

Міністерство освіти і науки України

Національний аерокосмічний університет ім. М.Є. Жуковського  
«Харківський авіаційний інститут»

Кафедра інформаційних технологій проектування (№ 105)

**ЗАТВЕРДЖУЮ**

Голова НМК 2

  
(підпис) Д.М.Крицький  
(ініціали та прізвище)

« 31 » 08 2021 р.

**РОБОЧА ПРОГРАМА ВИБІРКОВОЇ  
НАВЧАЛЬНОЇ ДИСЦИПЛІНИ**

**Компонентне програмування**

(назва навчальної дисципліни)

Галузі знань: 12 “Інформаційні технології”  
(шифр і найменування галузі знань)

Спеціальності: 126 “Інформаційні системи та технології”  
(код та найменування спеціальності)

Освітні програми: Інформаційні системи та технології  
підтримки віртуальних середовищ  
(найменування освітньої програми)

**Форма навчання: денна**

**Рівень вищої освіти: перший (бакалаврський)**

**Харків 2021 рік**

Робоча програма Компонентне програмування

(назва дисципліни)

для студентів за спеціальністю 126 “Інформаційні системи та технології”

Освітня програма: Інформаційні системи та технології підтримки  
віртуальних середовищ

(найменування освітньої програми)

«08» 06 2021 р., - 14 с.

Розробник: Овсяннік В.М., доцент к.т.н.,

(прізвище та ініціали, посада, наукова ступінь та вчене звання)



(підпис)

Робочу програму розглянуто на засіданні кафедри  
інформаційних технологій проектування

(назва кафедри)

Протокол № 1 від «31» серпня 2021 р.

Завідувач кафедри

к.т.н., доцент

(науковий ступінь та вчене звання)



(підпис)

Д.М.Крицький

(ініціали та прізвище)

## 1. Опис навчальної дисципліни

Найменування показників	Галузь знань, напрям підготовки (спеціальність, спеціалізація), рівень вищої освіти	Характеристика навчальної дисципліни (денна форма навчання)
Кількість кредитів – 6,5	<b>Галузь знань</b> <u>12 «Інформаційні технології»</u> <small>(шифр і назва)</small>  <b>Спеціальності</b> <u>126 «Інформаційні системи та технології»</u> <small>(шифр і назва)</small>  <b>Освітні програми</b> <u>Інформаційні системи та технології</u> <u>підтримки віртуальних середовищ</u> <small>(найменування освітньої програми)</small>  <b>Рівень вищої освіти:</b> перший (бакалаврський)	Цикл професійної підготовки за вибором
Модулів – 2		<b>Навчальний рік:</b> 2021 / 2022
Змістових модулів – 4		<b>Семестр</b> 4-й
Загальна кількість годин – 96/195		<b>Лекції<sup>2)</sup></b> 48 години
Тижневих годин для денної форми навчання: аудиторних – 6 самостійної роботи студента – 6,1		<b>Практичні, семінарські</b>  <b>Лабораторні<sup>2)</sup></b> 48 годин <b>Самостійна робота</b> 99 години <b>Вид контролю</b> іспит

Співвідношення кількості годин аудиторних занять до самостійної роботи становить для денної форми  $96/99=0.97$ .

<sup>2)</sup> Аудиторне навантаження може бути зменшеним або збільшеним на одну годину в залежності від розкладу занять.

## 2. Мета та завдання навчальної дисципліни

Мета вивчення навчальної дисципліни полягає у підготовці спеціалістів з інформаційних систем до виконання робіт з розроблення програмного забезпечення з використанням принципів та методів об'єктно-орієнтованого програмування.

Основними завданнями дисципліни є вивчення засобів розробки програмного забезпечення для роботи під керівництвом ОС Windows з широким використанням можливостей об'єктно-орієнтованого програмування, правила будування програмних засобів в середовищах візуального програмування, налагодження програм та розв'язання типових задач

**Згідно з вимогами освітньо-професійної програми студенти повинні досягти таких загальних та фахових компетентностей:**

КЗ 2. Здатність застосовувати знання у практичних ситуаціях.

КЗ 3. Здатність до розуміння предметної області та професійної діяльності.

КС 2. Здатність застосовувати стандарти в області інформаційних систем та технологій при розробці функціональних профілів, побудові та інтеграції систем, продуктів, сервісів і елементів інфраструктури організації.

### **Програмні результати навчання:**

ПР 4. Проводити системний аналіз об'єктів проектування та обґрунтовувати вибір структури, алгоритмів та способів передачі інформації в інформаційних системах та технологіях.

ПР 9. Здійснювати системний аналіз архітектури підприємства та його ІТ-інфраструктури, проводити розроблення та вдосконалення її елементної бази і структури.

ПР 10. Розуміти і враховувати соціальні, екологічні, етичні, економічні аспекти, вимоги охорони праці, виробничої санітарії, пожежної безпеки та існуючих державних і закордонних стандартів під час формування технічних завдань та рішень.

ПР 11. Демонструвати вміння розробляти техніко-економічне обґрунтування розроблення інформаційних систем та технологій та вміння оцінювати економічну ефективність їх впровадження.

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

- алгоритмічну мову програмування C++ та C++/CLI;
- найбільш поширені інструментальні засоби для програмування у середовищі Microsoft Visual Studio;
- особливості роботи програм під керуванням Windows;
- правила будування програм в візуальних середовищах програмування;

### **уміти:**

- створювати програмні засоби, що працюють під керуванням Windows;
- розробляти інтерфейси користувача за допомогою засобів середовищ візуального програмування;
- програмувати елементи інтерфейсу користувача з використанням керуючих елементів, що є типовими для сучасних застосунків;
- використовувати основні програмні механізми при програмуванні інтерфейсів користувача;
- проектувати та програмувати ієрархію класів для вирішення типових задач систем автоматизованого проектування (САПР).

Студент має надбати **навички** самостійного проектування, розробки та налагодження застосунків, призначених для розв'язання типових математичних та прикладних задач, що виникають під час розробки функціональних складових модулів САПР та застосунків іншого призначення.

**Міждисциплінарні зв'язки:** навчальна дисципліна розрахована на студентів, що мають підготовку по інформатиці і програмуванню, знання основних базових алгоритмів і базових конструкцій мов програмування. Знання та уміння можуть бути використані для розробки алгоритмів і програм на базі об'єктно-орієнтованого підходу. Викладання

дисципліни базується на курсах «Основи програмування», «Алгоритмізація і програмування» та «Об'єктно-орієнтоване програмування». Знання та навички, що здобувають студенти під час вивчення дисципліни, використовуються при вивченні таких дисциплін, як «Технологія розробки програмних продуктів», «Сучасні технології та інструментарій програмування», «Організація баз даних та знань».

### **3. Програма навчальної дисципліни**

#### **Модуль 1. Засоби мови C++, що значно розширюють її можливості.**

##### **Змістовий модуль 1. Перевантаження операцій**

###### **Тема 1. Вступ до дисципліни.**

Ціль і задачі курсу. Види занять та їх призначення, модульно-рейтингова система. Рекомендована та додаткова література, у тому числі у електронному форматі. Основні етапи та складові розвитку візуальних систем розробки застосувань Windows.

###### **Тема 2. Статичні компоненти класів.**

Призначення статичних членів класів, правила їх опису та використання, особливі властивості. Приклади використання статичних членів класів для розв'язання типових задач, що виникають у процесі розробки модулів САПР.

###### **Тема 3. Дружні функції та класи.**

Призначення дружніх функцій класів, правила їх опису та використання, особливі властивості. Опис та використання дружніх класів. Сфери та приклади застосування дружніх функцій та класів.

###### **Тема 4. Перевантаження операцій.**

Поняття та правила перевантаження операцій. Формат операцій-функцій, способи їх реалізації. Перелік операцій, що не підлягають перевантаженню. Залежність способів оформлення операцій-функцій від арістії операцій та їх призначення. Приклади класів, у яких використовується перевантаження операцій. Переваги перевантаження операцій.

#### **Змістовий модуль 2. Важливі властивості системи Windows та їх використання**

##### **Тема 5. Адресні простори, застосування, процеси та потоки.**

Поняття та визначення адресного простору, застосування, процесу та потоку. Взаємодії та взаємовідносини процесів та потоків. Синхронізація потоків. Класифікація пріоритетів процесів та потоків Windows. Використання потоків Windows для вирішення практичних задач.

##### **Тема 6. Оброблення виняткових ситуацій.**

Види помилок у програмах. Проблеми традиційного підходу до обробки помилок. Механізм обробки виключень. Класи виключень стандартної бібліотеки мови C++. Синтаксис обробки виключень. Розробка власних класів оброблення виняткових ситуацій.

#### **МОДУЛЬ 2. Стандартна бібліотека C++ та елементи керування**

##### **Змістовий модуль 3. Стандартна бібліотека мови C++**

###### **Тема 7. Інтелектуальні покажчики.**

Недоліки використання «звичайних» покажчиків. Перелік помилок, що зазвичай виникають при використанні «звичайних» покажчиків та їх наслідки. Головна ідея інтелектуальних покажчиків та способи її реалізації. Переваги, сфери та приклади застосування інтелектуальних покажчиків.

## Тема 8. Класифікація засобів стандартної бібліотеки мови C++.

Поняття та класифікація засобів бібліотеки STL (Standard Template Library). Перелік та загальні характеристики шаблонних класів бібліотеки STL. Найбільш поширені засоби бібліотеки STL, їх опис та приклади використання.

## Тема 9. Контейнери стандартної бібліотеки та їх використання.

Класифікація та характеристики контейнерів стандартної бібліотеки, опис доцільності їх використання, перелік переваг та недоліків. Класифікація та опис найбільш поширених функцій контейнерів, приклади їх використання.

## Тема 10. Функціональні об'єкти та їх використання у алгоритмах стандартної бібліотеки.

Поняття функціонального об'єкту, його призначення та сфера використання. Опис та реалізація функціональних об'єктів. Поняття предикатів. Приклади використання функціональних об'єктів у застосуваннях, що використовують алгоритми стандартної бібліотеки.

## Змістовий модуль 4. Використання найбільш поширених елементів керування Windows

### Тема 11. Реалізація та використання немодальних діалогових вікон.

Визначення та призначення немодальних діалогових вікон, їх відмінності від модальних вікон. Способи та приклади реалізації та використання немодальних діалогових вікон, їх переваги та недоліки. Взаємодія застосування Windows з немодальними діалоговими вікнами.

### Тема 12. Використання елементів керування ActiveX у застосуваннях Windows.

Історична довідка. Визначення та призначення елементів керування ActiveX, їх відмінності від «звичайних» елементів керування. Способи та приклади реалізації та використання елементів керування ActiveX, їх переваги та недоліки.

## 4. Структура навчальної дисципліни

Назви змістових модулів і тем	Кількість годин					
	денна форма					
	усього	у тому числі				
л		п	лаб	курс	с.р.	
<b>Модуль 1. Засоби мови C++, що значно розширюють її можливості</b>						
<b>Змістовий модуль 1. Перевантаження операцій</b>						
Тема 1. Вступ до дисципліни	2	2		–		–
Тема 2. Статичні компоненти класів	20	6		6		8
Тема 3. Дружні функції та класи.	16	4		4		8
Тема 4. Перевантаження операцій	16	4		4		8
Разом за змістовим модулем 1	54	18		14		24
<b>Змістовий модуль 2. Важливі властивості системи Windows та їх використання</b>						
Тема 5. Адресні простори, застосування, процеси та потоки	22	4		6		12
Тема 6. Оброблення виняткових ситуацій	22	4		6		12
Разом за змістовим модулем 2	44	8		12		24
<b>Усього годин за модулем 1</b>	<b>98</b>	<b>24</b>		<b>26</b>		<b>48</b>

Назви змістових модулів і тем	Кількість годин					
	денна форма					
	усього	у тому числі				
		л	п	лаб	курс	с.р.
<b>Модуль 2. Стандартна бібліотека C++ та елементи керування</b>						
<b>Змістовий модуль 3. Стандартна бібліотека мови C++</b>						
Тема 7. Інтелектуальні показники	12	4		4		4
Тема 8. Класифікація засобів стандартної бібліотеки мови C++	4	4		–		–
Тема 9. Контейнери стандартної бібліотеки та їх використання	22	4		4		14
Тема 10. Функціональні об'єкти та їх використання у алгоритмах стандартної бібліотеки	20	4		6		10
Разом за змістовим модулем 3	58	16		14		28
<b>Змістовий модуль 4. Використання найбільш поширених елементів керування Windows</b>						
Тема 11. Реалізація та використання немодальних діалогових вікон	18	4		4		10
Тема 12. Використання елементів керування ActiveX у застосуваннях Windows	21	4		4		13
Разом за змістовим модулем 4	39	8		8		23
<b>Усього годин за модулем 2</b>	97	24		22		51
<b>Усього годин за дисципліною</b>	195	48		48		99

### 5. Теми семінарських занять

Семінарські заняття навчальним планом не передбачені.

### 6. Теми практичних занять

Практичні заняття навчальним планом не передбачені.

### 7. Теми лабораторних занять

№ з/п	Назва теми	Кількість годин
1	Розробка типового консольного додатка	6
2	Дослідження похибок обчислень	6
3	Обробка текстового файлу	6
4	Пошук даних у масиві	6
5	Додаток з вікном виду	6
6	Розробка графічного редактора	6
7	Додаток, заснований на модальному діалоговому вікні	6
8	Клас вектор	6
	Разом	48

## 8. Самостійна робота

№ з/п	Назва теми	Кількість годин
1	Статичні компоненти класів	8
2	<u>Дружні функції та класи.</u> Опис та використання дружніх класів. Застосування дружніх функцій та класів для обраної предметної області.	8
3	<u>Перевантаження операцій.</u> Особливості перевантаження операцій доступу до елементів масивів та операцій присвоєння. .	8
4	<u>Адресні простори, застосування, процеси та потоки.</u> Засоби синхронізації потоків, що надаються Windows	12
5	<u>Оброблення виняткових ситуацій.</u> Розробка власних класів оброблення виняткових ситуацій	12
6	Інтелектуальні покажчики	4
7	<u>Контейнери стандартної бібліотеки та їх використання.</u> Застосування контейнерів типу вектор та асоціативних масивів для обробки масивів об'єктів користувача	14
8	<u>Функціональні об'єкти та їх використання у алгоритмах стандартної бібліотеки.</u> Самостійна розробка функціональних об'єктів, що призначені для обробки масивів об'єктів користувача	10
9	<u>Реалізація та використання немодальних діалогових вікон.</u> Доступ до даних, що вводяться за допомогою немодальних діалогових вікон	10
10	<u>Використання елементів керування ActiveX у застосуваннях Windows.</u> Використання елементів ActiveX, що призначені для відображення даних у графічній формі	13
	Разом	99

## 9. Курсовий проект

Курсовий проект навчальним планом не передбачено.

## 10. Методи навчання

**Лекції** проводяться з використанням основних розділів конспекту лекцій в електронній формі.

**Лабораторні роботи** виконуються з використанням навчальних та ліцензованих версій Microsoft Visual Studio, методичних рекомендацій та детальних сценаріїв у друкованому та електронному вигляді.

**Самостійна робота** включає підготовку до лабораторних робіт, модульних контролів та іспиту, виконання позааудиторної частини індивідуального завдання з курсового проекту і вивчення вказаних вище тем за конспектом, літературними і електронними джерелами та програмною документацією.

## 11. Методи контролю

Поточний контроль здійснюється відповідно до повноти, своєчасності та якості виконання лабораторних робіт та курсового проектування згідно з інструкцією з модульно-рейтингового оцінювання знань студентів; проміжний (модульний) контроль – письмові контрольні роботи на 9-му та 16-му тижнях; підсумковий контроль – письмовий іспит.



## 12. Критерії оцінювання та розподіл балів, які отримують студенти

### 12.1. Розподіл балів, які отримують студенти (кількісні критерії оцінювання)

Складові навчальної роботи	Бали за одне заняття (завдання)	Кількість занять (завдань)	Сумарна кількість балів
<b>Змістовний модуль 1</b>			
Робота на лекціях	0...1	5	0...5
Виконання і захист лабораторних (практичних) робіт	4...6	4	16...25
Модульний контроль	14...20	1	14...20
<b>Змістовний модуль 2</b>			
Робота на лекціях	0...1	5	0...5
Виконання і захист лабораторних (практичних) робіт	4...6	4	16...25
Модульний контроль	14...20	1	14...20
<b>Усього за семестр</b>			<b>60...100</b>

### 12.2. Якісні критерії оцінювання

Необхідний обсяг знань для одержання позитивної оцінки:

- мати уявлення про весь теоретичний матеріал дисципліни;
- знати в повному обсязі не менш половини тем теоретичного матеріалу;
- знати додатковий матеріал по трьом темам.

Необхідний обсяг умінь для одержання позитивної оцінки:

- володіти технологією застосування теоретичних знань на практиці;
- уміти самостійно знаходити довідкові відомості;
- уміти застосовувати на практиці отримані знання.

### 12.3 Критерії оцінювання роботи студента протягом семестру

**Задовільно (60-74).** Знати основні положення теоретичного матеріалу. Вміти користуватися засобами інструментального середовища Visual Studio. Вміти розробляти додатки Windows, що вирішують задачі з використанням таких типів даних, як шаблонні класи.

**Добре (75-89).** Знати основний теоретичний матеріал в повному обсязі. Володіти технологією пошуку довідкової літератури. Вміти розробляти та налагоджувати додатки в інструментальному середовищі Visual Studio, що використовують шаблонні класи STL.

**Відмінно (90-100).** Знати основний і додатковий теоретичний матеріал в повному обсязі. Добре орієнтуватися в довідковій літературі. Вміти вирішувати задачі та розробляти додатки з використанням платформи .NET FrameWork. Вміти розробляти додатки, що використовують бібліотек WinForms.

Переведення підсумкових семестрових оцінок в оцінки за національною шкалою та в оцінки за міжнародною шкалою ECTS здійснюється відповідно до «Положення про модульно-рейтингову систему оцінювання знань студентів» за наведеною нижче таблицею.

### Шкала оцінювання: національна та ECTS

Сума балів за всі види навчальної діяльності	Оцінка ECTS	Оцінка за національною шкалою
90–100	A	відмінно
83–89	B	добре
75–82	C	
68–74	D	задовільно
60–67	E	
1–59	FX	незадовільно з можливістю повторного складання

### 13. Методичне забезпечення

1. Овсянник В.Н. Инструментальные средства визуального программирования. Конспект лекций. – [Электронный ресурс] – Режим доступа: – 10.1.108.193.(01.09.2019)
2. Овсянник В.Н. Лабораторные работы по курсу «Инструментальные средства визуального программирования» – [Электронный ресурс] – Режим доступа: – 10.1.108.193.(01.09.2019)
3. Овсянник В.Н. Курсовой проект по дисциплине «Инструментальные средства визуального программирования» – [Электронный ресурс] – Режим доступа: – 10.1.108.193.(01.09.2019)

### 14. Рекомендована література

#### 14.1. Базова

4. Кравець П.О. Об'єктно-орієнтоване програмування: навч. посібник/ П.О.Кравець. – Львів: Видавництво Львівської політехніки, 2014. – 624с.
5. Юрченко І.В., Сікора В.С. Інформатика та програмування. Частина 2.– Чернівці: Видавець Яворський С.Н., 2015.– 210 с.
6. Г.Шилдт. С++: базовий курс / Пер. с англ. – М.: "Диалектика Вильямс", 2018. – 624с..
7. Овсянник В.Н. Язык С++ не для чайников. Учебное пособие – [Электронный ресурс] – Режим доступа: – 10.1.108.193.(01.09.2019)

#### 14.2. Допоміжна

8. Дьюхерст С. С++. Священные знания, 2-е издание. – Пер. с англ. – СПб.: Символ-Плюс, 2016. – 240с.
9. Джеффри Рихтер. CLR via C#. Программирование на платформе Microsoft.NET Framework 4.5 на языке C# / Пер. с англ., 4-е изд. – Изд-во: Питер, 2013 - 896с.
10. Джеффри Рихтер, Мартен ван де Боспурт. WinRT: программирование на C# для профессионалов. Windows Runtime via C#// Пер. с англ. – М.: Вильямс, 2014. – 368 с.
11. Буч Г., Рамбо Д., Якобсон И. Язык UML. Руководство пользователя. 2-е изд.: Пер. с англ. Мухин Н. – М.: ДМК Пресс, 2007.-496с.:ил.

### 15. Інформаційні ресурси

12. База знань Microsoft Developer Network (MSDN) – [Электронный ресурс] – Режим доступа: – <http://msdn.microsoft.com/> (01.09.2017)
13. Офіційний сайт компанії Microsoft щодо технологій WPF та Windows Forms – [Электронный ресурс] – Режим доступа: – <http://window-sclient.net> (01.09.2016)
14. Internet-інститут інформаційних технологій – [Электронный ресурс] – Режим доступа: – [www.intuit.ru](http://www.intuit.ru) (01.09.2017)