

Київський університет імені Бориса Грінченка
Факультет інформаційних технологій та управління
Кафедра інформаційної та кібернетичної безпеки



«ЗАТВЕРДЖУЮ»

Проректор з науково-методичної та
навчальної роботи

О.Б.Жильцов

« 14 » _____ 2018 р.

РОБОЧА ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ
«ПРИКЛАДНІ АСПЕКТИ ТЕХНОЛОГІЙ БЕЗПЕЧНОГО
ПРОГРАМУВАННЯ»

для студентів

| | |
|--------------------|--|
| спеціальності | 125 Кібербезпека |
| освітнього рівня | першого (бакалаврського) |
| освітньої програми | 125.00.01 Безпека інформаційних і комунікаційних систем |

КИЇВСЬКИЙ УНІВЕРСИТЕТ
ІМЕНІ БОРИСА ГРІНЧЕНКА
Ідентифікаційний код 02136554
Начальник відділу
моніторингу якості освіти

Програма № 1246/18
Жильцов
(підпис) _____ (прізвище, ініціали)
« _____ » _____ 20 18 р.

Київ – 2018

Розробник:

Спасітелева Світлана Олексіївна, кандидат фізико-математичних наук, доцент, доцент кафедри комп'ютерних наук і математики Факультету інформаційних технологій та управління Київського університету імені Бориса Грінченка.

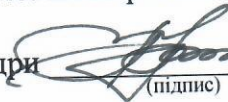
Викладач:

Спасітелева Світлана Олексіївна, кандидат фізико-математичних наук, доцент, доцент кафедри комп'ютерних наук і математики Факультету інформаційних технологій та управління.

Робочу програму розглянуто і затверджено на засіданні кафедри інформаційної та кібернетичної безпеки

Протокол від 13.09.2018 р. № 6

Завідувач кафедри



(підпис)

В.Л. Бурячок

Робочу програму погоджено з гарантом освітньої програми (керівником освітньої програми 125.00.01 Безпека інформаційних і комунікаційних систем)

____.____.20__ р.

Керівник освітньої програми



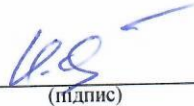
(підпис)

(В.В. Семко)

Робочу програму перевірено

____.____.20__ р.

Заступник директора/декана



(підпис)

І.Ю. Мельник

Пролонговано:

на 20__/20__ н.р. _____ (підпис) _____ (ПІБ), « ____ » ____ 20__ р., протокол № ____

на 20__/20__ н.р. _____ (підпис) _____ (ПІБ), « ____ » ____ 20__ р., протокол № ____

на 20__/20__ н.р. _____ (підпис) _____ (ПІБ), « ____ » ____ 20__ р., протокол № ____

на 20__/20__ н.р. _____ (підпис) _____ (ПІБ), « ____ » ____ 20__ р., протокол № ____

Опис навчальної дисципліни

| Найменування показників | Характеристика дисципліни за формами навчання | |
|--|---|--------|
| | денна | заочна |
| «Прикладні аспекти технологій безпечного програмування» | | |
| Вид дисципліни | обов'язкова | |
| Мова викладання, навчання та оцінювання | українська | |
| Загальний обсяг кредитів / годин | 6 / 180 | |
| Курс | 3 | - |
| Семестр | 5 | - |
| Кількість змістових модулів з розподілом: | 5 | |
| Обсяг кредитів | 6 | - |
| Обсяг годин, в тому числі: | 180 | - |
| Аудиторні | 84 | - |
| Модульний контроль | 10 | - |
| Семестровий контроль | 60 | - |
| Самостійна робота | 26 | - |
| Форма семестрового контролю | екзамен | - |

2. Мета та завдання навчальної дисципліни

Робоча навчальна програма дисципліни «Прикладні аспекти технологій безпечного програмування» є нормативним документом Київського університету імені Бориса Грінченка, який розроблено кафедрою інформаційної та кібернетичної безпеки на основі освітньо-професійної програми підготовки здобувачів першого (бакалаврського) рівня із скороченим терміном навчання відповідно до навчального плану спеціальності 125 Кібербезпека, освітньої програми 125.00.01 Безпека інформаційних і комунікаційних систем.

Робочу навчальну програму укладено згідно з вимогами Європейської кредитної трансферно-накопичувальної системи (ЄКТС) організації навчання.

Програма визначає: обсяги знань, який повинен опанувати здобувач першого (бакалаврського) рівня відповідно до вимог освітньо-кваліфікаційної характеристики; алгоритм вивчення навчального матеріалу дисципліни «Прикладні аспекти технологій безпечного програмування»; необхідне методичне забезпечення; складові та технологію оцінювання навчальних досягнень студентів.

Метою викладання навчальної дисципліни «Прикладні аспекти технологій безпечного програмування» є засвоєння студентами фундаментальних знань в області теорії і практики безпечного програмування із застосуванням процедурної та об'єктно-орієнтовної технологій програмування.

Завдання навчальної дисципліни полягає у формуванні теоретичних знань та практичних умінь у сфері розробки безпечного програмного забезпечення та набуття таких компетентностей:

Загальні компетентності

КЗ-2: Знання та розуміння предметної області, вміння виявляти, ставити та вирішувати проблеми за професійним спрямуванням.

Фахові компетентності спеціальності

КФ-3: Здатність до використання програмних та програмно-апаратних комплексів засобів захисту інформації в інформаційно-телекомунікаційних (автоматизованих) системах.

3. Результати навчання за дисципліною

У результаті вивчення навчальної дисципліни «Прикладні аспекти технологій безпечного програмування» студент повинен

знати:

- теорію та методи безпечного програмування для розв'язування прикладних задач та створення програмного забезпечення систем ІБ;
- етапи життєвого циклу розробки безпечного програмного забезпечення;
- основи процедурного та об'єктно-орієнтованого програмування із застосуванням принципів безпечного програмування;
- методи проектування та реалізації систем ІКБ;

уміти:

- застосовувати сучасні технології безпечного програмування в системах інформаційної та кібербезпеки;
- обирати відповідну технологію програмування, виконувати аналіз специфікації задач;
- виконувати аналіз програмного забезпечення з метою пошуку, ідентифікації, виявлення та усунення помилок програмування та вразливостей;
- виконувати декомпозицію (структурування) алгоритму, розробляти структурні схеми програм із відображенням різноманітних зв'язків між компонентами та визначенням їх функцій;
- визначати класи та створювати об'єкти при застосуванні об'єктно-орієнтованих технологій програмування;
- виконувати модифікацію програм, додаючи, змінюючи або усуваючи окремі їх компоненти з метою зміни їх функцій або спроможності їх функціонування в нових технічних умовах;
- застосовувати прийоми проектування та створення власних захищених програм за допомогою IDE Visual C#;
- забезпечувати працездатність всієї програми, виконуючи індивідуальне та інтегральне тестування, а також застосовуючи засоби безпечного програмування;
- виконувати аналіз зловмисного програмного коду.

Студенти мають досягти наступних **програмних результатів навчання:**

ПРз-3: забезпечувати процеси захисту інформаційно-телекомунікаційних (автоматизованих) систем шляхом встановлення та коректної експлуатації програмних та програмно-апаратних комплексів засобів захисту; забезпечувати функціонування спеціального програмного забезпечення, щодо захисту даних від руйнуючих програмних впливів, руйнуючих кодів в інформаційних, інформаційно-телекомунікаційних (автоматизованих) системах; - виконувати розробку експлуатаційної документації на КЗЗ.

ПРз-10: аналізувати та визначати можливість застосування технологій, методів та засобів технічного захисту інформації; аналізувати та визначати можливість застосування технологій, методів та засобів криптографічного захисту інформації.

ПРз-11: забезпечувати процеси моніторингу доступу до ресурсів і процесів ІТС; забезпечувати конфігурування та функціонування систем моніторингу ресурсів та процесів в ІТС.

ПРз-12: виконувати впровадження та підтримку систем виявлення вторгнень та використовувати комплекси захисту для забезпечення необхідного рівня захищеності інформації в інформаційних, інформаційно-телекомунікаційних та автоматизованих системах; аналізувати та впроваджувати системи захисту від зловмисних програмних кодів.

4. Структура навчальної дисципліни

Тематичний план для денної форми навчання

| Назви змістових модулів і тем | Розподіл годин між видами робіт | | | | | |
|---|---------------------------------|------------|-----------|-------------|---------------|-------------|
| | Усього | Аудиторні: | | | | |
| | | Лекції | Практичні | Лабораторні | Індивідуальні | Самостійна. |
| Змістовий модуль 1. Технології безпечного програмування | | | | | | |
| Тема 1. Основні принципи безпечного програмування | 4 | 2 | | | | 2 |
| Тема 2. Характеристика технології .Net Framework. | 8 | 2 | 2 | 2 | | 2 |
| Тема 3. Базові поняття C#. Данні програми. Безпека даних. | 8 | 2 | 2 | 2 | | 2 |
| Тема 4. Управління потоком виконання програми | 8 | 2 | 2 | 2 | | 2 |
| Модульний контроль | 2 | | | | | |
| Разом | 30 | 8 | 6 | 6 | | 8 |
| Змістовий модуль 2. Технологія структурного процедурного програмування | | | | | | |
| Тема 5. Структурний підхід в програмуванні. Структуровані типи даних. | 12 | 2 | 4 | 4 | | 2 |
| Тема 6. Розбиття програми на підпрограми. Методи. | 10 | 2 | 2 | 2 | | 2 |
| Модульний контроль | 2 | | | | | |
| Разом | 22 | 4 | 6 | 6 | | 4 |
| Змістовий модуль 3. Технологія об'єктно-орієнтованого програмування | | | | | | |
| Тема 7. Принципи об'єктно-орієнтованого програмування. Класи та об'єкти. | 12 | 2 | 4 | 4 | | 2 |
| Тема 8. Похідні класи. | 8 | 2 | 2 | 2 | | 2 |
| Модульний контроль | 2 | | | | | |
| Разом | 22 | 4 | 6 | 6 | | 4 |
| Змістовий модуль 4. Створення базових елементів графічного інтерфейсу користувача | | | | | | |
| Тема 9. Проектування та розробка основних компонент графічного інтерфейсу користувача (GUI) | 10 | 2 | 2 | 4 | | 2 |
| Тема 10. Класи стандартних елементів управління бібліотеки класів Windows Forms. | 14 | 2 | 2 | 8 | | 2 |
| Модульний контроль | 2 | | | | | |
| Разом | 26 | 4 | 4 | 12 | | 4 |
| Змістовий модуль 5. Розробка додатків в інтегрованих середовищах. Побудова керованого та захищеного коду за допомогою Windows Forms, ADO.NET, WPF, ASP.NET | | | | | | |
| Тема 11. Засоби збереження даних. DataGridView, DB | 12 | 2 | 2 | 6 | | 2 |
| Тема 12. Засоби для створення веб-додатків - ASP.NET | 6 | | | 2 | | 4 |
| Модульний контроль | 2 | | | | | |
| Разом | 20 | 2 | 2 | 8 | | 6 |
| Семестровий контроль | 30 | | | | | |
| Курсова робота | 30 | | | | | |
| Усього годин | 180 | 22 | 24 | 38 | | 26 |

5. Програма навчальної дисципліни

Змістовий модуль 1. Технології безпечного програмування.

Вступ. Поняття безпечного програмування. Вразливості програмного коду. Властивості безпечного програмного забезпечення. Головні принципи розробки безпечного та надійного коду. Життєвий цикл розробки безпечного програмного забезпечення (Security Development Lifecycle, SDL) . Особливості SDL. Специфікація ПЗ (вимоги безпеки). Проектування ПЗ (моделювання загроз, аналіз атак, перевірка вхідних даних, границі довіри). Методи проектування ПЗ: структурні методи, об'єктно-орієнтовані методи. Реалізація ПЗ (глибинний захист коду). Тестування та налагодження ПЗ (аудит коду, огляд області атак). Експлуатація та супроводження ПЗ. Методи аналізу коду. Протоколи та стандарти захисту програм. Правила написання безпечного коду .Net Framework. Базові елементи мови на прикладі мови C#.

Змістовий модуль 2. Технологія структурного процедурного програмування.

Модулі. Структурний підхід в програмуванні. Типи структур управління: послідовна структура, структура вибору, структура повторення. Структуровані типи даних. Безпечний доступ до структурованих даних. Лінійні структури даних: список, стек, черга. Розбиття програми на підпрограми. Методи. Параметри методів. Безпека доступу до методів.

Змістовий модуль 3. Технологія об'єктно-орієнтованого програмування.

Абстрактні типи даних. Класи та об'єкти. Захист внутрішньої структури об'єкту. Конструктори та деструктори. Основні операції з об'єктом. Властивості. Перевантаження операторів та методів класу. Типи відношень між класами. Відношення успадкування. Правила створення похідних класів. Віртуальні методи. Абстрактні методи. Індексатори. Інтерфейси. Делегати. Базові принципи об'єктно-орієнтованого програмування: інкапсуляція, успадкування, поліморфізм. Використання класів для реалізації файлового введення/виведення. Створення, відкриття та закриття файлу. Коректний доступ до файлів. Визначення режимів роботи з файлами.

Змістовий модуль 4. Створення базових елементів графічного інтерфейсу користувача.

Загальна характеристика бібліотеки класів Windows Forms.. Загальна характеристика Windows та основи програмування для ОС Windows. Інтерфейс прикладного програмування Win32API. Динамічні бібліотеки Win32API. Структура Windows-програми. Проектування та розробка основних компонент графічного інтерфейсу користувача (GUI). Обробка повідомлень. Засоби обробки подій прикладної програми. Стандартні ресурси прикладної програми: меню, шаблони діалогових блоків, бітові масиви, курсори, піктограми, акселератори, шаблони панелей інструментів. Класи стандартних елементів управління. Класи графічного інтерфейсу. Графічні об'єкти. Графічні методи. Типи меню. Клас для роботи з меню.

Змістовий модуль 5. Розробка додатків в інтегрованих середовищах. Побудова керованого та захищеного коду за допомогою бібліотеки класів Windows Forms, ADO.NET, WPF, ASP.NET

Можливості системи безпеки Code Access Security (CAS). Типи проектів. Бібліотека BCL для написання безпечних програм, орієнтованих на загальнономовне середовище виконання Common Language Runtime (CLR). Бібліотека класів System.Security.Cryptography. Засоби збереження даних. Стандарти доступу до баз даних. ODBC, OLE DB постачальники. Класи для роботи з базами даних бібліотеки ADO.NET (ActiveX Data Objects .NET). Способи роботи з файлами. Класи побудови графічних додатків з багатим інтерфейсом бібліотеки WPF (Windows Presentation Foundation). Засоби для створення веб-додатків та веб-сервісів - ASP.NET.

6. Контроль навчальних досягнень

Навчальні досягнення студентів з дисципліни оцінюються за модульно-рейтинговою системою, в основу якої покладено принцип поопераційної звітності, обов'язковості модульного контролю, накопичувальної системи оцінювання рівня знань, умінь та навичок, розширення кількості підсумкових балів до 100.

Оцінка за кожний змістовий модуль включає бали за поточну роботу студента на практичних та лабораторних заняттях, за виконання індивідуальних завдань, за модульну контрольну роботу. Виконання модульних контрольних робіт здійснюється в електронному вигляді. Модульний контроль знань студентів здійснюється після завершення вивчення навчального матеріалу змістового модуля.

У процесі оцінювання навчальних досягнень студентів застосовуються такі методи:

- *Методи усного контролю*: індивідуальне опитування, фронтальне опитування, співбесіда, екзамен.
- *Комп'ютерного контролю*: тестові програми, програмний проект.
- *Методи самоконтролю*: уміння самостійно оцінювати свої знання, самоаналіз.

Кількість балів за роботу з теоретичним матеріалом, на практичних заняттях, під час виконання самостійної роботи залежить від дотримання таких вимог:

- систематичність відвідування занять;
- своєчасність виконання навчальних і індивідуальних завдань;
- повний обсяг їх виконання;
- якість виконання навчальних і індивідуальних завдань;
- самостійність виконання;
- творчий підхід у виконанні завдань;
- ініціативність у навчальній діяльності;
- виконання тестових завдань.

Контроль успішності студентів з урахуванням поточного і підсумкового оцінювання здійснюється відповідно до навчально-методичної карти дисципліни, де зазначено види контролю і кількість балів за видами. Систему рейтингових балів для різних видів контролю та порядок їх переведення у національну (4-бальну) та європейську (ECTS) шкалу подано нижче у таблицях.

6.1. Розрахунок рейтингових балів за видами поточного (модульного) контролю

| № з/п | Вид діяльності студента | Макс. кількість балів за одиницю | Модуль 1 | | Модуль 2 | | Модуль 3 | | Модуль 4 | | Модуль 5 | |
|---------------------------------------|---|----------------------------------|---------------------------------|------------------------------|---------------------------------|------------------------------|---------------------------------|------------------------------|---------------------------------|------------------------------|---------------------------------|------------------------------|
| | | | Кількість одиниць до розрахунку | Макс. кількість балів за вид | Кількість одиниць до розрахунку | Макс. кількість балів за вид | Кількість одиниць до розрахунку | Макс. кількість балів за вид | Кількість одиниць до розрахунку | Макс. кількість балів за вид | Кількість одиниць до розрахунку | Макс. кількість балів за вид |
| 1 | Відвідування лекцій | 1 | 4 | 4 | 2 | 2 | 2 | 2 | 2 | 2 | 1 | 1 |
| 2 | Відвідування практичних та лабораторних занять | 1 | 6 | 6 | 6 | 6 | 6 | 6 | 8 | 8 | 5 | 5 |
| 3 | Виконання завдань для самостійної роботи | 5 | 1 | 5 | 1 | 5 | 1 | 5 | 1 | 5 | 1 | 5 |
| 4 | Робота на практичних (семінарських) заняттях | 10 | 3 | 30 | 3 | 30 | 3 | 30 | 2 | 20 | 1 | 10 |
| 5 | Лабораторне заняття (допуск, виконання, захист) | 10 | 3 | 30 | 3 | 30 | 3 | 30 | 6 | 60 | 4 | 40 |
| 6 | Виконання модульної контрольної роботи | 25 | 1 | 25 | 1 | 25 | 1 | 25 | 1 | 25 | 1 | 25 |
| | Макс. кількість балів за видами поточного контролю (МВ) | - | - | 100 | - | 98 | - | 98 | - | 120 | - | 86 |
| Максимальна кількість балів: 502 | | | | | | | | | | | | |
| Розрахунок коефіцієнта: $502/60=8,37$ | | | | | | | | | | | | |

6.2. Завдання для самостійної роботи та критерії її оцінювання.

Самостійна робота є видом поза аудиторної індивідуальної діяльності студента, результати якої використовуються у процесі вивчення програмового матеріалу навчальної дисципліни та містить результати дослідницького пошуку, відображає певний рівень його навчальної компетентності.

Перелік тем та оцінювання самостійної роботи студента

| № з/п | Назва теми | Кількість годин | Бали |
|---|--|-----------------|----------|
| Змістовий модуль 1. Технології безпечного програмування | | 8 | 5 |
| 1. | Класифікація та види вразливостей програмного забезпечення. Статичні та динамічні засоби аналізу програмного коду на наявність вразливостей часу проектування та реалізації. Протоколи та стандарти захисту додатків. | 8 | 5 |
| Змістовий модуль 2. Технологія структурного процедурного програмування | | 4 | 5 |
| 2. | Обробка виняткових ситуацій. Безпека виняткових ситуацій. Безпечні способи роботи з пам'яттю. Цілочисельна безпека: цілочисельне переповнення, помилки цілочисельного перетворення та відсікання, логічні помилки. Безпечний доступ до структурованих даних. | 4 | 5 |
| Змістовий модуль 3. Технологія об'єктно-орієнтованого програмування | | 4 | 5 |
| 7. | Головні принципи об'єктного підходу в програмуванні. Інтерфейси. Делегати. | 4 | 5 |

| | | | |
|---|--|-----------|-----------|
| Змістовий модуль 4. Створення базових елементів графічного інтерфейсу користувача | | 4 | 5 |
| 9. | Система безпеки Code Access Security технології .NET. | 4 | 5 |
| Змістовий модуль 5. Розробка додатків в інтегрованих середовищах. Побудова керованого та захищеного коду за допомогою Windows Forms, ADO.NET, WPF, ASP.NET | | 6 | 5 |
| 11 | Огляд стандартів процесу створення ПЗ. Якість ПЗ. Характеристики якості ПЗ. Моделі якості процесу розробки. | 6 | 5 |
| Разом | | 26 | 25 |

Критерії оцінювання самостійної роботи студента

| № п/п | Критерії оцінювання роботи | Максимальна кількість балів за кожним критерієм |
|--------------|---|---|
| 1 | Повний обсяг їх виконання. Якість виконання, відповідність змісту завдання. | 2 бали |
| 2 | Самостійність виконання. Творчий підхід у виконанні завдань. Доказовість висновків, обґрунтованість власної позиції, пропозиції щодо розв'язання проблеми, визначення перспектив дослідження. | 2 бали |
| 3 | Дотримання вимог щодо технічного оформлення. | 1 бал |
| Разом | | 5 балів |

Індивідуальні завдання

| Вид індивідуальних завдань | Тематика індивідуальних завдань | Бали |
|----------------------------|--|------|
| Курсова робота | <p>Проектування та розробка об'єктно-орієнтованої прикладної програми з графічним інтерфейсом користувача (область застосування може пропонуватися студентом) згідно до методичних порад з виконання курсової роботи пропонуються такі напрямки створення прикладних програм:</p> <ol style="list-style-type: none"> 1. Проектування та реалізація прикладної програми з графічним інтерфейсом «Шифрування тексту». 2. Проектування та реалізація прикладної програми з графічним інтерфейсом «Інтернет послуги». 3. Проектування та реалізація прикладної програми з графічним інтерфейсом «Успішність студентів». 4. Проектування та реалізація прикладної програми з графічним інтерфейсом «Операції з многочленами». 5. Проектування та реалізація прикладної програми з графічним інтерфейсом «Довідник з перетворення одиниць виміру». 6. Проектування та реалізація прикладної програми «Операції з матрицями». 7. Проектування та реалізація прикладної програми презентаційної графіки для побудови діаграм. 8. Проектування та реалізація ігрової програми (назва гри). 9. Проектування та реалізація прикладної програми зміни системних установок. 10. Проектування та реалізація довідкової системи з мови програмування у прикладній програмі. | 100 |

| | | |
|--|--|--|
| | 11. Проектування та реалізація навчальної програми створення та використання загальних елементів управління. 12. Проектування та реалізація прикладної програми з використанням бази даних (назва бази даних). 13. Проектування та реалізація веб-додатка (назва). | |
|--|--|--|

Розрахунок рейтингових балів за виконання курсової роботи

| Форми контролю та критерії оцінювання | Кількість балів |
|---|-----------------|
| Курсова робота виконана вчасно згідно із завданням. | 20 |
| Робота структурована, матеріал роботи викладено логічно. | 10 |
| Якість оформлення протоколу виконання курсової роботи. (відсутність стилістичних та граматичних помилок). | 10 |
| Програмний проект має дружній та безпечний інтерфейс. | 10 |
| Данні програми зберігаються на зовнішніх носіях. | 10 |
| Наявність засобів розмежування доступу користувачів програми. | 10 |
| Презентація роботи. Доповідь логічна, стисла, структурована, проголошена вільно. | 10 |
| Відповіді на запитання (повні та обґрунтовані). | 20 |
| Разом | 100 |

6.3. Форми проведення модульного контролю та критерії оцінювання.

Модульний контроль здійснюється відповідно до навчально-методичної карти дисципліни та перевіряє рівень досягнення результатів навчання студентів. Форма проведення комбінована. Модульна контрольна робота оцінюється у 25 балів за розподілом: – 15 балів комп'ютерний тест, що складається з 15 запитань закритої та відкритої форм та 10 балів – виконання практико-орієнтованого завдання за темами, визначеними в тематичному плані.

Засоби підсумкового контролю – комп'ютерний тест для тематичного (модульного) контролю, комплект друкованих завдань для практичного завдання.

Розрахунок рейтингових балів за семестровий контроль (екзамен)

| Форма проведення | Кількість завдань | Кількість балів за одиницю | Максимальна підсумкова кількість балів |
|------------------------------|-------------------|----------------------------|--|
| Тематичний тест | 15 | 1 | 15 балів |
| Тематичне практичне завдання | 1 | 10 | 10 балів |
| Разом | | | 25 балів |

Критерії оцінювання виконання тематичного практичного завдання

| № п/п | Критерії оцінювання роботи | Максимальна кількість балів за кожним критерієм |
|-------|--|---|
| 1 | Програмний проект виконано вчасно і в повному обсязі | 4 бали |

| | | |
|-------|--|----------|
| 2 | Відповідність змісту завдання, структурований код, дружній та безпечний інтерфейс. | 3 бали |
| 3 | Повні та обґрунтовані відповіді на запитання. Самостійність виконання | 3 бали |
| Разом | | 10 балів |

6.4. **Форми проведення семестрового контролю та критерії оцінювання.**

Семестрове (підсумкове) оцінювання здійснюється у формі екзамену, умовою допуску до якого є отриманням студентом 35 балів (з врахуванням коефіцієнту) за результатами поточного контролю.

Форма проведення екзамену – комбінована. Екзамен оцінюється у 40 балів за розподілом: 20 балів – комплексний тест з дисципліни; 20 балів – виконання практико-орієнтованого завдання (програмний проект). Засоби підсумкового контролю – тест (комп'ютерний) для підсумкового контролю, комплект практичних завдань.

Виконання практичного завдання передбачає перевірку рівня оволодіння студентом теоретичними знаннями та практичними вміннями з проектування та реалізації прикладних програм.

Оцінювання практичного завдання відбувається в межах від 0 до 20 балів, згідно критеріїв оцінювання, й здійснюється з урахуванням: рівнів сформованості аналітико-синтетичних, творчих та методичних умінь необхідних для написання безпечного коду.

Бали за виконання тесту та бали за виконання практичного завдання додаються. Оцінювання результатів засвоєння теоретичних знань та оцінювання сформованості практичних навичок володіння технологіями безпечного програмування, продемонстровані студентами на екзамені, представлене у таблицях.

Розрахунок рейтингових балів за семестровий контроль (екзамен)

| Форма проведення | Кількість завдань | Кількість балів за одиницю | Максимальна підсумкова кількість балів |
|--------------------|-------------------|----------------------------|--|
| Комплексний тест | 20 | 1 | 20 балів |
| Практичне завдання | 1 | 20 | 20 балів |
| Разом | | | 40 балів |

Критерії оцінювання виконання практичного завдання

| № п/п | Критерії оцінювання роботи | Максимальна кількість балів за кожним критерієм |
|-------|---|---|
| 1 | Повний обсяг виконання | 5 бали |
| 2 | Якість виконання, відповідність змісту завданню. | 5 бали |
| | Творчий підхід у виконанні завдання. Структурований код, дружній та безпечний інтерфейс, наявність структур збереження даних, засобів розмежування доступу. | 5 бали |
| 3 | Повні та обґрунтовані відповіді на запитання. Самостійність виконання | 5 бали |
| Разом | | 20 балів |

6.5. **Орієнтовний перелік питань для семестрового контролю.**

1. Основні поняття безпечного програмування.
2. Стратегії безпеки, визначення вразливостей та шляхів їх подолання.
3. Життєвий цикл розробки безпечного програмного забезпечення (ПЗ).
4. Методології та технології програмування.

5. Створення керованого та захищеного об'єктно-орієнтованого коду з використанням технології .NET.
6. Базові (стандартні) типи даних.
7. Цілочисельна безпека: цілочисельне переповнення, помилки цілочисельного перетворення та відсікання, логічні помилки.
8. Перетворення типів даних.
9. Статичний та динамічний контроль типів.
10. Ідентифікатор. Змінні. Константи.
11. Вирази.
12. Основні операції. Пріоритет операцій.
13. Структурне програмування.
14. Типи структур управління: послідовна структура, структура вибору, структура повторення.
15. Обробка виняткових ситуацій.
16. Безпека виняткових ситуацій.
17. Одновимірні масиви та символічні рядки.
18. Стратегії захисту: перевірка розміру даних та виходу за межі, захист від пошкодження стеку та отримання управління стеком, захист від вбудовування коду, контроль за недійсними посиланнями на рядки.
19. Алгоритми сортування, пошуку та шифрування елементів масиву.
20. Багатовимірні масиви.
21. Стратегії захисту: коректна обробка збоїв виділення пам'яті, неможливість многократного звільнення пам'яті, витоку пам'яті, перевірка значень, які повертаються.
22. Динамічне виділення пам'яті: функції розподілення, звільнення пам'яті, збирання сміття.
23. Функції визначені програмістом. Область дії функції.
24. Типи функцій.
25. Формальні та фактичні параметри функції.
26. Стратегії захисту від вразливостей (переповнення буферу, перегляд стеку та пам'яті, перезапис пам'яті)
27. Реалізація файлового введення/виведення.
28. Коректний доступ до файлів.
29. Визначення режимів відкриття файлу.
30. Загальна характеристика структур збереження даних: лінійні та ієрархічні структури даних.
31. Лінійні структури даних: список, стек, черга.
32. Фундаментальні алгоритми для роботи з даними: вставка, вилучення елементів, сортування, пошук, шифрування.
33. Базові концепції класів та об'єктів.
34. Визначення класу та методів класу.
35. Модифікатори доступу до елементів класу.
36. Конструктори та ініціалізація.
37. Перевантаження операторів та методів класу.
38. Статичні члени класу.
39. Типи відношень між класами.
40. Просте успадкування класів. Правила створення похідних класів.
41. Віртуальні методи. Заміщені методи. Приховані (нові) методи.
42. Абстрактні класи. Закриті класи.
43. Базові принципи об'єктно-орієнтованого програмування: інкапсуляція, успадкування, поліморфізм.
44. Загальна характеристика інтегрованого середовища розробки програм IDE Visual C#.
45. Можливості системи безпеки CAS.
46. Бібліотека Base Class Library (BCL) для написання безпечних програм, орієнтованих на загальномовне середовище виконання Common Language Runtime (CLR).

47. Інтерфейс прикладного програмування Win32API. Динамічні бібліотеки Win32API.
48. Характеристика класів бібліотеки BCL (Base Class Library).
49. Проектування та розробка основних компонент графічного інтерфейсу користувача (GUI).
50. Засоби обробки подій прикладної програми.
51. Стандартні елементи прикладної програми: меню, шаблони діалогових блоків, бітові масиви, курсори, піктограми, шаблони панелей інструментів.
52. Графічний інтерфейс пристроїв (GDI).
53. Класи форми та стандартних елементів управління.
54. Класи графічного інтерфейсу.
55. Графічні об'єкти. Графічні функції.
56. Типи меню.
57. Класи для роботи з меню, панеллю швидкого вибору елементів, рядком стану.
58. Способи роботи з файлами.
59. Стандарти доступу до баз даних.
60. ODBC-драйвери.
61. Класи для роботи з базами даних.
62. Моделі процесу створення ПЗ.
63. Особливості SDL.
64. Специфікація ПЗ (вимоги безпеки).
65. Проектування ПЗ (моделювання загроз, аналіз атак, перевірка вхідних даних, границі довіри).
66. Реалізація ПЗ (глибинний захист коду).
67. Тестування та налагодження ПЗ (аудит коду, огляд області атак).
68. Експлуатація та супроводження ПЗ. CASE-засоби.

6.6. Шкала відповідності оцінок

| Рейтингова оцінка | Сума балів за всі види навчальної діяльності | Значення оцінки |
|-------------------|--|--|
| A | 90-100 | Відмінно — відмінний рівень знань (умінь) в межах обов'язкового матеріалу з, можливими, незначними недоліками |
| B | 82-89 | Дуже добре - достатньо високий рівень знань (умінь) в межах обов'язкового матеріалу без суттєвих (грубих) помилок |
| C | 75-81 | Добре - в цілому добрий рівень знань (умінь) з незначною кількістю помилок |
| D | 69-74 | Задовільно - посередній рівень знань (умінь) із значною кількістю недоліків, достатній для подальшого навчання або професійної діяльності |
| E | 60-68 | Достатньо - мінімально можливий допустимий рівень знань (умінь) |
| FX | 35-59 | Незадовільно з можливістю повторного складання - незадовільний рівень знань, з можливістю повторного перескладання за умови належного самостійного доопрацювання |
| F | 1-34 | Незадовільно з обов'язковим повторним вивченням курсу - досить низький рівень знань (умінь), що вимагає повторного вивчення дисципліни |

7. Навчально-методична картка дисципліни

Разом: 180 год., лекції – 22 год., практичні роботи – 24 год., лабораторні роботи – 38 год., модульний контроль – 10 год., самостійна робота – 26 год., семестровий контроль – 30 год.

| Модулі (назви, бали) | Змістовий модуль 1. Технології безпечного програмування (100 балів) | | | | Змістовий модуль 2. Технологія структурного процедурного програмування (98 балів) | | | Змістовий модуль 3. Технологія об'єктно-орієнтованого програмування (98 балів) | | | Змістовий модуль 4. Створення базових елементів графічного інтерфейсу користувача (120 балів) | | | Змістовий модуль 5. Розробка додатків в інтегрованих середовищах. Побудова керованого та захищеного коду за допомогою Windows Forms, ADO.NET, WPF, ASP.NET (86 балів) | | |
|----------------------------------|---|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|
| Лекції (теми, бали) | Основні принципи безпечного програмування (1 бал) Характеристика технології .Net Framework (1 бал) Базові поняття C#. Данні програми. Безпека даних (1 бал) Управління потоком виконання програми (1 бал) | | | | Структурний підхід в програмуванні. (1 бал) Розбиття програми на підпрограми. Методи (1 бал) | | | Принципи об'єктно-орієнтованого програмування. Класи та об'єкти (1 бал) Похідні класи (1 бал) | | | Розробка основних компонент графічного інтерфейсу користувача (1 бал) Класи стандартних елементів управління бібліотеки класів Windows Forms (1 бал) | | | Засоби збереження даних. DataGridView, DB (1 бал) | | |
| Практичні роботи (теми, бали) | Знайомство з IDE Visual C#. Робота з консольним проектом (11 балів) Розробка та налагодження консольних програм з розгалуженням (11 балів) Розробка та налагодження консольних програм повторенням (11 балів) | | | | Реалізація одновимірних та двовимірних масивів (11 балів) Розбиття коду на підпрограми. Розробка методів. (11 балів) Файлове введення/виведення даних (11 балів) | | | Визначення класів та створення об'єктів даного класу (11 балів) Визначення набору функцій для роботи з об'єктами класу (11 балів) Створення похідних класів (11 балів) | | | Реалізація програм з графічним інтерфейсом з використанням бібліотеки Windows Forms (11 балів) Використання стандартних елементів управління. Реалізація різних типів меню програми (11 балів) | | | Реалізація проектів для роботи з базами даних (11 балів) | | |
| Лабораторні заняття (теми, бали) | Проектування, реалізація та налагодження консольних прикладних програм лінійної структури (11 балів) Проектування, реалізація та налагодження консольних прикладних програм з розгалуженням (11 балів) Розробка та налагодження консольних програм повторенням (11 балів) Проектування, реалізація та налагодження ПП з використанням одновимірних масивів (11 балів) Проектування, реалізація та налагодження ПП для виконання операцій з матрицями (11 балів) Проектування, реалізація та налагодження ПП з використанням функцій (11 балів) | | | | Проектування, реалізація та налагодження ПП з використанням одновимірних масивів (11 балів) Проектування, реалізація та налагодження ПП для виконання операцій з матрицями (11 балів) Проектування, реалізація та налагодження ПП з використанням функцій (11 балів) | | | Проектування, реалізація та налагодження об'єктно-орієнтованих ПП з використанням класів (11 балів) Проектування, реалізація та налагодження об'єктно-орієнтованих ПП з використанням похідних класів (11 балів) Проектування, реалізація та налагодження ПП з використанням файлового введення/виведення (11 балів) | | | Проектування, реалізація та налагодження об'єктно-орієнтованих прикладних програм на базі форми (22 бали) Обмін даними між діалоговими вікнами, збереження даних (22 бали) Створення меню різних типів, рядка стану та панелі швидкого вибору пунктів меню (11 балів) Використання графічних об'єктів та функцій (11 балів) | | | Проектування, реалізація та налагодження об'єктно-орієнтованих ПП з використанням таймерів (11 балів) Проектування, реалізація та налагодження об'єктно-орієнтованих ПП з використанням елементу DataGridView (11 балів) Проектування, реалізація та налагодження ПП з використанням баз даних (11 балів) Проектування, реалізація та налагодження веб-додатку (11 балів) | | |
| Самостійна робота | Самостійна робота 1 (5 балів) | | | | Самостійна робота 2 (5 балів) | | | Самостійна робота 3 (5 балів) | | | Самостійна робота 4 (5 балів) | | | Самостійна робота 5 (5 балів) | | |
| Модульний контроль | Модульна контрольна робота 1 (25 балів) | | | | Модульна контрольна робота 2 (25 балів) | | | Модульна контрольна робота 3 (25 балів) | | | Модульна контрольна робота 4 (25 балів) | | | Модульна контрольна робота 5 (25 балів) | | |
| Підсумковий контроль (вид, бали) | Екзамен (40 балів) Курсова робота (100 балів) | | | | | | | | | | | | | | | |

8. Рекомендовані джерела

Основна (базова):

1. Коноваленко І.В. Програмування мовою С# 6.0. Навчальний посібник для технічних спеціальностей вищих навчальних закладів. - Тернопіль.: ТНТУ, 2016. – 229 с.
2. Хортон А. Visual Studio 2010 : базовий курс.: Пер. с англ. — М.: Издательский дом "Вильямс", 2011. — 1152 с.
3. Ховард М., Лебланк Д., Виєга Д. Как написать безопасный код на С++, Java, Perl, PHP, ASP.NET. М.: Издательство ДМК Пресс, 2014. - 288 с.
4. Голуб Б.М. С#. Концепція та синтаксис. Навчальний посібник. - Львів.: ЛНУ, 2006. – 136 с.
5. Дейтел П, Дейтел Х. Как программировать на Visual С# 2012. 5-е изд. – СПб.: Питер, 2014. - 864 с.

Додаткова:

1. Сикорд Р. Безопасное программирование на С и С++» (для С11/С++11), 2-е издание. – М.: Вильямс, 2015. — 496 с.
2. Ховард М., Лебланк Д., Виєга Д. 19 смертных грехов, угрожающих безопасности программ. Как не допустить типичных ошибок. — М.: Издательство ДМК Пресс, 2006. — 288с.
3. Ховард М., Лебланк Д. Защищенный код. /Пер. с англ. — 2-е изд., испр. — М.: Издательство «Русская Редакция», 2005. — 704 с.
4. Эспозито Д., Салтарелло А. Microsoft.NET: архитектура корпоративных приложений, — М.: Издательский дом "Вильямс", 2-е издание, 2016. - 432 с.
5. Камаев В.А., Костерин В.В. Технологии программирования: Учебник – М.: Высш. шк., 2006. — 454 с.

6. Додаткові ресурси

1. Visual Studio 2017 (рос.) [Електронний ресурс]. – Режим доступу: <https://docs.microsoft.com/en-us/visualstudio/?view=vs-2017>
2. ВікіПідручник C Sharp 2017 (укр.) [Електронний ресурс]. – Режим доступу: https://uk.wikibooks.org/wiki/C_Sharp
3. С# : Використання баз даних (укр.) [Електронний ресурс]. – Режим доступу: <http://programmersworld.xyz/article/4/92>
4. Создание приложения Windows Forms (рос.) [Електронний ресурс]. – Режим доступу: <http://olocoder.ru/VS0.html>
5. Руководство по ASP.NET Core 2.0(рос.) [Електронний ресурс]. – Режим доступу: <https://metanit.com/sharp/aspnet5/>