної системи.

18. Перевірка характеру подій у журналі безпеки ОС. Основні номери подій журналу безпеки ОС, які потребують перегляду та контролю.

19. Призначення DNS серверу MS Windows. Основні операції з адміністрування DNS серверу. Поняття зон доступу.

20. Перевірка величини завантаження мережного сервера за допомогою програмного забезпечення Performance Monitor.

21. Моніторинг роботи Active Directory сервера. Основні лічильники.

22. Аналіз вільного місця на жорстких дисках. Виконання та дефрагментації дискової пам'яті, очищення тимчасових каталогів операційної системи серверу.

23. Поняття DHCP сервер. Особливості встановлення та адміністрування DHCP сервер. Призначення утиліти командного рядку «Netch».

24. Поняття маршрутизації. Служба маршрутизації та віддаленого доступу. Особливості застосування.

25. Поняття технології NAT. Спеціальні порти для використання NAT.

26. Поняття VPN. Типи VPN з'єднань. Характеристика PPTP – протоколу. Номери портів, для роботи з VPN.

27. Налаштування операційної системи Windows у режимі роботи термінального сервера. Особливості ліцензування віддалених робочих столів.

28. Технічні вимоги щодо встановлення MS SQL Server. Порядок конфігурування та перевірки роботи. Типи автентифікації. Використання

обліковий записи «sa».

29. Поняття клієнт-серверна архітектура. Характеристика програмного забезпечення для забезпечення роботи клієнт-серверної архітектури. Перевірка параметрів з'єднання на сервері бази даних MS SQL Server.

6. ТЕХНОЛОГІЇ ЗАХИСТУ ІНФОРМАЦІЇ

6.1. Основний зміст дисципліни

Метою дисципліни є надання студентам теоретичних та практичних знань та навичок щодо побудови комплексних систем захисту інформації, розробки, дослідженню та застосуванню механізмів захисту інформації, що засновані на використанні програмних зобів, алгоритмів криптографії іншого програмного забезпечення сучасних автоматизованих інформаційно - телекомунікаційних систем.

Завданням дисципліни є отримання студентами поглиблених знань щодо використання організаційних та технічних заходів щодо захисту інформації в автоматизованих інформаційно-телекомунікаційних системах, вивчення та практичне засвоєння основних операцій з адміністрування програмних засобів захисту інформації.

За результатами вивчення дисципліни студенти повинні

знати:

- основні положення законодавства України в галузі захисту інформації;
- основні міжнародні та національні стандарти з безпеки інформаційних систем та програмних комплексів;
- сучасні програмні засоби щодо захисту інформації, методика їх використання та налаштування;
- механізми аутентифікації та авторизації;
- моделі порушника, основні види атак, засоби їх виявлення та усунення;

вміти:

- встановлювати та налаштовувати на роботу спеціалізоване програмне забезпечення щодо захисту інформації;
- здійснювати основні операції з адміністрування програмних засобів захисту інформації;
- забезпечувати обґрунтований підбір програмно-апаратних та програмних засобів для забезпечення необхідного рівня захисту інформації;
- проводити аналіз ефективності прийнятих технічних рішень щодо забезпечення захисту інформації в інформаційних системах, користуватися математичним та статистичним апаратом щодо вирішення інженерних завдань, які виникають під час розробки та дослідження механізмів;

6.2. Програма навчальної дисципліни**Тема 1. Основні поняття і тенденції захисту інформації**

Основні поняття та визначення безпеки. Роль захисту інформаційних системах, умови функціонування підсистеми безпеки в комп'ютерних мережах та системах. Вимоги щодо безпеки системи, ризику безпеки. Послуги безпеки: конфіденційність, цілісність, автентичність.

Тема 2 Нормативна база забезпечення безпеки інформації в інформаційно-телекомунікаційних системах

Загальні положення щодо захисту інформації в комп'ютерних системах від несанкціонованого доступу. Класифікація автоматизованих систем і стандартні функціональні профілі захищеності оброблюваної інформації від несанкціонованого доступу. Методичні вказівки щодо розробки технічного завдання на створення комплексної системи захисту інформації в автоматизованій системі

Тема 3. Ідентифікація и аутентифікація

Поняття ідентифікації та аутентифікації, двофакторний ідентифікація. Загальна процедура ідентифікації і аутентифікації користувача при його доступі

в систему. Методи аутентифікації, їх класифікація. Спеціалізоване обладнання та програмні засоби щодо виконання операцій аутентифікації.

Тема 4. Комп'ютерні віруси та антивірусні програми

Поняття “вірус”, класифікація вірусів, способи зараження автоматизованих систем. Класифікація та характеристика антивірусних програм. Приклади захисту від вірусних загроз.

Тема 5. Захист даних від несанкціонованого доступу.

Способи несанкціонованого зняття інформації з технічних каналів її витоку. Класифікація каналів витоку та методів несанкціонованого зняття інформації. Методи та засоби блокування технічних каналів витоку інформації

Типи та характеристика спеціалізованого обладнання щодо контролю та захисту даних.

Тема 6. Безпека комп'ютерної мережі

Поняття типової мережі підприємства. Класифікація загроз та атак. Існуючі механізми захисту комп'ютерних мереж. Захист інформації на фізичному та каналному рівнях моделі OSI. Заміна MAC адреси, програмні засоби захисту MAC адреси. Мережні аналізатори, методика їх використання.

Основні недоліки безпеки мережного рівня. Захист протоколу ARP. Атаки на протокол ARP. Підміна адреси комп'ютера, шляхом відправлення невірної відповіді. ARP Spoofing, характеристика, механізм реалізації. Протокол IP, атаки на протокол IP, механізми захисту протоколу IP.

Тема 7. Фільтрація трафіку. Мережні екрани.

Базові поняття про мережні екрани. Захист периметру. Завдання, що вирішуються за допомогою мережних екранів. Класифікація мережних екранів. Захисні технології з використанням мережних екранів. Приклади роботи типових мережних екранів.

Тема 8. Криптографічні методи захисту інформації та VPN

Криптографічні методи захисту інформації, характеристика, класифікація. Алгоритми та режими шифрування. Асиметричний алгоритм шифрування. Алгоритми хешування. Протоколи захисту та цілісності IPSec, SSL, TLS, їх

сутність. Віртуальні мережі. Характеристика VPN. Технологія побудови та використання мережі на основі VPN. Основні поняття про цифровий підпис. Функції обчислення підпису. Алгоритми хешування. Нормативні аспекти застосування цифрового підпису. Загальна схема використання електронно-цифровий підписі. Типи ключів.

6.3. Перелік питань з дисципліни «Технології захисту інформації»

1. Значення інформації і її захисту.
2. Основні поняття захисту інформації і інформаційної безпеки.
3. Класифікація методів захисту даних.
4. Загальні положення щодо захисту інформації в КС від несанкціонованого доступу.
5. Класифікація автоматизованих систем і стандартні функціональні профілі захищеності оброблюваної інформації від несанкціонованого доступу.
6. Методичні вказівки щодо розробки технічного завдання на створення комплексної системи захисту інформації в автоматизованій системі.
7. Поняття ідентифікації і аутентифікації.
8. Поняття двухфакторної ідентифікації.
9. Поняття “вірус”, класифікація вірусів, способи зараження автоматизованих систем.
10. Класифікація та характеристика антивірусних програм.
11. Приклад захисту від вірусних загроз.
12. Способи несанкціонованого зняття інформації з технічних каналів її витоку.
13. Класифікація каналів витоку та методів несанкціонованого зняття інформації.
14. Методи та засоби блокування технічних каналів витоку інформації.
15. Поняття типової мережі підприємства.
16. Класифікація загроз та атак.
17. Рівні інфраструктури мережі підприємства.
18. Протоколи мережі, що потребують захисту.

19. Механізми реалізації мережних атак
20. Проблеми безпеки протоколу розширення адресів ARP
21. Призначення, класифікація мережних екранів.
22. Поняття електронно-цифровій підписи. Поняття закритого та відкритого ключа.
23. Нормативні аспекти призначення цифрового підпису.
24. Поняття про віртуальну приватну мережу (VPN). Основні протоколи та їх призначення для роботи у віртуальній мережі. Призначення протоколу IPSec.

ОЦІНЮВАННЯ РЕЗУЛЬТАТІВ КОМПЛЕКСНОГО ЕКЗАМЕНУ ЗІ СПЕЦІАЛЬНОСТІ

Оцінювання рівня якості підготовки студентів здійснюють члени екзаменаційної комісії на основі встановлених правил, принципів, критеріїв, системи і шкали оцінювання.

Об'єктом оцінювання якості підготовки студентів є сукупність здобутих знань, набутих умінь і навичок, сформованої системи професійних компетенцій, відтворених у процесі розв'язання кваліфікаційних завдань.

Оцінювання рівня якості підготовки студентів здійснюється членами екзаменаційної комісії на основі принципів об'єктивності, індивідуальності, комплексності, етичності для оцінювання набутих типових універсальних (базових) і спеціальних професійних компетенцій.

Результати оцінювання рівня якості підготовки молодшого спеціаліста повинні довести, що випускник:

- має концептуальні знання, здобуті у процесі навчання та практичної діяльності, включаючи основи знань сучасних досягнень науки про туризм;
- вміє розв'язувати складні та непередбачувані завдання і проблеми у сфері туризму, що вимагає самостійного збирання та інтерпретації інформації, вибір методів та засобів вирішення, застосування інноваційних підходів.

Оцінювання відповідей на екзаменаційні питання відбувається за 100 бальною шкалою відповідно до наведених нижче таблиці 1 і таблиці 2.

Таблиця розподілу балів за блоками дисциплін

Блоки	Теоретичний	Практичний
	Частина 1 (тестові завдання) Розв'язок тестових завдань	Частина 2 (письмові завдання) Виконання комплексного завдання або розв'язок розрахунково- аналітичної чи ситуаційної задачі
Максимальна кількість балів за окремо виконані частини і завдання	40	60
Максимальна кількість балів за екзамен	100	

Підсумкова оцінка комплексного кваліфікаційного випробування на державному екзамені складається із сумарної оцінки за виконання завдань з кожного блоку.

Рівень якості підготовки бакалавра визначається з використанням систем оцінювання: Європейської кредитно- трансферної системи (ЄКТС) (за шкалою «А», «В», «С», «Б», «Е», «FX», «F»); національної («відмінно», «добре», «задовільно», «незадовільно»); власної системи ВНЗ; комплексної (поєднання національної системи, ЄКТС, системи оцінювання ВНЗ).

Державна комісія перевіряє бланк оцінювання за ключем і зазначає співпадання (правильні відповіді) і розбіжності (помилкові відповіді), підраховує і зазначає кількість правильних відповідей та проставляє оцінку за тестування.

Таблиця 2

Критерії оцінювання

За шкалою ECTS	За кредитно-модульною шкалою Університету	За національною шкалою	Коментар
А	90-100 %	відмінно	<p>Студент надав понад 90% правильних відповідей з <i>тестових завдань</i>.</p> <p>Комплексне <i>ситуаційне завдання</i> виконане повністю, відповідь обґрунтовано, висновки й пропозиції аргументовані й оформлені належним чином.</p>
В	80-89 %	дуже добре	<p>Студент надав понад 84 – 89% правильних відповідей з <i>тестових завдань</i>.</p> <p>Комплексне <i>ситуаційне завдання</i> виконане повністю, але припущено незначних помилок в розрахунках або оформленні; або за умови належного оформлення завдання виконано не менше як на 85%.</p>
С	70-79 %	добре	<p>Студент надав понад 76 – 83% правильних відповідей з <i>тестових завдань</i>.</p> <p>Комплексне <i>ситуаційне завдання</i></p>

За шкалою ECTS	За кредитно-модульною шкалою Університету	За національною шкалою	Коментар
			виконане повністю, але припущено незначних помилок в розрахунках або оформленні; або за умови належного оформлення завдання виконано не менше як на 80%.
D	60-69 %	задовільно	<p>Студент надав понад 68 – 75% правильних відповідей з <i>тестових завдань</i>.</p> <p>Комплексне <i>ситуаційне завдання</i> виконане не менш як на 70% за умови належного оформлення; або не менш як на 80% за умови припущення незначних помилок у розрахунках або оформленні.</p>
E	50-59 %	достатньо	<p>Студент надав понад 60 – 68% правильних відповідей з <i>тестових завдань</i>.</p> <p>Комплексне <i>ситуаційне завдання</i> виконане не менш як на 65% за умови належного оформлення; або не менш як на 75% за умови припущення незначних помилок у розрахунках або оформленні.</p>
FX	25-49 %	незадовільно	Студент надав менше 60%

За шкалою ECTS	За кредитно-модульною шкалою Університету	За національною шкалою	Коментар
			<p>правильних відповідей з <i>тестових завдань</i>.</p> <p>Комплексне <i>ситуаційне завдання</i> виконане або виконане менш як на 65 %, припущені принципові помилки в розрахунках і оформленні.</p>
F	0-25 %		<p>Студент надав менш 55% правильних відповідей з <i>тестових завдань</i>.</p> <p>Комплексне <i>ситуаційне завдання</i> виконане або виконане менш як на 50 %, припущені принципові помилки в розрахунках і оформленні.</p>

За результатами успішного складання комплексного екзамену зі спеціальності екзаменаційна комісія приймає рішення про присвоєння кваліфікації «молодший спеціаліст» і про видачу випускнику диплома державного зразка.

СПИСОК ЛІТЕРАТУРИ
РЕКОМЕНДОВАНА ЛІТЕРАТУРА З ДИСЦИПЛІНИ «ОСНОВИ
ПРОГРАМУВАННЯ ТА АЛГОРИТМІЧНІ МОВИ»:

Базова

1. І.Лехан С.А. «Інформатика. Мова програмування С++. Спецкурс» Шепетівка, «Аспект», 2007. - 159с.
2. Основи програмування та алгоритмічні мови: лабораторний практикум / уклад. О.В. Чебанюк, Ю.О. Міловідов. - К.: Університет економіки та права «КРОК», 2013. -108с.
3. Бублик В.В. Об'єктно-орієнтоване програмування: [Підручник] / В.В. Бублик. – К.: ІТ книга, 2015. – 624 с.: іл. .
4. Т.А. Павловская «С/С++ Программирование на языке высокого уровня», Санкт-Петербург, «Питер», 2007. - 461с.
5. Шаховська, Н. Б. Алгоритми і структури даних: посібник / Н.Б. Шаховська, Р.О. Голощук ; За ред. В.В. Пасічника. - Львів : Магнолія 2006, 2010. -215с.

Допоміжна

1. Бьерн Страуструп. Язык программирования С++. 1104 стр. Бином,
2. Невский Диалект, 2008 г. 1104 с.
3. А.Ахо, Дж.Хопкрофт, Дж.Ульман. Структуры данных и алгоритмы -М: СПб: Киев: Вильямс, 2003. 382с.
4. Грегори, Кейт. Использование Visual С++. - К.: Диалектика, 864 стр.; 2005 г. 882с.
5. Стефенс Д. Р. С++. Сборник рецептов. — КУДИЦ-ПРЕСС, 2007. -624 с.
6. Джесс Либерти, Дзвид Хорват. Освой самостоятельно С++ за 24 часа - 4-е изд. - М: Вильямс, 2007. — 448 с.
7. Анисимов А.Е., Пупышев В.В. Сборник заданий по основам программирования БИНОМ., 2006. 348 с.
8. Н. Культин «С/С++ в задачах и примерах» Санкт-Петербург, «БХВ-Петербург» 2009. - 368с.

9. Герберт Шилдт. С++: базовый курс - 4-е изд. — М.: Вильямс, 2012 - 624с.

РЕКОМЕНДОВАНА ЛІТЕРАТУРА З ДИСЦИПЛІНИ «ОБ'ЄКТНО-ОРІЄНТОВАНЕ ПРОГРАМУВАННЯ»:

Базова

1. Кравець, П.О. Об'єктно-орієнтоване програмування [Текст] : навчальний посібник / П.О.Кравець. – Львів: „Львівська політехніка”, 2012. – 624 с.
2. Павловская, Т. А. С#. Программирование на языке высокого уровня [Текст] : учебник для вузов / Т. А. Павловская. - С.-Пб.: Питер, 2014. – 432 с.
3. Троелсен, Эндрю. Язык программирования С# 2010 и платформа .NET 4.0, 5-е изд. : Пер. с англ. – М. : ООО "И.Д. Вильямс", 2011. — 1392 с.
4. Рихтер Дж. CLR via С#. Программирование на платформе Microsoft .NET Framework 4.0 на языке С#. 3-е изд. – Спб.: Питер, 2012. – 928 с.
5. Албахари, Д., Албахари, Б. С# 6.0. Справочник. Полное описание языка С# 6.0. [Текст] / Джозеф Албахари, Бен Албахари. – М.: ООО " И. Д. Вильямс", 2016. - 1040 с.;
6. Вайсфельд, М. Объектно-ориентированное мышление [Текст] / Мэтт Вайсфельд - : С.-Пб.: Питер, 2014 - 304 с.;

Допоміжна

1. Буч, Г. Объектно-ориентированный анализ и проектирование с примерами приложений на С++ [Текст] / Гради Буч. – С.-Пб.: Бинум, Невский Диалект, 1998 . - 560 с.
2. Маклафлин, Б. Объектно-ориентированный анализ и проектирование [Текст] / Бретт Маклафлин. – С.-Пб.: Питер, 2013. – 608 с.
3. Ватсон, Б. С#4.0 на примерах [Текст] / Бен Ватсон. - СПб.: БХВ-Петербург, 2011. - 608 с:
4. Дейтел, П., Дейтел, Х. Как программировать на Visual С# 2012 [Текст] / Пол Дейтел, Харви Дейтел. - С.-Пб.: Питер, 2014. – 858 с.;
5. Фаулер, М., Скотт, К. UML. Основы [Текст] / Мартин Фаулер, Кендалл Скотт. - СПб: Символ-Плюс, 2002. - 192 с

6. Шилдт, Г. С# 3.0: руководство для начинающих [Текст] : учебное пособие / Ш. Герберт. - 2-е изд. - М. : ООО "И.Д. Вильямс", 2009. - 688 с.

7. Зиборов, В.В. Visual C# 2010 на примерах [Текст] / Виктор Зиборов. – С.-Пб: БХВ-Петербург, 2013. – 432 с.

РЕКОМЕНДОВАНА ЛІТЕРАТУРА З ДИСЦИПЛІНИ «ОСНОВИ ПРОГРАМНОЇ ІНЖЕНЕРІЇ»

Базова

1. Лавріщева К.М. Програмна інженерія.– К. – 2008. – 319 с.

Допоміжна

1. Вигерс К. Разработка требований к программному обеспечению. – СПб: БХВ-Петербург, 2017. – 736 с.

2. Глаголев Вадим. Разработка технической документации. Руководство для технических писателей и локализаторов ПО. – Питер-Пресс, 2017. – 288 с.

3. Алистер Коберн. Современные методы описания функциональных требований к системам. – Лори, 2017. – 288 с.

РЕКОМЕНДОВАНА ЛІТЕРАТУРА З ДИСЦИПЛІНИ «ТЕХНОЛОГІЯ СТВОРЕННЯ ПРОГРАМНИХ ПРОДУКТІВ»

Базова

1. Карпенко М. Ю. Технології створення програмних продуктів та інформаційних систем : навч. посібник / М. Ю. Карпенко, Н. О. Манакова, І. О. Гавриленко ; Харків. нац. ун-т міськ. госп-ва ім. О. М. Бекетова. – Харків : ХНУМГ ім. О. М. Бекетова, 2017. – 93 с.

Допоміжна

4. Вигерс К. Разработка требований к программному обеспечению. – СПб: БХВ-Петербург, 2017. – 736 с.

5. Глаголев Вадим. Разработка технической документации. Руководство для технических писателей и локализаторов ПО. – Питер-Пресс, 2017. – 288 с.

6. Алистер Коберн. Современные методы описания функциональных требований к системам. – Лори, 2017. – 288 с.

**РЕКОМЕНДОВАНА ЛІТЕРАТУРА З ДИСЦИПЛІНИ
«АДМІНІСТРУВАННЯ ПРОГРАМНИХ СИСТЕМ ТА КОМПЛЕКСІВ»**

Базова

1. Microsoft Windows 2012 Полное руководство. – М.: - 2013 1456 с.
2. 20740A Installation, Storage, and Compute with Windows Server 2016, - OFFICIAL MICROSOFT LEARNING PRODUCT- 606 p.
3. 20741B Networking with Windows Server 2016 - OFFICIAL MICROSOFT LEARNING PRODUCT- 408 p.
4. Лимончелли Т., Хоган К., Чейлап С. Системное и сетевое администрирование. Практическое руководство, 2-е издание. – Пер. с англ. – СПб: Символ-Плюс, 2009. – 944 с.
5. Ю А Бродская, Ю. А. Денисов Практика сетевого администрирования – М. - SelfPub; 2018 – 87 с
6. Душан Петкович. MS SQL Server 2008. Руководство для начинающих, 2009 – 743 с.
7. Колисниченко Д. Н. Linux. От новичка к профессионалу. — 6-е изд., перераб. и доп. — СПб.: БХВ-Петербург, 2018. — 672 с.:-
8. Донцов В.П., Сафин И.В. Linux на примерах – СПб-: Наука и Техника, 2017 – 352 С.
9. Кофлер М. Linux. Установка, настройка, администрирование. — СПб.: Питер, 2014. — 768 с.
10. Exam Ref 70-740 Installation, Storage, and Compute with Windows Server 2016 Published with the authorization of Microsoft Corporation by: Pearson Education, Inc. Copyright © 2017 by Craig Zacker

Допоміжна

1. Mastering Windows Server 2016 A comprehensive and practical guide to Windows Server 2016- BIRMINGHAM – MUMBAI, 416 p.

2. Бреснахэн К., Блум Р. Ліпих на праQике. - СПб.: Питер, 2017. - 384 с.: ил. - (Серия «Для профессионалов»).

3. А.И. Бражук. Сетевые средства Linux//<http://www.intuit.ru/department/os/netapplinux>.

РЕКОМЕНДОВАНА ЛІТЕРАТУРА З ДИСЦИПЛІНИ «ТЕХНОЛОГІЇ ЗАХИСТУ ІНФОРМАЦІЇ»

Базова

1. Бурячок В. Л. Технології забезпечення безпеки мережевої інфраструктури. [Підручник] / В. Л. Бурячок, А. О. Аносов, В. В. Семко, В. Ю. Соколов, П. М. Складанний. – К.: КУБГ, 2019. – 218 с.

2. Технології захисту інформації [Електронний ресурс] : підручник для студ. спеціальності 122 «Комп'ютерні науки», спеціалізацій «Інформаційні технології моніторингу довкілля», «Геометричне моделювання в інформаційних системах»/ Ю. А. Тарнавський; КПІ ім. Ігоря Сікорського. – Електронні текстові дані (1 файл: 2,04 Мбайт). – Київ: КПІ ім. Ігоря Сікорського, 2018. – 162 с.

3. Остапов С. Е. Технології захисту інформації : навчальний посібник / С. Е. Остапов, С. П. Євсєєв, О. Г. Король. – Х. : Вид. ХНЕУ, 2013. – 476 с.

4. Рибальський О.В., Хахановський В.Г., Кудінов В.А. Основи інформаційної безпеки та технічного захисту інформації. Посібник для курсантів ВНЗ МВС України. – К.: Вид. Національної академії внутріш. справ, 2012. – 104 с.

5. Бондарев, В. В. Анализ защищенности и мониторинг компьютерных сетей. Методы и средства : учебное пособие / В. В. Бондарев. - Москва : издательство МГТУ им. Н. Э. Баумана, 2017. – 225 с

6. Монаппа К.А. Анализ вредоносных программ / пер. с англ. Д. А. Беликова. – М.: ДМК Пресс, 2019. – 452 с.:

7. Биячуев Т.А. / под ред. Л.Г. Осовецкого Безопасность корпоративных сетей. – СПб: СПб ГУ ИТМО, 2004.- 161 с.

8. Сикорски М., Хониг Э. Вскрытие покажет! Практический анализ

вредоносного ПО. — СПб.: Питер, 2018. — 768 с.:

9. Флєнов М. Е. Компьютер глазами хакера. — 3-е изд., перераб. и доп. — СПб.: БХВ-Петербург, 2012. — 272 с.

Допоміжна

1. Закон України від 25.06.93 № 3323-ХІІ-ВР "Про науково-технічну інформацію".

2. Закон України "Про інформацію". Із змінами, внесеними згідно із Законом від 07.02.2002 № 3047-ІІІ-ВР.

3. Закон України "Про Державну таємницю". Із змінами і доповненнями, внесеними Законом України від 21.09.1999 №1079-ХІV.

4. Закон України від 05.07.94 № 80/94-ВР "Про захист інформації в автоматизованих системах"

5. Закон України "Про Національну програму інформатизації" Із змінами, внесеними згідно із Законом від 13.09.2001 № 2684-ІІІ-ВР.

6. Закон України "Про зв'язок" Із змінами, внесеними згідно із Законом від 04.07.2002 № 36-ІV-ВР.

7. Закон України від 15 грудня 2005 року № 3200-ІV "Про основи національної безпеки України".