

Міністерство освіти і науки України
Національний аерокосмічний університет ім. М. Є. Жуковського
«Харківський авіаційний інститут»

Кафедра інженерії програмного забезпечення (№ 603)

ЗАТВЕРДЖУЮ

Гарант освітньої програми



I.V. Шевченко
(ініціали та прізвище)

« 31 » 08 2021 р.

**РОБОЧА ПРОГРАМА *ОБОВ'ЯЗКОВОЇ*
НАВЧАЛЬНОЇ ДИСЦИПЛІНИ**

«Об'єктно-орієнтоване програмування (Курсова робота)»
(назва навчальної дисципліни)

Галузь знань: 12 «Інформаційні технології»
(шифр і найменування галузі знань)

Спеціальність: 121 «Інженерія програмного забезпечення»
(код та найменування спеціальності)

Освітня програма: «Інженерія програмного забезпечення»
(найменування освітньої програми)

Форма навчання: денна

Рівень вищої освіти: початковий рівень (короткий цикл) вищої освіти

Харків 2021 рік

Розробник: Голуб Н.Г., доц. каф. 603, канд. техн. наук., доц.
(прізвище та ініціали, посада, наукова ступінь та вчене звання)


(підпис)

Робочу програму розглянуто на засіданні кафедри інженерія програмного забезпечення

(назва кафедри)

Протокол № 2 від « 31 » 08 2021 р.

Завідувач кафедри

д-р техн. наук., проф.
(науковий ступінь і вчене звання)


(підпис)

І.Б. Туркін
(ініціали та прізвище)

1. Опис навчальної дисципліни

Найменування показника	Галузь знань, спеціальність, освітня програма, рівень вищої освіти	Характеристика навчальної дисципліни (денна форма навчання)
Кількість кредитів – 2	Галузь знань <u>12 «Інформаційні технології»</u> (шифр і найменування) Спеціальність <u>121 «Інженерія програмного забезпечення»</u> (код і найменування) Освітня програма <u>«Інженерія програмного забезпечення»</u> (найменування) Рівень вищої освіти: <u>початковий рівень (короткий цикл) вищої освіти</u>	Обов’язкова
Кількість модулів – 1		Навчальний рік
Кількість змістовних модулів – 3		2021/2022
Індивідуальне завдання «Спадкування. Абстрактні та поліморфні класи. Динамічний поліморфізм» (назва)		Семестр
Загальна кількість годин – 32/60		<u>4</u> -й
		Лекції*
Кількість тижневих годин для денної форми навчання: аудиторних – 2 самостійної роботи студента – 1,75		<u>0</u> години
		Практичні, семінарські*
		<u>0</u> годин
		Лабораторні*
	<u>32</u> години	
	Самостійна робота	
	<u>28</u> години	
	Вид контролю	
	Модульний контроль, диф. залік	

Співвідношення кількості годин аудиторних занять до самостійної роботи становить: 32/28.

*Аудиторне навантаження може бути зменшене або збільшене на одну годину залежно від розкладу занять.

2. Мета та завдання навчальної дисципліни

Мета вивчення: Закріплення знань методів об'єктно-орієнтованого аналізування, проектування та програмування мовами C++ / C#. Мета досягається за рахунок сполучення таких форм навчання, як практичні (лабораторні) заняття по розробці курсового проекту, а також самостійної роботи студентів.

Завдання: завдяки виконанню курсового проекту опанування студентами навичками об'єктно-орієнтованого проектування мовами C++ / C# (середовище Visual Studio Professional 2019 або вище).

Компетентності, які набуваються:

Загальні компетентності

- ЗК01. Здатність до абстрактного мислення, аналізу та синтезу.

Фахові компетентності

- ФК11. Здатність до алгоритмічного та логічного мислення.

Очікувані результати навчання:

- ПРН3. Знати і застосовувати основні професійні стандарти та інші нормативно-правові документи в галузі інженерії програмного забезпечення.
- ПРН04. Знати і застосовувати відповідні математичні поняття, методи об'єктно-орієнтованого аналізу для розробки програмного забезпечення.
- ПРН05. Знати фундаментальні концепції, парадигми і основні принципи функціонування програмно-апаратних та хмарних засобів інженерії програмного забезпечення.
- ПРН09. Знати і застосовувати методи розробки алгоритмів, конструювання програмного забезпечення та структур даних і знань.

Пререквізити – учбові курси «Основи програмної інженерії», «Основи програмування», «Програмування мовою C#», «Об'єктно-орієнтоване програмування», «Якість програмного забезпечення та тестування», «Аналіз вимог до програмного забезпечення», «Основи організації спільної роботи в ІТ».

Кореквізити – «Міжнародні стандарти і менеджмент проектів програмного забезпечення»/«Економіка ІТ проектів».

3. Зміст навчальної дисципліни

Модуль 1

Змістовний модуль 1. *Вибір варіанту курсового проекту та форми роботи (командна, індивідуальна)*

Тема 1. *Вивчення за літературними або Інтернет джерелами предметну область варіанту завдання. Уточнення постановки задачі, в тому числі для кожного члена команди, зробити опис предметної області (електронний варіант).*

Тема 2. *Перша частина завдання. Попередня повна UML-діаграма основних класів (абстрактного і не менше 2-х похідних класів), зроблена в Visual Studio (або в іншому програмному інструменті). Тестовий приклад (вихідні дані генеруються випадково з урахуванням опису предметної області). Короткий опис класів і методів (фрагменти пояснювальної ЗАПИСКИ).*

Змістовний модуль 2. *Реалізація в курсовому проекті основних принципів ООП, відповідно до обраної мови програмування (C ++, C #)*

Тема 1. *Друга частина завдання. Створення масиву / списку / параметризованої колекції об'єктів цих класів в статичної / динамічної пам'яті. Тестовий приклад (вихідні дані генеруються випадково з урахуванням опису предметної області). Редагування вихідної інформації (додавання, видалення, зміна). Доповнена повна UML-діаграма класів. Додатковий опис класів і методів (фрагменти ЗАПИСКИ).*

Тема 2. *Третя частина завдання. Реалізація статичного / динамічного поліморфізму. Тестовий приклад. Доповнена повна UML-діаграма класів. Опис нових / модифікованих класів і методів (фрагменти ЗАПИСКИ).*

Тема 3. *Виправлення помилок. Файловий ввід-вивід. Тестовий приклад. Остаточна повна UML-діаграма всіх розроблених класів (Visual Studio). Повний опис класів і методів (фрагменти ЗАПИСКИ).*

Змістовний модуль 3. *Підготування до захисту курсового проекту*

Тема 1. *Підготування остаточного варіанту програми та повної пояснювальної записки.*

Тема 2. *Підготування презентації та захист курсового проекту.*

4. Структура навчальної дисципліни

Назва змістовного модуля і тем	Кількість годин				
	Усього	У тому числі			
		л	п	лаб.	с. р.
1	2	3	4	5	6
Модуль 1					
Змістовний модуль 1. Вибір варіанту курсового проекту та форми роботи (командна, індивідуальна)					
Тема 1. Уточнення постановки задачі	8	-	-	4	4
Тема 2. Перша частина завдання. Попередня повна UML-діаграма основних класів	8	-	-	4	4
Модульний контроль	-	-	-	-	-
Разом за змістовним модулем 1	16	-	-	8	8
Змістовний модуль 2. Реалізація в курсовому проекті основних принципів ООП, відповідно до обраної мови програмування (C ++, C #)					
Тема 1. Друга частина завдання. Створення масиву / списку / параметризованої колекції об'єктів цих класів в статичної / динамічної пам'яті.	8	-	-	4	4
Тема 2. Третя частина завдання. Реалізація статичного / динамічного поліморфізму.	8	-	-	4	4
Тема 3. Виправлення помилок. Файловий ввід-вивід.	16	-	-	8	8
Модульний контроль	-	-	-	-	-
Разом за змістовним модулем 2	32			16	16
Змістовний модуль 3. Підготування та захист курсового проекту					
Тема 1. Підготування остаточного варіанту програми та повної пояснювальної записки.	10	-	-	6	4
Тема 2. Підготування презентації та захист курсового проекту	2	-	-	2	-
Модульний контроль	-	-	-	-	-
Разом за змістовним модулем 3	12			8	4

Усього годин	60	-	-	32	28
Індивідуальне завдання	Курсовий проект				
Контрольний захід Захист курсового проекту					
Усього годин	60	-	-	32	28

5. Теми семінарських занять

№ п/п	Назва теми	Кількість годин
1		
2		
	Разом	0

6. Теми практичних занять

№ п/п	Назва теми	Кількість годин
1		
2		
	Разом	0

7. Теми лабораторних занять

№ п/п	Назва теми	Кількість годин
1	<i>Вивчення за літературними або Інтернет джерелами предметну область варіанту завдання. Уточнення постановки задачі, в тому числі для кожного члена команди, зробити опис предметної області (електронний варіант).</i>	4
2	<i>Перша частина завдання. Попередня повна UML-діаграма основних класів (абстрактного і не менше 2-х похідних класів), зроблена в Visual Studio (або в іншому програмному інструменті). Тестовий приклад (вихідні дані генеруються випадково з урахуванням опису предметної області). Короткий опис класів і методів (фрагменти пояснювальної ЗАПИСКИ).</i>	4
3	<i>Друга частина завдання. Створення масиву / списку / параметризованої колекції об'єктів цих класів в статичній / динамічній пам'яті. Тестовий приклад (вихідні дані</i>	4

	<i>генеруються випадково з урахуванням опису предметної області). Редагування вихідної інформації (додавання, видалення, зміна). Доповнена повна UML-діаграма класів. Додатковий опис класів і методів (фрагменти ЗАПИСКИ).</i>	
4	<i>Третя частина завдання. Реалізація статичного / динамічного поліморфізму. Тестовий приклад. Доповнена повна UML-діаграма класів. Опис нових / модифікованих класів і методів (фрагменти ЗАПИСКИ).</i>	4
5	<i>Виправлення помилок. Файловий ввід-вивід. Тестовий приклад. Остаточна повна UML-діаграма всіх розроблених класів (Visual Studio). Повний опис класів і методів (фрагменти ЗАПИСКИ).</i>	8
6	<i>Підготування остаточного варіанту програми та повної пояснювальної записки.</i>	6
7	<i>Підготування презентації та захист курсового проекту.</i>	2
	Разом	32

8. Самостійна робота

№ з/п	Назва теми	Кількість годин
1	<i>Вивчення за літературними або Інтернет джерелами предметну область варіанту завдання. Уточнення постановки задачі, в тому числі для кожного члена команди, зробити опис предметної області (електронний варіант).</i>	4
2	<i>Перша частина завдання. Попередня повна UML-діаграма основних класів (абстрактного і не менше 2-х похідних класів), зроблена в Visual Studio (або в іншому програмному інструменті). Тестовий приклад (вихідні дані генеруються випадково з урахуванням опису предметної області). Короткий опис класів і методів (фрагменти пояснювальної ЗАПИСКИ).</i>	4
3	<i>Друга частина завдання. Створення масиву / списку / параметризованої колекції об'єктів цих класів в статичної / динамічної пам'яті. Тестовий приклад (вихідні дані генеруються випадково з урахуванням опису предметної області). Редагування вихідної інформації (додавання, видалення, зміна). Доповнена повна UML-діаграма класів. Додатковий опис класів і методів (фрагменти ЗАПИСКИ).</i>	4
4	<i>Третя частина завдання. Реалізація статичного / динамічного поліморфізму. Тестовий приклад. Доповнена</i>	4

	<i>повна UML-діаграма класів. Опис нових / модифікованих класів і методів (фрагменти ЗАПИСКИ).</i>	
5	<i>Виправлення помилок. Файловий ввід-вивід. Тестовий приклад. Остаточна повна UML-діаграма всіх розроблених класів (Visual Studio). Повний опис класів і методів (фрагменти ЗАПИСКИ).</i>	8
6	<i>Підготування остаточного варіанту програми та повної пояснювальної записки.</i>	4
	Разом	28

9. Індивідуальне завдання

Виконання **курсової роботи** на тему «Спадкування. Абстрактні та поліморфні класи. Динамічний поліморфізм».

10. Методи навчання

За джерелами придбання знань – словесні: лекція (вступна, традиційна, проблемна, з помилками), бесіда (евристична), диспут, дискусія, робота з друкованими та інтернет-джерелами; наочні: ілюстрування у вигляді слайдів презентації MS PowerPoint та рисунків MS Visio, відео-файлів; практичні: вправа, лабораторна робота, курсовий проект/робота.

За характером пізнавальної діяльності тих, хто навчається – інформаційно-репродуктивний, репродуктивний, проблемне викладання, частково-пошуковий.

За логікою пізнання – індуктивний, дедуктивний, аналогій, вивідних знань.

Методи перевірки й оцінки знань, умінь, навичок: спостереження, усне опитування, контрольні роботи, програмований контроль, тестування (традиційне та машинне).

11. Методи контролю

Виконання та захист етапів курсової роботи на **лабораторних заняттях** у формі завантаження в середовище mentor.khai.edu фрагментів курсового проекту.

Форма підсумкового контролю успішності навчання: диф.залік у 4 семестрі.

12. Критерії оцінювання та розподіл балів, які отримують здобувачі

Розподіл балів, які отримують студенти за виконання курсової роботи (проекту)

Пояснювальна записка	Ілюстративна частина	Захист роботи	Сума
до _90_	до _0_	до _10_	100

Критерії оцінювання роботи здобувача протягом семестру

Задовільно (60-74). Показати мінімум знань та умінь по темі курсової роботи. Уявляти історію створення C++/C#, розвиток та синтаксичні об'єктно-орієнтовані особливості цих мов. Розуміти властивості різних середовищ розробки застосунків мовою C++/C#.

Добре (75-89). Твердо знати мінімум по темі курсової роботи. Досконало знати історію створення C++/C#; розуміти основні поняття об'єктно-орієнтованого програмування: інкапсуляція, поліморфізм (а також ad hoc поліморфізм), наслідування; особливості мови C++/C# при розробці об'єктно-орієнтованих застосунків; абстрактні та поліморфні класи, статичний та динамічний поліморфізм, особливості використання колекцій або об'єктів STL.

Відмінно (90-100). Здати всі контрольні точки з оцінкою «відмінно». Досконально знати всі теми учбового курсу та уміти практично застосовувати їх.

Шкала оцінювання: бальна і традиційна

Сума балів	Оцінка за традиційною шкалою	
	Іспит, диференційований залік	Залік
90 – 100	Відмінно	Зараховано
75 – 89	Добре	
60 – 74	Задовільно	
0 – 59	Незадовільно	Не зараховано

13. Методичне забезпечення

1. Навчально-методичне забезпечення дисципліни "Об'єктно-орієнтоване програмування (курсний проект)" для бакалаврів / М-во освіти і науки України, Нац. аерокосм. ун-т ім. М. Є. Жуковського "Харків. авіац. ін-т", Каф. інженерії програм. забезп. (№ 603) ; розроб. Н. Г. Голуб. - Харків. - Нац. аерокосм. ун-т ім.

- М. Є. Жуковського "Харків. авіац. ін-т", 2019. - 26 с. - http://library.khai.edu/library/fulltexts/doc/_1003Obyektno_Oriyentovane.pdf
2. Розроблений лекційний курс по Об'єктно-орієнтованому програмуванню з комплексом презентацій Power Point, а також відео лекцій, прикладів, тощо <https://mentor.khai.edu/course/view.php?id=5799>
 3. Розроблені Методичні вказівки по варіантам курсового проекту <https://mentor.khai.edu/course/view.php?id=5812>
 4. Розроблений Приклад оформлення пояснювальної записки до курсового проекту <https://mentor.khai.edu/course/view.php?id=5812>
 5. Розроблені методичні вказівки для виконання етапів курсового проекту, а також система їх приймання та оцінювання <https://mentor.khai.edu/course/view.php?id=5812>
 6. Дібрані матеріали для самостійної роботи студентів <https://mentor.khai.edu/course/view.php?id=5812>

14. Рекомендована література

Базова

- 1 Голуб Н.Г. Мова С++. Об'єктно-орієнтоване програмування. – К.: Діасофт, 2007-2021 : [Електронний ресурс]. – Режим доступу: <https://mentor.khai.edu/course/view.php?id=5799>
- 2 Основи програмування CS50 2019: [Електронний ресурс]. – Режим доступу: https://courses.prometheus.org.ua/courses/course-v1:Prometheus+CS50+2019_T1/about
- 3 Пекарський, Б.Г. Основи програмування [Текст] : навч. посіб. / Б.Г. Пекарський. – К.: Кондор, 2018. - 364 с.
- 4 Васильєв О. Програмування на С++ в прикладах і задачах [Текст] : навч. посіб. /О. Васильєв. – К.: Ліра-К, 2017. - 382 с.
- 5 Мова програмування С++: [Електронний ресурс]. – Режим доступу: <http://www.znannya.org/?view=group:Cpp>
- 6 Мова програмування С#: [Електронний ресурс]. – Режим доступу: <http://www.znannya.org/?view=csharp>
- 7 Основи програмування на С#: [Електронний ресурс]. – Режим доступу: https://courses.prometheus.org.ua/courses/Microsoft/CS201/2016_T1/about

Допоміжна

- 1 Голуб Н.Г. Особенности реализации перегрузки операторных функций для арифметических, логических операций и сравнений: С# vs С++ // «ІНТЕРНЕТ-ОСВІТА-НАУКА-2016», Десята міжнародна науково-практична конференція ІОН-2016, 11-14 жовтня, 2016 : Збірник праць. –

- Вінниця : ВНТУ, 2016 – с. 219-221. - <http://ir.lib.vntu.edu.ua/bitstream/handle/123456789/13373/219-221.pdf>
- 2 Голубь Н.Г. Особенности использования деструкторов при программировании управляемого и неуправляемого кода в объектно-ориентированном программировании на примере алгоритмических языков С++ и С# // Всеукраїнська науково-технічна конференція «Інтегровані комп'ютерні технології в машинобудуванні ІКТМ'2018». Збірник матеріалів конференції, том 3. – Харків, “ХАІ”, 2018. – с. 106. - <http://ies.vntu.edu.ua/reports/program/WORK-IES-2018.pdf>
 - 3 Голубь Н.Г. Особенности работы в языках С#/С++ с указателями и ссылками в объектно-ориентированном программировании // «ІНТЕРНЕТ-ОСВІТА-НАУКА-2018», Одинадцята міжнародна науково-практична конференція ІОН-2018, 22-25 травня, 2018 : Збірник праць. – Вінниця : ВНТУ, 2018 – с. 258-260. - <https://press.vntu.edu.ua/index.php/vntu/catalog/book/468>.
 - 4 Дейл Н. Программирование на С++ [Текст] : пер. с англ. / Н. Дейл, М. Хедингтон, Ч. Уимз. М.: Вильямс, 2016 — 288 с.
 - 5 Липпман С. Язык программирования С++. Базовый курс [Текст] : пер. с англ. / С. Липпман, Ж. Лажойе, Б. Му. М.: Вильямс, 2017 — 1120 с.
 - 6 Шилдт Г. С++. Базовый курс, 3-е изд. [Текст] : пер. с англ. / Г. Шилдт. К.: Диалектика, 2020 — 624 с.
 - 7 Шилдт Г. С# 4.0: полное руководство, 3-е изд. [Текст] : пер. с англ. / Г. Шилдт. М.: Вильямс, 2019 — 1056 с.
 - 8 Страуструп Б. Программирование. Принципы и практика с использованием С++ [Текст] : пер. с англ. / Б. Страуструп. М.: Вильямс, 2016 — 1328 с.
 - 9 Прата С. Язык программирования С++. Лекции и упражнения, 6-е изд. [Текст] : пер. с англ. / С. Прата. М.: Вильямс, 2019 — 1248 с.
 - 10 Албахари Дж. С# 7.0. Справочник. Полное описание языка, 7-е изд. [Текст] : пер. с англ. / Джозеф Албахари, Бен Албахари. К.: Диалектика, 2019 — 1024 с.
 - 11 Скит Дж. С# для профессионалов: тонкости программирования. 3-е изд. [Текст] : пер. с англ. / Джон Скит. К.: Диалектика, 2019 — 608 с.
 - 12 Троелсен Э. Язык программирования С# 7 и платформы .NET и .NET Core, 8-е изд. [Текст] : пер. с англ. / Эндрю Троелсен, Филипп Джепикс. К.: Диалектика, 2019 — 1328 с.

15. Інформаційні ресурси

1. С# documentation: [Електронний ресурс]. – Режим доступу: <https://docs.microsoft.com/en-us/dotnet/csharp/>
2. С# reference: [Електронний ресурс]. – Режим доступу: <https://docs.microsoft.com/en-us/dotnet/csharp/language-reference/>
3. С++ Language Reference: [Електронний ресурс]. – Режим доступу: <https://docs.microsoft.com/en-us/cpp/cpp/cpp-language-reference?view=msvc-160&viewFallbackFrom=vs-2019>

4. C/C++ language and standard libraries reference: [Электронный ресурс]. – Режим доступа: <https://docs.microsoft.com/en-us/cpp/cpp/c-cpp-language-and-standard-libraries?view=msvc-160>