

Методичні рекомендації з самостійної роботи студентів
з опанування навчального матеріалу
з дисципліни “ООП”

Самостійна робота студента полягає у опануванні додаткового теоретичного матеріалу до лекційних занять та у підготовці до виконання лабораторних робіт.

Тематичний план СРС навчальної дисципліни

№ п/п	Назва теми	СРС ОЧ	СРС ЗДН
1	2	3	4
1.1	Класи і об'єкти. Конструктори і деструктори (C++).	2	5
1.2	Основи C#. Класи і об'єкти. Конструктори і деструктори (C#).	3	5
1.3	Методи класів. Об'єкти як параметри. Масиви об'єктів. Параметри-посилання (C#).	2	5
1.4	Функції класів. Дружні функції. Об'єкти як параметри. Масиви об'єктів. Указівники і посилання (C++).	3	6
1.3	Перевантаження операторів. Індексатори. Властивості (C#).	3	6
1.4	Перевантаження операторів. Перевантаження оператора індекса масиву (C++).	2	5
1.4	Домашнє завдання № 1	8	8
1.5.	Модульна контрольна робота №1	4	
	Усього за модулем №1	27	40
2.1	Успадкування (C#)	3	5
2.2	Успадкування (C++)	2	5
2.3	Динамічний поліморфізм. Віртуальні функції. Абстрактні класи (C# та C++)	3	5
2.4	Інтерфейси C#. Об'єктно-орієнтовані принципи	3	5
2.5	Параметричний поліморфізм. Шаблони (templates).	3	10
2.6	Узагальнення (generics). Узагальнені методи і класи.	2	10
2.7	Узагальнення (generics). Узагальнений тип.	2	10
2.8	Оброблення виняткових ситуацій (C#)	2	10
2.9	Оброблення виняткових ситуацій (C++)	2	10
2.10	Система введення-виведення C++	1	10
2.11	Модульна контрольна робота №2	5	-
	Усього за модулем №2	28	80
	Усього за 2 семестр	55	120
3.1	Система введення-виведення C#.	3	7

1	2	3	4
	Потоки		
3.2	Система введення-виведення С#. Файлові потоки	3	8
3.3	Архітектура об'єктно-орієнтованих додатків	4	7
3.4	Колекції	4	7
3.5	Узагальнені колекції.	4	7
3.6	Узагальнені колекції. Бібліотека стандартних шаблонів.	4	7
3.7	Рядки і форматування	3	7
3.8	Серіалізація. Робота з XML.	4	7
3.9	Серіалізація. Робота з JSON. Архівація і стиснення файлів.	4	7
3.10	Делегати, лямбда-вирази.	3	7
3.11	Події	4	7
3.12	Динамічна ідентифікація типів, рефлексія і атрибути.	4	7
3.13	Технологія LINQ	4	7
3.14	Модульне тестування	5	7
3.15	Небезпечний код, вказівники, обнулювані типи.	4	7
3.16	Простори імен, препроцесор і збірки.	3	7
3.17	Модульна контрольна робота №3	7	-
3.18	Підсумкова семестрова контрольна робота	-	7
Усього за модулем №3		67	120
4.1	Виконання курсової роботи	30	30
Усього за модулем №4		30	30
Усього за навчальною дисципліною		152	270

Теми для самостійного опанування
(додатково до теоретичного та практичного матеріалу)

Модуль №1 «Абстрактні типи даних, інкапсуляція та статичний поліморфізм»

Тема 1.1. Класи і об'єкти. Конструктори і деструктори.

Поняття ООП. Історія ОО мов програмування. Мови С++ та С#. Основні принципи ООП. Інкапсуляція. Поліморфізм. Успадкування. Абстрактні типи даних. Поняття об'єкта.

С++: Класи. Закриті та відкриті члени класів. Функції (методи) класів. Об'єкти класів. Оператор розширення області видимості. Конструктори і деструктори. Конструктори з параметрами. Оголошення об'єктів. Перевантаження конструкторів. Взаємозв'язок класів і структур. Статичні (static) члени класу. Статичні змінні, статичні функції класу.

С#: Виникнення С#. Версії. Поняття .NET (.NET Core, .NET Standard). Поняття .NET Framework. CLR, BCL. IL. JIT. С# і .NET Framework. Керований і некерований код (managed and unmanaged code). CLS, CTS. Основи С#. С# та ООП. Области імен (namespaces). The System namespace. Класи у С#. Методи класів. Статичні методи

класів. Ключові слова. Ключові слова як ідентифікатори. Типи даних. Типи значень (value types). Типи посилання (reference types). Можливості виведення у консоль, форматування. Літерали і змінні. Область дії та час існування змінних. Перетворення та приведення типів. Класи і об'єкти. Консольний .NET проект у Visual Studio. Методи класів. Конструктори. Об'єкти класів. Відмінності об'єктів С++ та С#. Типи посилання та присвоєння. Збирання сміття та деструктори. Деструктори. Ключове слово this (у С++ та С#). Масиви. Цикл foreach. Рядки. Створення рядків. Методи рядків.

Тема 1.2. Функції класів. Об'єкти як параметри. Масиви об'єктів. Указівники і посилання.

С#: Управління доступом до класів. Модифікатори доступу. Передавання посилань на об'єкти до методів класів. Параметри ref, out. Змінна кількість аргументів. Повернення об'єктів. Повернення масивів. Перевантаження методів. Перевантаження конструкторів. Ініціалізатори об'єктів. Опціональні (необов'язкові) аргументи. Іменовані аргументи. Використання static. Статичні конструктори. Статичні класи.

С++: Дружні функції. Дружні класи. Вбудовані у рядок функції (inline). Статичні члени класів. Статичні члени-дані класів. Статичні члени-функції класів. Передавання об'єктів функціям-членам класу. Поняття конструктора копій. Повернення об'єктів із функцій. Присвоєння об'єктів. Масиви об'єктів. Указівники на об'єкти. Посилання. Параметри-посилання. Передавання посилань на об'єкти. Повернення посилань. Виділення пам'яті під об'єкти (оператор new). Конструктор копій. Аргументи за замовчуванням. Змінний список аргументів.

Тема 1.3. Перевантаження операторів. Індексатори.

С#: Основи перевантаження операторів. Методи операторів. Класи на UML-діаграмах. Перевантаження бінарних операторів. Перевантаження унарних операторів. Операції над убудованими типами даних. Перевантаження операторів відношення. Перевантаження true, false. Перевантаження логічних операторів. Оператори перетворення. Обмеження під час перевантаження операторів. Індексатори. Одновимірні індексатори. Аксесори set, get. Перевантаження індексаторів. Багатовимірні індексатори. Властивості.

С++: Оператор-функції. Оператор-функції як члени класів. Перевантаження бінарних операторів як членів класу. Перевантаження унарних операторів як членів класу. Перевантаження операторів як дружніх функцій. Перевантаження new, delete. Перевантаження оператора індекса масиву []. Перевантаження (). Перевантаження ->. Перевантаження оператора-коми. Функції-перетворення. Явні конструктори. Синтаксис ініціалізації членів класу.

Модуль №2 «Успадкування, динамічний та параметричний поліморфізм»

Тема 2.1. Успадкування.

С#: Успадкування. Поняття базового та похідного класів. Основи успадкування. Успадкування на UML-діаграмах. Доступ до членів класу під час успадкування. Захищені члени класу. Конструктори під час успадкування. Виклик конструкторів базового класу. Успадкування та скривання ідентифікаторів. Мультирівнева ієрархія. Коли викликаються конструктори.

С++: Успадкування. Поняття базового та похідного класів. Специфікатор доступу базового класу. Захищені члени класу. Захищене успадкування базового класу. Успадкування кількох базових класів. Конструктори, деструктори і успадкування. Передавання параметрів конструкторам базових класів. Множинне успадкування. Віртуальні базові класи.

Тема 2.2. Динамічний поліморфізм. Віртуальні функції. Абстрактні класи.

C#: віртуальні методи, абстрактні методи, абстрактні класи, посилання на абстрактні класи. Посилання на базові класи та похідні об'єкти. Віртуальні методи і перевизначення (overriding). Динамічна диспетчеризація методів. Поліморфізм під час виконання програм. Абстрактні класи. Абстрактні методи. Попередження успадкування (sealed). Клас object. Методи класу object. Упакування та розпакування.

C++: віртуальні функції, чисто віртуальні функції, абстрактні класи, вказівники на абстрактні класи. Поліморфізм. Віртуальні функції. Вказівники на базові класи і віртуальні функції. Чисто віртуальні функції. Абстрактні класи. Ранне та пізні зв'язування.

Тема 2.3. Інтерфейси C#.

Інтерфейси C#. Синтаксис. Обмеження. Реалізація інтерфейсів. Посилання на інтерфейси. Властивості інтерфейсів. Індикатори інтерфейсів. Успадкування інтерфейсів. Явна реалізація. Порівняння абстрактних класів та інтерфейсів. Інтерфейси .NET Standard. Методи інтерфейсів за умовчанням. Конкретні методи інтерфейсів. Перевизначення в інтерфейсах.

Тема 2.4. Об'єктно-орієнтовані принципи.

ОО принципи. ОО основи та ОО принципи. Інкапсуляція. Кодування інтерфейсів над імплементаціями. Зв'язність класів (cohesion). Принципи OCP, DRY, SRP, LSP. Альтернативи успадкуванню. Делегування, композиція, агрегація. Зв'язки між класами. UML-діаграми зв'язків між класами. Залежності. Зв'язок "is-a". Успадкування. Зв'язок "has-a". Асоціація. Агрегація. Композиція. Принципи проектування класів.

Тема 2.5. Параметричний поліморфізм. Шаблони (templates). Узагальнення (generics).

C++: Узагальнені функції (generic functions) або функції-шаблони. Узагальнений (фіктивний) тип даних. Спеціалізація або генерована функція. Конкретизація (instantiation). Функція кількох узагальнених типів. Явне перевантаження узагальненої функції. Перевантаження функції-шаблону. Звичайні параметри та параметри узагальненого типу. Узагальнені класи або класи-шаблони.

C#: Узагальнення. Параметризовані типи. Обмеження на створення узагальнень. Узагальнені методи. Обмеження на узагальнений тип. Узагальнені класи. Відкритий тип. Закритий тип. Сконструйований тип. Конкретизація узагальнених типів. Обмежені типи. Обмеження базовим класом. Обмеження інтерфейсами. Обмеження конструктором new(). Обмеження посиланнями та типами значень. Обмеження як зв'язок між параметризованими типами. Множинні обмеження. Значення за умовчанням параметризованого типу. Порівняння об'єктів (екземплярів) параметрів-типів. Інтерфейси IComparable, IComparable.

Тема 2.6. Оброблення виняткових ситуацій.

C#: Виняткові ситуації або винятки. Клас System.Exception. Клас Exception. Основи оброблення виняткових ситуацій. Блоки try, catch. Не спіймана виняткова ситуація. Множинні catch. Перехоплення усіх виняткових ситуацій. Вкладені блоки try. Викидання винятку. Блок throw. Повторне викидання винятку. Блок finally.

C++: перехоплення виняткових ситуацій. Оператори try, catch, throw. Перехоплення типів класів. Використання множинного перехоплення. Перехоплення виняткових ситуацій похідних класів. Перехоплення усіх помилок. Обмеження винятків. Повторне викидання винятку. Розуміння terminate(), unexpected(). Функція uncaught_exception(). Класи виняткових ситуацій: exception, bad_exception.

Тема 2.7. Система введення-виведення C++. Потоки

Потоки. Стандартні потоки введення-виведення C++. Форматування введення-виведення C++. Прапори формату. Маніпулятори. Перевантаження операторів вилучення і вставки. Маніпулятори користувача.

Модуль №3 «Об'єктно-орієнтована розробка програм»

Тема 3.1. Система введення-виведення C#.

Потоки. Байтові та символні потоки. Вбудовані потоки. Класи потоків. Класи Stream, System.IO, TextReader, TextWriter, Консольне введення-виведення. Клас FileStream. Файлове введення-виведення. Читання байтів. Символьне введення-виведення до файлу. Класи StreamWriter, StreamReader. Переадресація потоків. Класи BinaryWriter, BinaryReader. Файли із довільним доступом. Клас MemoryStream. Класи StringReader, StringWriter. Клас File. Парсинг. Регулярні вирази.

Тема 3.2. Архітектура об'єктно-орієнтованих додатків.

Багатошарова архітектура. Модель MVC. Зв'язок Model, View, Controller. Рівень доступу до даних (DAL). Рівень бізнес-логіки (BLL). Презентаційний рівень (PL). Зв'язок DAL, BLL, PL. Рівень сервісів (SL). Проектування зв'язків між типами. Слабке зв'язування і високе зв'язування (loose coupling and high cohesion). Інверсія контролю (inversion of control).

Тема 3.3. Рядки і форматування.

Клас String. Форматування. Використання методів форматування даних. Користувацький формат даних. Форматування дати і часу. Форматування проміжків часу. Перерахування. Форматування перерахувань.

Тема 3.4. Колекції, перерахування та ітератори.

Колекції C#. Простір імен System.Collections. Інтерфейси неузгаальнених колекцій. ICollection, IList, IDictionary, IComparer. Класи неузгаальнених колекцій. ArrayList, Hashtable, Queue, SortedList, Stack. Спеціальні колекції. Узгаальнені колекції. Інтерфейси узгаальнених колекцій. ICollection<T>, IList<T>, IDictionary<TKey, TValue>, IEnumerable<T>, IComparable<T>, ISet<T>, KeyValuePair<TKey, TValue>. Класи узгаальнених колекцій. List<T>, LinkedList<T>, Dictionary<TKey, TValue>, SortedDictionary<TKey, TValue>, Stack<T>, Queue<T>, HashSet<T>, SortedSet<T>. Паралельні колекції (поняття). Збереження об'єктів класів користувача у колекції. Реалізація інтерфейсів IComparer, IComparable, IEnumerable, IEnumerator. Ітератори. Оператор yield. Ініціалізатори колекцій.

Тема 3.5. Серіалізація.

Поняття серіалізації та десеріалізації. Загальний процес серіалізації. Збереження об'єкта. Простори імен серіалізації. System.Runtime.Serialization. Бінарна серіалізація. Binary. Soap. XML серіалізація. JSON серіалізація. Базова та користувачька серіалізація. Архівація та стиснення файлів.

Тема 3.6. Делегати, події, лямбда-вирази.

Поняття делегата. Вказівники на функції C++ та делегати C#. Групове перетворення делегованих методів. Використання методів екземпляра як делегатів. Групова адресація. Коваріантність та контрваріантність. Клас System.Delegate. Анонімні методи. Лямбда-вирази. Лямбда-оператор. Одиночні лямбда-вирази. Блочні лямбда-вирази. Події. Групова адресація події. Методи екземпляра та статичні методи як обробники подій. Аксесори подій. Анонімні методи та лямбда-вирази із подіями. Обробка подій у .NET Framework. Делегати EventHandler<T>, EventHandler.

Узагальнені структури. Узагальнені делегати. Узагальнені інтерфейси. Варіантні делегати. Коваріантність та контрваріантність узагальненого типу. Перерахування.

Тема 3.7. Динамічна ідентифікація типів, рефлексія і атрибути.

Динамічна ідентифікація типів. Рефлексія. Використання рефлексії. Атрибути. Вбудовані атрибути.

Тема 3.8. Технологія LINQ

Основи LINQ. Оператори запитів. Анонімні типи. Методи запитів. Древа запитів.

Тема 3.9. Модульне тестування.

Поняття модульного тестування (unit testing). Види тестувань програмного забезпечення. Логічне розташування тестів. Контроль версій та тести. Проекти з тестами. Класи тестів. Методи тестів. Використання фреймворків. XUnit, NUnit. Стиль написання тіла тесту. Модульні тести. Переваги та недоліки модульних тестів. Вимірювання успішності модульних тестів. Розроблення програмного забезпечення через тестування (TDD). Вимоги до тестового класа і метода. Типи класів Assert простору імен Microsoft.VisualStudio.TestTools.UnitTesting.

Тема 3.10. Небезпечний код, вказівники, обнулювані типи.

Небезпечний код. Обнулювані типи. Часткові типи. Часткові методи. Динамічний тип. Дружні збірки. Ключові слова.

Тема 3.11. Простори імен препроцесор та збірки.

Простори імен. Адитивність. Глобальний простір. Вкладеність просторів імен. Препроцесор. Директиви. Збірки і модифікатори доступу.

СПИСОК РЕКОМЕНДОВАНИХ ДЖЕРЕЛ

Базова література

1. Schildt H. C# 4.0: The Complete Reference. eBook, McGraw-Hill, 2010. 949 p.
2. Schildt H. C++: The Complete Reference, Fifth edition. McGraw-Hill, 2012. 1200p.
3. Deitel H.M., Deitel P.J. C++ How to Program, Fifth Edition. Prentice Hall, 2005. 1536 p.
4. Lippman, S.B., Lajotte J., Moo B.E. C++ primer, 5th ed. Pearson, 2013. 938p.
5. Троедсен Э., Джепикс Ф. Язык программирования C# 7 и платформы .NET и .NET Core. СПб.: Диалектика, 2018. 1328 с.

Допоміжна література

6. Josuttis N.M. C++17 – The Complete Guide. NicoJosuttis, 2019. 370p.
7. Skeet J. C# in Depth, Fourth Edition. Manning, Shelter Island, 2019. 497p.
8. Booch G. et.al. Object-Oriented Analysis and Design with Applications, Third Ed.. Addison-Wesley, Pearson, 2007. 692 p.
9. Страуструп Б. Язык программирования C++. Краткий курс. СПб.: Диалектика, 2019. 320 с.
10. McLaughlin B.D., West D. Head First Object-Oriented Analysis and Design. O'Reilly Media, 2007. 636 p.

3.3. Інформаційні ресурси в інтернеті

1. <https://docs.microsoft.com/ru-ru/dotnet/csharp/>
2. <https://metanit.com/sharp/tutorial/>
3. <https://www.w3schools.com/cpp/default.asp>
4. <https://www.w3schools.com/cs/default.asp>