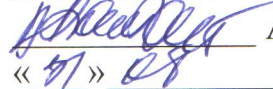


Міністерство освіти і науки України
Київський національний торговельно-економічний університет
Харківський торговельно-економічний інститут КНТЕУ

Факультет економіки та управління
Кафедра обліку та інформаційних технологій у бізнесі

ЗАТВЕРДЖУЮ

Заступник директора з
науково-педагогічної роботи
ХТЕІ КНТЕУ

 А. В. Катаєв
« 27 » 08 2018 р.

РОБОЧА ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

Об'єктно-орієнтоване програмування

(повна назва навчальної дисципліни)

для підготовки
студентів освітнього
ступеня

бакалавр
(бакалавр чи магістр)

року набору 2018

галузі знань

12 Інформаційні технології
(шифр і назва галузі знань)

Спеціальності

126 Інформаційні системи та технології
(шифр і найменування спеціальності)

напряму підготовки

(шифр і найменування напряму підготовки)

спеціалізації

«Інформаційні технології у бізнесі»
(найменування спеціалізації)

Харків, 2018 рік

Розробник Галуза Олексій Анатолійович

професор кафедри обліку та інформаційних технологій у бізнесі

доктор фізико-математичних наук, професор

Програму обговорено та схвалено на засіданні кафедри
обліку та інформаційних технологій у бізнесі

(назва кафедри)

протокол № 1 від 31.08.2018 р.

Зав. кафедри



(підпис)

С. О. Кузнецова

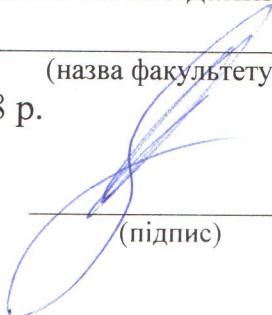
(ініціали та прізвище)

Програму розглянуто та схвалено на засіданні методичної комісії факультету
економіки та управління

(назва факультету)

протокол № 1 від 31.08.2018 р.

Голова методичної комісії



(підпис)

О. М. Гавриш

(ініціали та прізвище)

1. ОПИС НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

Робоча програма навчальної дисципліни «Об'єктно-орієнтоване програмування» розроблена відповідно до освітньо-професійної програми підготовки студентів спеціальності 126 Інформаційні системи та технології (спеціалізації «Інформаційні технології у бізнесі») галузі знань 12 Інформаційні технології.

Метою викладання навчальної дисципліни є вивчення основних понять та концепцій об'єктно-орієнтованого підходу до створення програмного забезпечення, принципів побудови об'єктно-орієнтованих програм; набуття практичних навичок створення об'єктно-орієнтованих програм та використання стандартних об'єктно-орієнтованих бібліотек.

Предметом вивчення дисципліни є об'єктно-орієнтовані технології розробки програм, система програмування мовою C++.

Міждисциплінарні зв'язки Дисципліна «Об'єктно-орієнтоване програмування» відповідно до структурно-логічної схеми викладення дисциплін, передбачених навчальним планом, викладається на першому курсі у першому семестрі. Знання, що студенти отримають при вивченні дисципліни «Об'єктно-орієнтоване програмування», є основою для подальшого засвоєння професійних навчальних дисциплін, таких як «Прикладне програмування в бізнесі», «Управління ІТ проектами», а також при проходженні практик, написанні курсових та випускної кваліфікаційної роботи.

НАВЧАЛЬНИЙ ВІДДІЛ
ХАРКІВСЬКИЙ ТОРГОВЕЛЬНО-
ЕКОНОМІЧНИЙ ІНСТИТУТ КНТЕУ

2. ЗАПЛАНОВАНІ РЕЗУЛЬТАТИ НАВЧАННЯ

Навчальна дисципліна забезпечує набуття студентами:

фахових компетентностей:

ФК 2. Здатність використання принципів структурного програмування, сучасних процедурно-орієнтованих та об'єктно-орієнтованих мов програмування, основних структур даних і вміння їх застосовувати під час програмної реалізації алгоритмів професійних завдань;

ФК 4. Здатність використання концепцій накопичення даних, сучасних теорій організації баз даних та знань, їх оперативної аналітичної обробки та інтелектуального аналізу, методів і технологій їх розробки, уміння проектувати логічні та фізичні моделі баз даних і запити до них;

ФК 6. Здатність використання принципів, методів і алгоритмів комп'ютерної графіки, уміння застосовувати їх під час розробки графічних інтерфейсів взаємодії людини з комп'ютером;

програмних результатів навчання:

ПРН 15. Вміти застосовувати методи компонентної розробки програмного забезпечення, виділяючи інтерфейси і реалізації та взаємодію між модулями, підсистемами і компонентами;

ПРН 16. Знати, розуміти і застосовувати відповідні математичні поняття, методи доменного, системного і об'єктно-орієнтованого аналізів та математичного моделювання для розробки програмного забезпечення;

ПРН 19. Застосовувати на практиці інструментальні програмні засоби об'єктно-орієнтованого, системного та доменного аналізу, проектування, тестування, візуалізації, вимірювань та документування програмного забезпечення;

ПРН 20. Мотивовано обирати мови програмування для розв'язання завдань створення і супроводження інформаційних систем;

ПРН 21. Аналізувати, оцінювати і вибирати інструментальні та обчислювальні засоби технології, алгоритмічні і програмні рішення для розв'язання завдань інформаційних систем і технологій;

ПРН 22. Знати і застосовувати методи розробки алгоритмів, конструювання програмного забезпечення та структур даних і знань;

ПРН 23. Знати та вміти застосовувати технології та методи проектування та програмування.

3. ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

Тема 1. Вступ до об'єктно-орієнтованого програмування

Класи. Особливості описання полів класу. Приватні, відкриті та захищені члени класів. Особливості описання локальних класів. Описання об'єктів класу. Доступ до елементів класу. Поняття інкапсуляції. Показчик `this`.

Тема 2. Конструктори та деструктори

Особливості конструктора як методу класу. Конструктор за замовчуванням. Конструктор копіювання. Деструктори. Особливості деструктора як методу класу. Виклик деструкторів.

Тема 3. Статичні елементи класу. Дружні функції та класи

Проблема доступу до загального ресурсу. Особливості статичних елементів. Проблема відсутності доступу до закритих членів класу. Дружні функції. Особливості дружніх функцій. Дружні класи.

Тема 4. Перевантаження операцій 1

Операторні функції. Бінарні та унарні оператори. Зумовлений сенс операторів. Оператори та типи, що визначено користувачем. Оператори у просторах імен.

Тема 5. Перевантаження операцій 2

Оператори перетворень типів. Вирішення неоднозначності. Операція присвоєння. Явні конструктори. Індексція. Розіменування. Інкремент та декремент.

Тема 6. Наслідування

Похідні класи. Особливості режиму доступу до членів класів. Перевизначення елементів базового класу. Виклик конструктора базового класу. Порядок виклику конструкторів. Виклик деструктора базового класу. Віртуальні методи. Правила визначення та використання віртуальних методів. Чисто віртуальні методи.

Тема 7. Множинне наслідування

Ієрархія класів. Розв'язання неоднозначностей. Базові класи, що повторюються. Віртуальні базові класи. Використання множинного наслідування. Управління доступом. Захищені члени класів. Доступ до елементів базового класу.

Тема 8. Шаблони функцій та класів

Простий шаблон вектора. Визначення шаблону. Інстанціювання. Параметри шаблонів. Еквівалентність типів. Перевірка типів. Шаблони функцій. Перевантаження шаблонів функцій. Параметри шаблонів за замовченням.

Тема 9. Простори імен

Розділення програми на модулі та інтерфейси. Простори імен. Імена з кваліфікаторами. Об'ява using. Множинні інтерфейси. Вирішення конфліктів імен. Пошук імен. Псевдоніми імен та просторів імен. Об'єднання просторів імен.

Тема 10. Обробка виключних ситуацій 1

Проблема обробки помилок та підходи до її вирішення. Традиційна обробка помилок. Механізм виключень (виключних ситуацій). Синтаксис виключень. Оператор throw. Блок catch. Вибір обробника виключення. Оператор throw без операнду.

Тема 11. Обробка виключних ситуацій 2

Список виключень функції. Обробка непередбачених виключень. Перекидання необроблених виключень. Класи та ієрархії виключень. Приклади стандартних класів виключень (класи xmsg, bad_alloc). Інформація про виключення. Системні виключення. Механізм обробки виключень.

Тема 12. Бібліотека STL. Потоки

Класи потокового вводу та виводу та їх ієрархія. Стандартні потоки (cin, cout, cerr, clog). Отримання даних з потоку та передача даних у потік. Операції вводу та виводу для вбудованих типів. Перевантаження операцій вводу та виводу для типів користувача. Форматований ввід та вивід; методи, що форматують. Маніпулятори та флаги. Стан потоку. Файлові потоки.

Тема 13. Бібліотека STL. Контейнерні класи та рядки

Поняття контейнеру. Вектори. Перевірка допустимості діапазону. Стандартні контейнери. Ітератори. Список. Стек. Черка. Дека. Асоціативні контейнери. Майже контейнери.

Тема 14. Бібліотека STL. Алгоритми

Огляд алгоритмів стандартної бібліотеки. Послідовності та контейнери. Об'єкти-функції. Предикати. Алгоритми, що не модифікують послідовності (алгоритми `for_each`, `find`, `count`, рівність та неспівпадання, пошук). Алгоритми, що модифікують послідовності.

4. СТРУКТУРА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

4.1. Структура навчальної дисципліни за формами навчання

Тема дисципліни	Обсяг у годинах																							
	денна форма												заочна форма											
	повна						скорочена						повна						скорочена					
	усього	у тому числі					усього	у тому числі					усього	у тому числі					усього	у тому числі				
		Л	СЗ	ПЗ	ЛЗ	СРС		Л	СЗ	ПЗ	ЛЗ	СРС		Л	СЗ	ПЗ	ЛЗ	СРС		Л	СЗ	ПЗ	ЛЗ	СРС
Тема 1. Вступ до об'єктно-орієнтованого програмування.							12	2		4		6							13	1		1		11
Тема 2. Конструктори та деструктори.							10	2		2		6							13	1		1		11
Тема 3. Статичні елементи класу. Дружні функції та класи.							12	2		4		6							11					11
Тема 4. Перевантаження операцій I.							10	2		2		6							13	1		1		11
Тема 5. Перевантаження операцій II.							10	2		2		6							13	1		1		11
Тема 6. Наслідування.							10	2		2		6							13	1		1		11
Тема 7. Множинне наслідування.							12	2		4		6							13	1		1		11
Тема 8. Шаблони функцій та класів.							11	2		4		5							13	1		1		11
Тема 9. Простори імен.							9	2		2		5							11					11
Тема 10. Обробка виключних ситуацій 1.							9	2		2		5							13	1		1		11
Тема 11. Обробка виключних ситуацій 2.							9	2		2		5							13	1		1		11
Тема 12. Бібліотека STL. Потoki.							12	2		4		6							13	1		1		11
Тема 13. Бібліотека STL. Контейнерні класи та рядки.							12	2		4		6							14	1		1		12
Тема 14. Бібліотека STL. Алгоритми.							12	2		4		6							14	1		1		12
Курсова робота (проект)							30																	
Усього годин / кредитів ECTS							180/ 6	28		42		80							180/6	12		12		156

При формуванні структури навчальної дисципліни можливим є: 1) поділ дисципліни на розділи (частини); 2) планування окремих занять (не на лекціях) для проведення певних контрольних заходів, підбиття кількісних підсумків результативності навчання студентів.

4.2. Обсяги та структура навчальної дисципліни за навчальними роками

Форма навчання	Вид навчальних занять	Навчальні роки					
		2018/2019		2019/2020		2020/2021	
		осінь	весна	осінь	весна	осінь	весна
Денна повна	Лекційні заняття						
	Семінарські заняття						
	Практичні заняття						
	Лабораторні заняття						
	Курсова робота (проект)						
	Самостійна робота студентів						
	Усього годин						
Денна скорочена	Лекційні заняття	28					
	Семінарські заняття						
	Практичні заняття	42					
	Лабораторні заняття						
	Курсова робота (проект)	30					
	Самостійна робота студентів	80					
	Усього годин	180					
Заочна повна	Лекційні заняття						
	Семінарські заняття						
	Практичні заняття						
	Лабораторні заняття						
	Курсова робота (проект)						
	Самостійна робота студентів						
	Усього годин						
Заочна скорочена	Лекційні заняття	10	2				
	Семінарські заняття						
	Практичні заняття	8	4				
	Лабораторні заняття						
	Курсова робота (проект)						
	Самостійна робота студентів	102	54				
	Усього годин	120	60				

5. ТЕМИ ЛЕКЦІЙНИХ, СЕМІНАРСЬКИХ (ПРАКТИЧНИХ) І ЛАБОРАТОРНИХ ЗАНЯТЬ

5.1. Теми лекційних занять

Тема дисципліни	Тема лекції	Обсяг у годинах			
		денна		заочна	
		повна	скорочена	повна	скорочена
Тема 1. Вступ до об'єктно-орієнтованого програмування.	Вступ до об'єктно-орієнтованого програмування.		2		1
Тема 2. Конструктори та деструктори.	Конструктори та деструктори.		2		1
Тема 3. Статичні елементи класу. Дружні функції та класи.	Статичні елементи класу. Дружні функції та класи.		2		
Тема 4. Перевантаження операцій 1.	Перевантаження операцій 1.		2		1
Тема 5. Перевантаження операцій 2.	Перевантаження операцій 2.		2		1
Тема 6. Наслідування.	Наслідування.		2		1
Тема 7. Множинне наслідування.	Множинне наслідування.		2		1
Тема 8. Шаблони функцій та класів.	Шаблони функцій та класів.		2		1
Тема 9. Простори імен.	Простори імен.		2		
Тема 10. Обробка виключних ситуацій 1.	Обробка виключних ситуацій 1.		2		1
Тема 11. Обробка виключних ситуацій 2.	Обробка виключних ситуацій 2.		2		1
Тема 12. Бібліотека STL. Потоки.	Бібліотека STL. Потоки.		2		1
Тема 13. Бібліотека STL. Контейнерні класи та рядки.	Бібліотека STL. Контейнерні класи та рядки.		2		1
Тема 14. Бібліотека STL. Алгоритми.	Бібліотека STL. Алгоритми.		2		1
Усього			28		12

5.2. Теми семінарських занять

Тема дисципліни	Тема семінарського заняття	Обсяг у годинах	
		денна форма	заочна форма
Усього		-	-

5.3. Теми практичних занять

Тема дисципліни	Тема лекції	Обсяг у годинах			
		денна		заочна	
		повна	скорочена	повна	скорочена
Тема 1. Вступ до об'єктно-орієнтованого програмування.	1. Знайомство з середовищем розробки 2. Створення найпростіших класів		4		1
Тема 2. Конструктори та деструктори.	Конструктори та деструктори.		2		1
Тема 3. Статичні елементи класу. Дружні функції та класи.	1. Статичні елементи класу. 2. Дружні функції та класи.		4		
Тема 4. Перевантаження операцій 1.	Перевантаження унарних та бінарних операцій.		2		1
Тема 5. Перевантаження операцій 2.	Перевантаження операцій new, delete, виклик функції, індексація.		2		1
Тема 6. Наслідування.	Наслідування.		2		1
Тема 7. Множинне наслідування.	1. Множинне наслідування. 2. Віртуальні функції		4		1
Тема 8. Шаблони функцій та класів.	1. Шаблони функцій 2. Шаблони класів		4		1
Тема 9. Простори імен.	Простори імен.		2		
Тема 10. Обробка виключних ситуацій 1.	Обробка виключних ситуацій		2		1
Тема 11. Обробка виключних ситуацій 2.	Класи та ієрархії виключень.		2		1
Тема 12. Бібліотека STL. Потoki.	1. Форматований ввід та вивід до потоків. 2. Перевантаження операцій вводу та виводу для типів користувача.		4		1
Тема 13. Бібліотека	1. Контейнерні класи		4		1

STL. Контейнерні класи та рядки.	2. Клас String				
Тема 14. Бібліотека STL. Алгоритми.	1. Алгоритми, що не модифікують послідовності 2. Алгоритми, що модифікують послідовності.		4		1
Усього			42		12

6. ЗАВДАННЯ ДЛЯ САМОСТІЙНОЇ РОБОТИ

Організація самостійної роботи студентів регламентується наступними нормативними документами:

- Положенням про самостійну роботу студентів Харківського торговельно-економічного-інституту КНТЕУ;
- Положенням про організацію освітнього процесу у Харківському торговельно-економічному інституті КНТЕУ.

Основні види самостійної роботи, які запропоновані студентам:

1. Вивчення лекційного матеріалу.
2. Робота з опрацювання та вивчення рекомендованої літератури.
3. Вивчення основних термінів та понять рекомендованої літератури.
4. Підготовка до практичних занять, дискусій, індивідуальної роботи, роботи в малих групах, тестування.
5. Контрольна перевірка кожним слухачем особистих знань, запитання для самоконтролю, вирішення задач.

Питання навчальної дисципліни, які пропонуються для самостійного вивчення:

- | | |
|----------|---|
| Тема 1. | Основні типи даних. Представлення цілих та дійсних чисел у пам'яті комп'ютера. Базові алгоритмічні конструкції. |
| Тема 2. | Локальні змінні. Локальна статична пам'ять. Нелокальна пам'ять. |
| Тема 3. | Класи vector та string. |
| Тема 4. | Механізм перевантаження функцій. |
| Тема 5. | Оператори перетворень типів. |
| Тема 6. | Механізм пізнього зв'язування. Абстрактні класи. |
| Тема 7. | Інформація про тип на етапі виконання. Динамічне приведення типів. |
| Тема 8. | Спеціалізація шаблонів. Наслідування та шаблони. |
| Тема 9. | Простори імен та старий код. |
| Тема 10. | Виключення, пов'язані з керування ресурсами. |
| Тема 11. | Специфікації виключних ситуацій. |
| Тема 12. | Файлові та рядкові потоки. Буферизація. Національні особливості. |
| Тема 13. | Створення нових контейнерних класів. |
| Тема 14. | Алгоритми сортування послідовностей. Алгоритми генерацію перестановок. |

7. ІНДИВІДУАЛЬНІ ЗАВДАННЯ, ПЕРЕДБАЧЕНІ НАВЧАЛЬНИМ ПЛАНОМ

Не передбачено навчальним планом.

8. СХЕМА НАРАХУВАННЯ БАЛІВ

8.1. Схема нарахування балів для студентів денної форми навчання

Навчальні роки	Поточний контроль (максимум 60 балів, мінімум 34)														Підсумковий контроль	Сума
	усього	у тому числі за темами														
		Тема 1	Тема 2	Тема 3	Тема 4	Тема 5	Тема 6	Тема 7	Тема 8	Тема 9	Тема 10	Тема 11	Тема 12	Тема 13	Тема 14	
2018/2019	60	4	4	4	4	4	4	4	4	4	5	5	5	5	40	100

8.2. Схема нарахування балів для студентів заочної форми навчання

Навчальні роки	Поточний контроль (максимум 60 балів, мінімум 34)			Підсумковий контроль	Сума
	усього	у тому числі за видами активності			
		вид 1	вид 2		
2018/2019	60	50	10	40	100

Види активності студента з навчальної дисципліни протягом семестру:

1. Самостійне опрацювання тестів відповідно до тем курсу
2. Відвідування аудиторних занять

9. МЕТОДИ КОНТРОЛЮ

Для визначення рівня засвоювання студентами навчального матеріалу використовуються такі форми та методи оцінювання:

Поточний контроль:

- для студентів денної форми навчання: оцінювання роботи на семінарських заняттях, поточне тестування в системі дистанційного навчання ХТЕІ КНТЕУ, оцінювання відвідування аудиторних занять, виконання аудиторної контрольної роботи;

- для студентів заочної форми навчання: тестування в системі дистанційного навчання ХТЕІ КНТЕУ.

Умовою допуску до підсумкового контролю є виконання програми дисципліни (відпрацювання всіх практичних занять) і отримання оцінки за виконання завдань поточного контролю не менше 34 балів.

Підсумковий контроль:

- для студентів денної форми навчання: письмовий екзамен;
- для студентів заочної форми навчання: письмовий екзамен.

Організація та проведення контрольних заходів регламентується наступними нормативними документами:

- Положенням про оцінювання результатів навчання студентів;
- Положенням про організацію освітнього процесу у Харківському торговельно-економічному інституті КНТЕУ.

10. РЕКОМЕНДОВАНА ЛІТЕРАТУРА (ОСНОВНА, ДОПОМІЖНА)

10.1. Основна література

1. Дэвис С. С++ для чайников / С. Дэвис. – Киев : Диалектика, 2018. – 400 с.
2. Липпман С. Язык программирования С++. Базовый курс / С. Липпман, Ж. Лажойе, Б. Му. – Киев: Вильямс, 2017. – 1120 с.

10.2. Допоміжна література

1. Уоррен Г. Алгоритмические трюки для программистов / Г. Уоррен. – Киев : Вильямс, 2017. – 512 с.
2. Кормен Т. Алгоритмы: построение и анализ / Т. Кормен, Ч. Лейзерсон, Р. Ривест. – Киев: Диалектика, 2014. – 1328 с.

11. Інформаційні ресурси в Інтернеті

1. Введение в программирование (С++) [Електронний ресурс]: Освітній ресурс // Stepik. – Режим доступу до сайту: <https://stepik.org>. – Станом на 30.08.18 р. – Назва з екрана.