

Міністерство освіти і науки України
Харківський національний університет імені В.Н. Каразіна
Кафедра моделювання систем і технологій

“ЗАТВЕРДЖУЮ”

Проректор з науково-педагогічної
роботи



Антон ПАНТЕЛЕЙМОНОВ

“ 30 ” 2021 р.

РОБОЧА ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ
«Мови прикладного програмування»

рівень вищої освіти _____ перший (бакалаврський) рівень _____
галузь знань _____ 12 інформаційні технології _____
спеціальність _____ 122 Комп'ютерні науки _____
освітня програма _____ Комп'ютерні науки _____
спеціалізація _____
вид дисципліни _____ за вибором _____
факультет _____ комп'ютерних наук _____

2021/ 2022 навчальний рік

Програму рекомендовано до затвердження Вченою радою факультету комп'ютерних наук

«30» червня 2021 року, протокол №15

РОЗРОБНИК ПРОГРАМИ: старший викладач кафедри моделювання систем і технологій
Рудичев Дмитро Володимирович.

Програму схвалено на засіданні кафедри моделювання систем і технологій

Протокол від «04» червня 2021 року №12

Завідувач кафедри моделювання систем і технологій


Микола ТКАЧУК

Програму погоджено з гарантом освітньо-професійної програми 122 «Комп'ютерні науки»

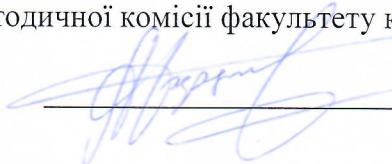
Гарант освітньо-професійної програми 122 «Комп'ютерні науки»


Микола СТЕРВОЄДОВ

Програму погоджено науково-методичною комісією факультету комп'ютерних наук

Протокол від «25» червня 2021 року №9

Голова науково-методичної комісії факультету комп'ютерних наук


Анатолій БЕРДНІКОВ

ВСТУП

Програма навчальної дисципліни «Мови прикладного програмування» складена відповідно до освітньо-професійної програми підготовки першого (бакалаврського) рівня спеціальності 122 Комп'ютерні науки.

1. Опис навчальної дисципліни

1.1. Мета викладання навчальної дисципліни

Метою викладання навчальної дисципліни «Мови прикладного програмування» є підготовка кваліфікованих фахівців у галузі інформаційних технологій.

1.2. Основні завдання вивчення дисципліни.

Основними завданнями вивчення дисципліни «Мови прикладного програмування» є забезпечити відповідні сучасним вимогам знання студентів як у теоретичних засновках Visual Studio .Net, Framework .Net так і забезпечити практичні знання та уміння по використанню C# , .Net, Framework і основним принципам їхнього застосування на сторінках ASP і сайтах World Wide Web.

В ході вивчення дисципліни у студента повинні формуватися наступні компетентності.

Загальні компетентності (ЗК):

- здатність до пошуку, оброблення та аналізу інформації з різних джерел (ЗК7);
- здатність генерувати нові ідеї (креативність) (ЗК8);
- здатність приймати обґрунтовані рішення (ЗК9);
- здатність бути критичним і самокритичним (ЗК11);
- здатність оцінювати та забезпечувати якість виконуваних робіт (ЗК12);
- здатність зберігати та примножувати моральні, культурні, наукові цінності і досягнення суспільства на основі розуміння історії та закономірностей розвитку предметної області, її місця у загальній системі знань про природу і суспільство та у розвитку суспільства, техніки і технологій, використовувати різні види та форми рухової активності для активного відпочинку та ведення здорового способу життя (ЗК15).

Спеціальні (фахові, предметні) компетентності (ФК):

- здатність до виявлення статистичних закономірностей недетермінованих явищ, застосування методів обчислювального інтелекту, зокрема статистичної, нейромережевої та нечіткої обробки даних, методів машинного навчання та генетичного програмування тощо (ФК2);
- здатність здійснювати формалізований опис задач дослідження операцій в організаційно-технічних і соціально-економічних системах різного призначення, визначати їх оптимальні розв'язки, будувати моделі оптимального управління з урахуванням змін економічної ситуації, оптимізувати процеси управління в системах різного призначення та рівня ієрархії (ФК5);
- здатність до розробки мережевого програмного забезпечення, що функціонує на основі різних топологій структурованих кабельних систем, використовує комп'ютерні системи і мережі передачі даних та аналізує якість роботи комп'ютерних мереж (ФК13);

1.3. Кількість кредитів - 5

1.4. Загальна кількість годин – 150

1.5. Характеристика навчальної дисципліни	
Обов'язкова / <u>за вибором</u>	
Денна форма навчання	Денна форма навчання
Рік підготовки	
3-й	3-й
Семестр	
5-й	6-й
Лекції	
год.	48 год.
Практичні, семінарські заняття	
год.	год.
Лабораторні заняття	
год.	48 год.
Самостійна робота	
год.	14 год.
у тому числі індивідуальні завдання	
40 год.	

1.6. Заплановані результати навчання

Згідно з вимогами освітньо-професійної програми студенти повинні досягти таких результатів навчання:

знати:

- теоретичні засновки C#, .Net, Framework;
- основи побудови WinForms та ASP сценаріїв та прийоми, які використовуються;
- принципи проектування сайтів; знати особливості розробки ASP.

вміти:

- проектувати WinForms , з використанням C#
- складати програми та знаходити помилки в C#-кодi;
- використовувати мову ASP для розробки скриптів;
- використовувати мову ASP для розробки скриптів щодо ADO бази даних.

В Результаті вивчення дисципліни у студента повинні формуватися наступні програмні результати навчання (ПРН):

- застосовувати знання основних форм і законів абстрактно-логічного мислення, основ методології наукового пізнання, форм і методів вилучення, аналізу, обробки та синтезу інформації в предметній області комп'ютерних наук (ПРН1);

- використовувати знання закономірностей випадкових явищ, їх властивостей та операцій над ними, моделей випадкових процесів та сучасних програмних середовищ для розв'язування задач статистичної обробки даних і побудови прогнозних моделей (ПРН3);
- проектувати, розробляти та аналізувати алгоритми розв'язання обчислювальних та логічних задач, оцінювати ефективність та складність алгоритмів на основі застосування формальних моделей алгоритмів та обчислюваних функцій (ПРН5);
- використовувати методи чисельного диференціювання та інтегрування функцій, розв'язання звичайних диференціальних та інтегральних рівнянь, особливостей чисельних методів та можливостей їх адаптації до інженерних задач, мати навички програмної реалізації чисельних методів (ПРН6);
- розуміти принципи моделювання організаційно-технічних систем і операцій; використовувати методи дослідження операцій, розв'язання одно- та багатокритеріальних оптимізаційних задач лінійного, цілочисельного, нелінійного, стохастичного програмування (ПРН7);
- володіти мовами системного програмування та методами розробки програм, що взаємодіють з компонентами комп'ютерних систем, знати мережні технології, архітектури комп'ютерних мереж, мати практичні навички технології адміністрування комп'ютерних мереж та їх програмного забезпечення (ПРН14);
- розуміти концепцію інформаційної безпеки, принципи безпечного проектування програмного забезпечення, забезпечувати безпеку комп'ютерних мереж в умовах неповноти та невизначеності вихідних даних (ПРН16);
- виконувати паралельні та розподілені обчислення, застосовувати чисельні методи та алгоритми для паралельних структур, мови паралельного програмування при розробці та експлуатації паралельного та розподіленого програмного забезпечення (ПРН17).

2. Тематичний план навчальної дисципліни

Розділ 1

Тема 1. Visual Studio. Net, Framework .Net .

Бренд .Net. Visual Studio .Net - відкрите середовище розробки. Каркас Framework. Net. Бібліотека класів FCL - статичний компонент каркаса. Загальномовне виконавче середовище CLR - динамічний компонент каркаса. Керований код. Загальномовного специфікації CLS і сумісні модулі.

Тема 1.1 Мова C # і перші проекти.

Створення мови. Її особливості. Рішення, проекти, простору імен. Консольні і Windows-додатки C #, побудовані за замовчуванням.

Тема 2. Система типів мови C #.

Загальний погляд. Система типів. Типи-значення і посилальні типи. Вбудовані типи. Порівняння з типами C ++. Типи або класи? І типи, і класи! Перетворення змінних в об'єкти і vice versa. Операції "упакувати" і "розпакувати". Перетворення типів. Перетворення всередині арифметичного типу. Перетворення строкового типу. Клас Convert і його методи. Перевіряються перетворення. Управління перевіркою арифметичних перетворень.

Тема 2.1 Перетворення типів.

Перетворення типів. Перетворення всередині арифметичного типу. Перетворення строкового типу. Клас Convert і його методи. Перевіряються перетворення. Управління перевіркою арифметичних перетворень.

Тема 2.2 Змінні і вираження.

Оголошення змінних. Синтаксис оголошення. Ініціалізація. Час життя і область видимості. Де оголошуються змінні? Локальні і глобальні змінні. Чи є глобальні змінні в C #? Константи.

Тема 2.3 Вирази. Операції у вираженнях.

Побудова виразів. Операції і їх пріоритети. Опис операцій.

Тема 2.4 Присвоєння і вбудовані функції.

Присвоєння. Новинка C # - певне присвоювання. Класи Math, Random і вбудовані функції.

Тема 2.5 Оператори мови C #.

Оператори мови C #. Оператор присвоювання. Складовою оператор. Порожній оператор. Оператори вибору. If-оператор. Switch-оператор. Оператори переходу. Оператор goto. Оператори break, continue. Оператори циклу. For-оператор. Цикли while. Цикл foreach.

Тема 3. Процедури і функції - методи класу.

Процедури і функції - дві форми функціонального модуля. Чим відрізняються ці форми? Процедури і функції - це методи класу. Опис методів (процедур і функцій). Синтаксис. Атрибути доступу. Статичні і динамічні методи. Формальні аргументи. Статус аргументів. Тіло методів. Виклик процедур і функцій. Фактичні аргументи. Семантика виклику. Поля класу або аргументи методу? Поля класу або функції без аргументів? Проектування класу Account. Функції з побічним ефектом. Перевантаження методів.

Тема 3.1. Масиви мови C #.

Загальний погляд на масиви. Порівняння з масивами C ++. Чому масиви C # краще, ніж масиви C ++. Види масивів - одномірні, багатовимірні і порізані. Динамічні масиви.

Тема 3.2. Клас Array і нові можливості масивів.

Сімейство класів-масивів. Батьківський клас Array і успадковані їм інтерфейси. Нові можливості масивів в C #. Як коректно працювати з масивами об'єктів?

Тема 3.3 Символи і рядки постійної довжини в C # Загальний погляд.

Рядки в C ++. Рядки в C #. Рядки постійної і змінної довжини. Класи char, char [].

Тема 3.4. Рядки C #. Класи String і StringBuilder.

Рядки C #. Клас String. Змінювані і незмінні рядкові класи. Класи Net Framework, що розширюють строковий тип. Клас StringBuilder.

Тема 3.5. Регулярні вираження.

Регулярні вираження. Простір RegularExpressions і його класи. Регулярні вираження і мови. Теорія регулярних виразів. Практика застосування регулярних виразів. Розбір текстів і пошук за зразком. Властивості і методи класу Regex і інших класів, пов'язаних з регулярними вираженнями. Приклади застосування регулярних виразів.

Тема 4. Класи. Дві ролі класу в ООП.

Синтаксис опису класу. Поля і методи класу. Конструктори і деструктори. Статичні поля і методи. Статичні конструктори. Поля тільки для читання. Закриті поля. Стратегії доступу до полів класу. Процедури властивості. Індексатори. Приклади.

Тема 4.1 Структури і перерахування.

Поняття розгорнутого і посилального типу. Структури - реалізація розгорнутих класів. Синтаксис структур. Порівняння структур і класів. Вбудовані структури. Перерахування - окремий випадок класу. Особливості перерахувань. Приклади.

Тема 4.2. Відносини між класами.

Клієнти і спадкоємці Класи. Відносини між класами. Ставлення клієнти - постачальники. Ставлення успадкування. Одиначне успадкування. Батьки і спадкоємці. Предки і нащадки. Що успадковують нащадки. Що можуть змінити нащадки. Одностороннє присвоювання. Контроль типів і зв'язування - статична і динамічна. Поліморфізм. Проектування класів. Абстрактні класи. Класи поведінки.

Тема 4.3. Інтерфейси. Множинне спадкування.

Інтерфейси як окремий випадок класу. Множинне спадкування. Проблеми. Множинне спадкування інтерфейсів. Вбудовані інтерфейси. Інтерфейси IComparable, ICloneable, ISerializable. Поверхневе і глибоке клонування і серіалізація. Збереження і обмін даними.

Тема 5. Функціональний тип в C#. Делегати.

Нове слово для старого поняття. Функціональний тип. Функції вищих порядків. Обчислення інтеграла і сортування. Два способи взаємодії частин при побудові складних систем. Функції зворотного виклику. Спадкування і функціональні типи. Порівняння двох підходів. Клас Delegate. Методи і властивості класу. Операції над делегатами. Комбінування делегатів. Список викликів.

Тема 5.1. Події Класи з подіями.

Загальний погляд. Клас Sender і класи Receivers. Клас Sender. Як оголошуються події? Делегати і події. Класи з подіями, які допускаються .Net Framework. Клас EventArgs і його нащадки. Вхідні і вихідні аргументи події. Клас Receiver. Обробник події. Вбудовування об'єкта Sender. Зв'язування обробника з подією. Відключення обробника. Взаємодія об'єктів sender і receiver. Події - поля або процедури-властивості? Динамічне зв'язування подій з їх обробниками.

Тема 6. Універсальність. Класи з родовими параметрами.

Спадкування і універсальність - взаємно доповнюють базові механізми створення сімейства класів. Родові параметри універсального класу. Синтаксис універсального класу. Родове породження примірників універсального класу. Методи з родовими параметрами. Обмежена універсальність - обмеження, що накладаються на родові параметри. Види обмежень. Обмеження універсальності - це свобода дій. Приклади. Родові параметри і окремі випадки класів: структури, інтерфейси, делегати. Універсальність і Framework .Net.

Тема 6.1. Налаштування і обробка виняткових ситуацій.

Коректність і стійкість. Специфікація системи. Коректність і стійкість програмних систем. Виняткові ситуації. Обробка виняткових ситуацій. Життєвий цикл програмної системи. Три закони програмотехніки. Налаштування. Створення надійного коду. Мистецтво налаштування. Налаштування і інструментальне середовище Visual Studio .Net.

Тема 7. Введення в windows-форми.

Основи роботи з Visual Studio .NET. Форми. Solution Explorer. Class View. Properties Window. Toolbox. Режими дизайну і коду. Властивості проекту. Компіляція програми. Делегати. Многооб'єктних делегати. Події. Клас з подією. Події в Windows-додатках. Події миші. Форматування елементів управління. Додаток "калькулятор". Переміщення по коду, вікна Types і Members. XML-документування коду. Створення іконки для додатка.

Тема 8. Робота з елементами управління.

Створення головного меню. Створення MDI-додатків. Перерахування MdiLayout. Виризання, копіювання і вставка текстових фрагментів. Контекстне меню. Діалогові вікна. OpenFileDialog. SaveFileDialog. OpenFileDialog і SaveFileDialog для SDI-додатків. FontDialog. ColorDialog. StatusBar. CheckBox. Властивість TabIndex елементів управління. Label, LinkLabel і PictureBox. Режими запуску вікон. Модифікатори доступу та успадкування форм. ToolBar і ImageList. Читання і запис файлів. Читання і запис текстових файлів. Object Browser і IntelliSense. Перевірка існування файлу. Робота з файловою системою Windows. Читання і запис двійкових файлів. Запуск програм з програми.

Тема 9. Робота з елементами управління (продовження).

Операція Drag-and-Drop. Додавання елементів керування в режимі роботи програми. Елементи управління CheckBox, GroupBox, RadioButton, ComboBox. Перевірка вводяться значень. Події KeyPress і Validating елемента управління TextBox. Елемент управління ErrorProvider. Створення призначених для користувача (композитних) елементів управління. Елемент управління NumericUpDown. Запуск програми в області повідомлень. Елемент управління NotifyIcon. Збереження установок програми. XML-серіалізація. Поштова програма Ballet. Додавання проектів. Створення Мастера.

Тема 9.1 Організація інтерфейсу і малювання в формах.

Організація інтерфейсу. Шаблони форм. Заселення форми елементами управління. Класи елементів управління. Приклади класів. Клас ListBox. Спадкування форм. Організація меню, головне меню. Інструментальні панелі з кнопками. Малювання в формах. Класи малювання. Кисті і пір'я.

Розділ 2

Тема 1. Принципи роботи і структура Web-додатків на основі ASP.NET.

Розглядається архітектура сучасних Web-додатків, взаємодія клієнтської і серверної частин таких програм, принципи їх організації в середовищі ASP.NET

Тема 1.1. Основи роботи в Visual Studio .NET 2008

Розглядаються можливості інтегрованого середовища розробки Visual Studio 2008 для створення додатків ASP.NET і налаштування її інтерфейсу для потреб конкретного користувача. Розглядається процес створення нового Web-додатки, способи навігації по його структурі, перегляду і редагування інформаційної частини в режимі роботи з вихідним кодом і дизайну, використання різних елементів управління на сторінках, а також процедур - обробників подій цих елементів.

Тема 2. Основи Web-програмування з використанням ASP.NET

Розглядаються питання формування відгуку сервера і структура HTML-файлу, який отримуємо при цьому, а також питання виведення інформації у вікні браузера клієнта і особливості цього процесу. Наводиться ряд прикладів, які демонструють можливості організації введення клієнтом інформації та передачі її на сервер для подальшої обробки

Web-додатком. Розглядаються типові помилки, що виникають в процесі роботи з вихідним кодом Web-додатки, шляхи їх виявлення та виправлення. Вивчаються можливості динамічного створення елементів управління в Web-додатку і додавання їх на сторінку, створення обробників подій для них. Вивчається структура Web-додатки і таких її складових, як клас Page, колекція ViewState, об'єкти Session і Application. Розглядаються принципи роботи з файлами cookies.

Тема 3. Принципи розробки призначеного для користувача інтерфейсу інтернет-додатки.

Розглядаються питання створення призначеного для користувача інтерфейсу інтернет-додатки. Розглядаються принципи позиціонування елементів призначеного для користувача інтерфейсу, перераховуються основні інтерфейсні елементи, які використовуються в ASP.NET. Розглядаються принципи використання основних інтерфейсних елементів для виведення інформації на Web-сторінку, а також організації введення даних. Описуються можливості застосування каскадних таблиць стилів для оформлення зовнішнього вигляду елемента управління і всієї сторінки. Наводяться приклади динамічного керування вмістом сторінки з використанням елементів програмування, розглядаються найпростіші приклади здійснення прив'язки елементів управління до даних. Розглядаються базові принципи організації перевірки введених даних.

Тема 4. Використання Master Page і навігація при побудові інтернет-додатків.

Розглядаються принципи використання майстер-сторінок при розробці Інтернет-додатків. Наводяться приклади побудови типових майстер-сторінок і сторінок вмісту. Розглядаються можливості організації перемикачів між різними майстер-сторінками та програмного звернення до вмісту майстер-сторінок зі сторінок вмісту. Наводяться приклади використання вкладених майстер-сторінок.

Тема 4.1 Навігація по Web-додаткам.

Розглядаються питання організації навігації по сторінках Web-додатки, розбираються питання використання існуючих в ASP.NET елементів управління, призначених для організації ефективних механізмів навігації по Web-додаткам. Розглядаються можливості організації покрокових процесів збору інформації подібно до традиційних майстрів.

Тема 5. Використання тем при оформленні Web-додатки

Розглядаються питання стандартизації оформлення сторінок, включених в інтернет-додаток, за допомогою механізму тем, підтримуваного ASP.NET, наводяться приклади реалізації тем додатків для різних його елементів і сторінок. Порушуються питання динамічної зміни теми програми.

Тема 6. Використання кешування в Web-додатках.

Розглядаються питання використання різних видів кешування, підтримуваних ASP.NET. Наводяться приклади застосування кешування на сторінках ASP.NET додатки.

3. Структура навчальної дисципліни

Назви розділів і тем	Кількість годин					
	денна форма					
	усього	у тому числі				
л		п	лаб.	інд.	с. р.	
1	2	3	4	5	6	7
Розділ 1.						
Тема 1. Visual Studio. Net, Framework. Net.	2	2				
Тема 2. Система типів мови C #.	8,25	4		4		0,25
Тема 3. Процедури і функції - методи класу.	9	4		4		1
Тема 4. Класи Дві ролі класу в ООП.	6,5	2		4		0,5
Тема 5. Функціональний тип в C #. Делегати.	4,25	2		2		0,25
Тема 6. Універсальність. Класи з родовими параметрами.	4,25	2		2		0,25
Тема 7. Введення в windows-форми. Підготовка до контрольної роботи.	11	2		4		5
Тема 8. Робота з елементами управління.	4,5	2		2		0,5
Тема 9. Робота з елементами управління (продовження).	6,25	4		2		0,25
Разом за розділом 1	56	24		24		8
Розділ 2.						
Тема 1. Принципи роботи і структура Web-додатків на основі ASP.NET.	8	4		4		
Тема 2. Основи Web-програмування з використанням ASP.NET.	8,25	4		4		0,25
Тема 3. Принципи розробки призначеного для користувача інтерфейсу інтернет-додатки.	8,25	4		4		0,25
Тема 4. Використання Master Page і навігація при побудові інтернет-додатків.	8,25	4		4		0,25
Тема 5. Використання тем при оформленні Web-додатки. Підготовка до контрольної роботи.	13	4		4		5
Тема 6. Використання кешування в Web додатках.	8,25	4		4		0,25
Індивідуальне науково-дослідне завдання	40				40	
Разом за розділом 2	94	24		24	40	6
Усього годин	150	48		48	40	14

4. Теми семінарських (практичних, лабораторних) занять

Розділ 1		
№ з/п	Назва теми	Кількість годин
1	Перетворення типів. Змінні і вираження.	2
2	Вирази. Операції у вираженнях. Оператори мови C #.	2
3	Процедури і функції.	2
4	Масиви мови C #. Клас Array і нові можливості масивів.	2
5	Регулярні вираження.	2
6	Структури і перерахування.	2
7	Множинне спадкування.	2
8	Події. Делегати.	2
9	Обробка виняткових ситуацій.	2
10	Windows-форми.	2
11	Робота з елементами управління.	2
12	Організація інтерфейсу і малювання в формах.	2
Разом за розділом 1		24
Розділ 2		
13	Основи Web-програмування з використанням ASP.NET.	6
14	Принципи розробки призначеного для користувача інтерфейсу інтернет-додатки .	6
15	Використання Master Page і навігація при побудові інтернет-додатків.	6
16	Використання тем при оформленні Web-додатків.	6
Разом за розділом 2		24
Усього годин		48

5. Завдання для самостійної роботи

Розділ 1		
№ з/п	Види, зміст самостійної роботи	Кількість годин
	Завдання на самостійну роботу 1	
	«Створення елементів управління»	
	Мета роботи:	
	Вивчити основні елементи управління Windows-форм, їх властивості та методи, а також отримати практичні навички в розробці Windows-форм з елементами контролю.	
1	<ol style="list-style-type: none"> 1. Вивчити теоретичний матеріал. 2. Для форми FormEmployee створити необхідні елементи контролю. 3. Розробити методи для завдання режимів "Перегляд", "Редагування" для елементів контролю. 4. Розробити методи для завдання режимів "Перегляд", "Редагування" для управління активністю пунктів головного меню форми, контекстного меню і кнопок панелі інструментів. 5. Сформувавати обробник події Load. 6. Протестувати програму. 	1,5

№ з/п	Види, зміст самостійної роботи	Кількість годин
2	<p>Завдання на самостійну роботу 2 «Інтеграція звітів в Windows-додатки»</p> <p>Мета роботи: Освоїти технологію інтегрування і перегляду звітів з Windows-додатків, а також налаштування звітів на етапі виконання з використанням розвиненої об'єктної моделі.</p> <ol style="list-style-type: none"> 1. Вивчити теоретичний матеріал. 2. Розробити звіт і інтегрувати його з додатком. 3. Протестувати роботу програми зі звітом. 4. Продемонструвати роботу програми викладачеві. 5. Підготовка до контрольної роботи. 6. Розрахунково-графічна робота 	15
3	<p>Завдання на самостійну роботу 3 «Розгортання додатка»</p> <p>Мета роботи: Освоїти технологію створення інсталяційних програм Windows-додатків для розгортання створеного додатка.</p> <ol style="list-style-type: none"> 1. Вивчити теоретичний матеріал та розробити інсталятор для додатка. 2. Протестувати інсталятор. 3. Продемонструвати інсталяцію і деінсталяцію додатка викладачеві. 	1,5
Разом за розділом 1		18

Розділ 2

1	Створіть веб- додаток, що дозволяє визначити, чи використовується в операційній системі користувача російська мова, а так само попередні відвідані сторінки. Якщо використовується російська мова, то написати відповідь на російській, в іншому випадку відповісти англійською. Наприклад, повідомлення про мову може виглядати наступним чином: «Ви використовуєте російську мову!». Адреса попередньої сторінки оформите у заголовку, а якщо попередня сторінка не вказана, то виведіть відповідне повідомлення.	0,25
2	Створіть веб- додаток яке дозволяє вибудувати зростаючу послідовність з чисел переданих за поля введення форм.	0,25
3	Створіть веб- додаток, що дозволяє вибудувати спадаючу послідовність з чисел переданих за поля введення форм. Створіть веб- додаток, що дозволяє обчислити середнє арифметичне, а так само знайти максимальний елемент послідовності чисел, переданих через поля введення форми. Розрахунково-графічна робота	10
4	Створіть веб- додаток що дозволяє обчислити суму елементів, а так само знайти мінімальний елемент послідовності чисел переданих через поля введення форми. Створіть веб -додаток, що реалізує функцію зворотного зв'язку для вашого портфеля (формування і відправка mail).	0,25
5	Створіть сценарій, який генерує діаграму по введеному набору чисел (Використовуйте Microsoft Chart Controls for Microsoft. NET - http://www.microsoft.com/download/en/details.aspx?id=14422 - http://www.microsoft.com/download/en/details.aspx?id=23903)	0,25
6	Створіть веб- додаток, в якому сторінка з формою і обробник форми знаходяться в одному файлі. Використовуйте реєстраційну форму з декількох полів (мінімум 4) і прапорця (checkbox) позначає згоду з правилами поведінки або ліцензійною угодою. Якщо всі поля заповнені і прапорець встановлений, то результатом роботи скрипта має бути повідомлення про успішну реєстрацію нового користувача. Підготовка до контрольної роботи	5
7	Індивідуальне науково-дослідне завдання	20
Разом за розділом 2		16
Усього		54

6. Індивідуальні завдання

Контрольна робота –2
 Розрахунково-графічна робота –2
 Курсова робота

7. Методи навчання

Пояснювально- ілюстративні, репродуктивні, практичні.

8. Методи контролю

Протягом навчального семестру проводиться поточний контроль знань, який складається з виконання 16 лабораторних робіт, 2 контрольні роботи, 2 розрахунково-графічних робіт та курсової роботи. Загальна сума балів, яку студент може набрати, складає – 60 балів протягом навчального семестру.

Максимальний бал за лабораторну роботу – 2 бала.

Максимальна бал за кожну контрольну роботу – 4 балів.

Максимальний бал за кожну розрахунково-графічну роботу – 2 бала.

Курсова робота – максимальний бал 16 (включається в кількість балів за семестр).

Підсумкова форма контролю – екзамен.

Екзамен проводиться у вигляді тестування по 100 запитанням. На екзамені кожен студент одержує свій індивідуальний набір питань, які формуються дозволяючи з бази даних усіх запитань курсу (більш 200). Успішність проходження тесту оцінюється за національною шкалою

До підсумкового контролю допускаються тільки студенти, що мають зроблені усі самостійні роботи, і які в змозі відповісти на питання викладача по їх роботам.

Результати поточного та підсумкового тестування фіксуються в базі даних сервера MySQL. На протязі тестування ведеться протокол, де студент чи викладач має змогу проаналізувати відповіді студента на запитання тесту. Протоколи також зберігаються в базі даних та можуть бути роздруковані.

9. Схема нарахування балів

Поточний контроль, самостійна робота, індивідуальні завдання					Екзамен	Сума
Розділ 1	Розділ 2	Контрольна робота	Індивідуальне завдання	Разом		
T1-T9	T1-T6					
18	12	10	20	60	40	100

При контролі навчальних здобутків контроль здійснюється по модулям курсу без розподілу по темам.

**Критерії поточної оцінки знань студентів
(контрольна робота, крок оцінювання 1 бал)**

Кількість балів	Критерії оцінювання
0	Студент демонструє фрагментарні знання при незначному загальному їх обсязі (менше половини навчального матеріалу).
1	Студент знайомий з основними поняттями навчального матеріалу; може самостійно відтворити значну частину навчального матеріалу і робити певні узагальнення; вміє виконати просте навчальне завдання.
2	Студент демонструє вивчений матеріал у стандартних ситуаціях; пояснює основні процеси, що відбуваються під час роботи інформаційної системи та наводить власні приклади на підтвердження деяких тверджень; вміє виконувати навчальні завдання.
3	Студент демонструє міцні знання, самостійно визначає проміжні цілі власної навчальної діяльності, оцінює нові факти, явища; вміє самостійно знаходити додаткові відомості та використовує їх для реалізації поставлених перед ним навчальних цілей, судження його (її) логічні і достатньо обґрунтовані; має певні навички управління інформаційною системою.
4	Студент демонструє стійкі системні знання та продуктивно їх використовує; вміє вільно використовувати нові інформаційні технології для поповнення власних знань та розв'язування задач; має стійкі навички управління інформаційною системою у нестандартних ситуаціях.

**Критерії поточної оцінки знань студентів
(лабораторна робота, крок оцінювання 0,5 бал)**

Кількість балів	Визначення
2	Завдання по лабораторній роботі виконане самостійно в повному обсязі. Звіт оформлений акуратно відповідно до вимог методичних вказівок. При захисті звіту показано розуміння суті і змісту проведених досліджень
1,5	Завдання по лабораторній роботі виконане самостійно в повному обсязі. Звіт оформлений достатньо акуратно відповідно до вимог методичних вказівок. При захисті звіту були виявлені незначні помилки у знанні теоретичного матеріалу
1	Завдання по лабораторній роботі виконане в повному обсязі. Звіт оформлений достатньо акуратно, в оформленні звіту є незначні недоліки. При захисті звіту були виявлені незначні помилки у знанні теоретичного матеріалу
0,5	Завдання по лабораторній роботі виконане. Звіт оформлений з помилками і недоліками. При захисті звіту були виявлені суттєві помилки у знанні теоретичного матеріалу
0	Студент має фрагментарні знання при незначному загальному їх обсязі (менше половини навчального матеріалу) при відсутності сформованих умінь та навичок.

Критерії підсумкової оцінки знань студентів (екзамен)

Екзамен проводиться у вигляді тестування по 100 запитанням. Максимальна кількість набраних балів 40.

Відповідь на кожне питання тесту оцінюється у 0,4 бали.

Час на відповіді – 40 хвилин.

Усі запитання, на які не було дано відповіді оцінюються у 0 балів.

Шкала оцінювання

Сума балів за всі види навчальної діяльності протягом семестру	Оцінка
	для чотирирівневої шкали оцінювання
90 – 100	відмінно
70-89	добре
50-69	задовільно
1-49	незадовільно

10. Рекомендована література

Основна література

1. M. Metcalf, J. Reid., M. Cohen Modern Fortran Explained. – Oxford University Press, 2011. – 509 p.
2. S.J. Chapman Fortran 95 / 2003 for Scientists and Engineers. – 3-rd edition – McGraw-Hill, 2007. – 988 p.
3. S. Oliveira, D. E. Stewart Writing scientific software: a guide for good style – Cambridge University Press. 2006. – 316 p.
4. W. Cheney, D. Kincaid Numerical mathematics and computing – 6-th edition – Thomson Higher Education, 2008. – 789 p.
5. Shneiderman B. Designing the User Interface, 3-rd edn. – Reading, MA: Addison-Wesley, 1998.

Допоміжна література

1. P. O. J. Scherer Computational Physics. Simulation of Classical and Quantum Systems – 2-nd edition – Springer, 2013. – 456 p.
2. Pang An Introduction to Computational Physics. – 2-nd edition – Cambridge University Press – 2006, – 402 p.
3. Nicholas J. Giordano Computational physics. – Prentice Hall Inc., New Jersey – 1997, 419 p.

11. Посилання на інформаційні ресурси в Інтернеті, відео-лекції, інше методичне забезпечення

<http://www.c-sharpcorner.com>

- [Code::Blocks](#) — The free C/C++ and Fortran IDE.
- [SciDavis](#) — SciDAVis is a free application for Scientific Data Analysis and Visualization.
- [MagicPlot](#) - Software for nonlinear fitting, plotting and data analysis.
- [GFortran Manuals](#) — GFortran documentation is included with the GCC documentation. The GNU Fortran compiler, part of GCC.
- [Getting started with Fortran](#) - The GNU Fortran Compiler
- [The GNU Fortran Compiler](#) The GNU Fortran Compiler