



# СИЛАБУС НАВЧАЛЬНОЇ ДИСЦИПЛІНИ



## «Кросплатформне програмування»

Рівень освіти	Бакалавр	Тип дисципліни	Нормативна. Професійна
Шифр та назва спеціальності	122 – Комп'ютерні науки	Факультет / Інститут	ННІ КНІТ Навчально науковий інститут комп'ютерних наук та інформаційних технологій
Назва освітньо-наукової програми	Комп'ютерні науки	Кафедра	Системний аналіз та інформаційно-аналітичні технології

### ВИКЛАДАЧ



Марченко Ігор Іванович, [igor.marchenko@khpi.edu.ua](mailto:igor.marchenko@khpi.edu.ua)

Кандидат технічних наук, доцент, доцент кафедри системного аналізу та інформаційно-аналітичних технологій НТУ «ХПІ». Досвід роботи – 15 років. Автор понад 70 наукових та навчально-методичних праць. Провідний лектор з дисциплін: «Теорія ймовірностей», «Математична статистика», «Кросплатформне програмування»

Персональна сторінка: <https://web.kpi.kharkov.ua/say/uk/uaabout/uaprofs/marchenkoii/>

### ЗАГАЛЬНА ІНФОРМАЦІЯ ПРО ДИСЦИПЛІНУ

Анотація	Дисципліна спрямована на оволодіння теоретичних основ методики, яка реалізується через лекційний матеріал та лабораторні роботи, технології, що використовуються для створення кросплатформних програмних засобів зі складно архітектурою; підготувати розробників програмного проекту; підготувати студентів к самостійному вивченню довідкової та науково-технічної літератури за фахом. Розглянуто взаємозв'язок та особливості окремих підходів до проектування ПЗ, обговорено їх можливості та обмеження.
Мета та цілі	Ознайомити студентів із сучасними технологіями розробки програмного забезпечення для різних платформ; дати студентам основні концепції .NET; навчити студентів створенню програмних засобів на базі мови C#.
Формат	Лекції, лабораторні роботи, , самостійна робота, консультації. Підсумковий контроль – екзамен.
Результати навчання	Розробляти програмні моделі предметних середовищ, вибирати парадигму програмування з позицій зручності та якості застосування для реалізації методів та алгоритмів розв'язання задач в галузі комп'ютерних наук
Обсяг	Загальний обсяг дисципліни 180 год.: лекції – 48 год., лабораторні роботи – 32 год., самостійна робота – 100 год.
Пререквізити	«Об'єктно-орієнтовне програмування»

**Вимоги викладача**

Студент зобов'язаний відвідувати всі заняття згідно розкладу, не спізнюватися. Дотримуватися етики поведінки. Працювати з навчальної та додатковою літературою, з літературою на електронних носіях і в Інтернеті. При пропуску лекційних занять проводиться усна співбесіда за темою. Відпрацьовувати лабораторні заняття при наявності допуску викладача. З метою оволодіння необхідною якістю освіти з дисципліни потрібно відвідуваність і регулярна підготовленість до занять. Без особистої присутності студента підсумковий контроль не проводиться.

**СТРУКТУРА ДИСЦИПЛІНИ**

<b>Лекція 1</b>	Основні концепції .NET. Базовий синтаксис C#: Базові типи, Цикли.	<b>Лабораторна робота 1</b>	Створення консольного застосування у C#	<b>Самостійна робота</b>	Статичні класи та методи
<b>Лекція 2</b>	Базовий синтаксис C#: Масиви, Строки.				
<b>Лекція 3</b>	Поля, методи, конструктори/деструктори.	<b>Лабораторна робота 2</b>	Робота зі строками та масивами як з об'єктами у C#		
<b>Лекція 4</b>	Інкапсуляція. Властивості (Properties).				
<b>Лекція 5</b>	Спадкування	<b>Лабораторна робота 3</b>	Розробка базових класів		Запечатані класи
<b>Лекція 6</b>	Поліморфізм, Абстрактні класи	<b>Лабораторна робота 4</b>	Розробка ієрархії класів		
<b>Лекція 7</b>	Перевантаження арифметичних операторів				Перевантаження явних та неявних операцій переведення типів
<b>Лекція 8</b>	Перевантаження операторів порівняння	<b>Лабораторна робота 5</b>	Розробка класу-контейнера для збереження ієрархії класів		
<b>Лекція 9</b>	Структури	<b>Лабораторна робота 6</b>	Реалізація структур		
<b>Лекція 10</b>	Перерахування				
<b>Лекція 11</b>	Структурна обробка виняткових ситуацій	<b>Лабораторна робота 7</b>	Реалізація обробки виняткових ситуацій. Частина 1		
<b>Лекція 12</b>	Контрольна робота №1	<b>Лабораторна робота 8</b>	Реалізація обробки виняткових ситуацій. Частина 2		
<b>Лекція 13</b>	Інтерфейси				
<b>Лекція 14</b>	Операції упаковки-розпакування. Універсальні класи	<b>Лабораторна робота 9</b>	Розробка інтерфейсів		Паттерни розробки. Паттерн "interface"
<b>Лекція 15</b>	Шаблон програмування «ітератор». Генератори	<b>Лабораторна робота 10</b>	Розробка універсального класу-контейнеру		
<b>Лекція 16</b>	Базова робота з файлами. Потoki даних.				

<b>Лекція 17</b>	Відображення (Reflection).	<b>Лабораторна робота 11</b>	Реалізація ітераторів		
<b>Лекція 18</b>	Серіалізація	<b>Лабораторна робота 12</b>	Робота з текстовими файлами		Робота з оперативною пам'яттю як з потоком даних
<b>Лекція 19</b>	Делегати				
<b>Лекція 20</b>	Лямбда-вирази	<b>Лабораторна робота 13</b>	Реалізація серіалізації		
<b>Лекція 21</b>	Події	<b>Лабораторна робота 14</b>	Реалізація делегатів		
<b>Лекція 22</b>	LINQ				
<b>Лекція 23</b>	Entity framework	<b>Лабораторна робота 15</b>	Обробка подій. Частина 1		
<b>Лекція 24</b>	Контрольна робота №2	<b>Лабораторна робота 16</b>	Обробка подій. Частина 2		

## ЛІТЕРАТУРА ТА НАВЧАЛЬНІ МАТЕРІАЛИ

<b>Основна</b>	1. Коноваленко І.В. Платформа .NET та мова програмування C# 8.0: навчальний посібник / Коноваленко І.В., Марущак П.О. – Тернопіль: ФОП Паляниця В. А., 2020, 320 с.	<b>Додаткова</b>	1. Томашевський В.В. Технології програмування. Мова C#: навч.посібн. /В.В. Томашевський. – Житомир: ЖВІ НАУ, 2012. – 484 с.
	2. Коноваленко І.В. Програмування мовою C# 6.0 / Коноваленко І.В.– Тернопіль : ТНТУ, 2016, 227 с.		
	3. C# programming guide : [Електронний ресурс]. URL: <a href="https://docs.microsoft.com/uk-ua/dotnet/csharp/programming-guide/index">https://docs.microsoft.com/uk-ua/dotnet/csharp/programming-guide/index</a>		
	4. C# reference : [Електронний ресурс]. URL: <a href="https://docs.microsoft.com/uk-ua/dotnet/csharp/language-reference/index">https://docs.microsoft.com/uk-ua/dotnet/csharp/language-reference/index</a>		
	5. Using the Visual Studio Development Environment for C# : [Електронний ресурс]. URL: <a href="https://docs.microsoft.com/uk-ua/visualstudio/csharp-ide/using-the-visual-studio-development-environment-for-csharp">https://docs.microsoft.com/uk-ua/visualstudio/csharp-ide/using-the-visual-studio-development-environment-for-csharp</a>		

## ПЕРЕЛІК ЗАПИТАНЬ ДЛЯ ПІДГОТОВКИ ДО ІСПИТУ

1.Основні концепції NET.Framework. 2.Базові типи. 3.Цикли. 4.Масиви. 5.Поля. 6.Методи. 7.Конструктори/деструктори. 8.Інкапсуляція. Властивості. 9.Статичні методи. 10.Спадкування. 11.Поліморфізм. 12.Абстрактні класи. 13.Запечатані класи. 14.Перевантаження арифметичних операторів і операторів порівняння. 15.Перевантаження явних та неявних операцій переведення типів. 16.Структури.17.Перерахування. 18.Структурна обробка виняткових ситуацій.19.Інтерфейси. 20.Операції упаковки-розпакування. 21.Універсальні класи. 22.Шаблони програмування.23.Ітератори. 24.Базова робота з файлами. 25.Потоки даних. 26.Відображення (Reflection). 27.Серіалізація. 28.Делегати. 29.Лямбда-вирази. 30.Події. 31.LINQ.

## ПЕРЕЛІК ОБЛАДНАННЯ

Комп'ютер

## СИСТЕМА ОЦІНЮВАННЯ

Розподіл балів для оцінювання успішності аспіранта	Сума балів за всі види навчальної діяльності	Оцінка ECTS	Оцінка за національною шкалою	Нарахування балів
	90-100	A	Відмінно	
	82-89	B	добре	
	74-81	C		
	64-73	D	задовільно	
	60-63	E		
	35-59	FX	незадовільно з можливістю повторного складання	
	0-34	F	незадовільно з обов'язковим повторним вивченням дисципліни	

Бали нараховуються за наступним співвідношенням:

- лабораторні роботи: 20% семестрової оцінки;
- самостійна робота: 20% семестрової оцінки;
- іспит: 60% семестрової оцінки

## НОРМИ АКАДЕМІЧНОЇ ЕТИКИ

Студент повинен дотримуватися «Кодексу етики академічних взаємовідносин та доброчесності НТУ «ХПІ»»: виявляти дисциплінованість, вихованість, доброзичливість, чесність, відповідальність. Конфліктні ситуації повинні відкрито обговорюватися в навчальних групах з викладачем, а при нерозв'язності конфлікту доводиться до завідувача кафедри.

Силлабус за змістом повністю відповідає робочій програмі навчальної дисципліни