Міністерство освіти і науки України

Харківський національний університет імені В.Н. Каразіна

Кафедра інформаційних технологій в фізико-енергетичних системах

“ЗАТВЕРДЖУЮ”

Директор
навчально-наукового інституту
комп’ютерної фізики та енергетики

_____ Ірина ГАРЯЧЕВСЬКА

“ 28 ” серпня 2024 р.

РОБОЧА ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

ОБ’ЄКТНО-ОРІЄНТОВАНЕ ПРОГРАМУВАННЯ. ЧАСТИНА 2

рівень вищої освіти	перший (бакалаврський) рівень
галузь знань	10 Природничі науки
спеціальність	105 Прикладна фізика та наноматеріали
освітня програма	Прикладна фізика енергетичних систем
	Прикладна фізика нетрадиційної енергетики
	Комп’ютерна фізика
вид дисципліни	обов’язкова
ННІ	комп’ютерної фізики та енергетики

2024 / 2025 навчальний рік

Програму рекомендовано до затвердження Вченою радою навчально-наукового інституту комп'ютерної фізики та енергетики

" 28 " серпня 2024 року, протокол № 8/24

РОЗРОБНИКИ ПРОГРАМИ:

Протектор Денис Олегович, PhD, доцент кафедри інформаційних технологій в фізико-енергетичних системах.

Програму схвалено на засіданні кафедри інформаційних технологій в фізико-енергетичних системах

Протокол від " 28 " серпня 2024 року № 8/24

Завідувач кафедри інформаційних технологій в фізико-енергетичних системах

_____ Руслан СУХОВ
(підпис)

Програму погоджено з гарантом освітньо-професійної програми (керівником проектної групи) Комп'ютерна фізика

Гарант освітньо-професійної програми (керівник проектної групи)

_____ Світлана РОГОВА
(підпис)

Програму погоджено з гарантом освітньо-професійної програми (керівником проектної групи) Прикладна фізика енергетичних систем

Гарант освітньо-професійної програми (керівник проектної групи)

_____ Денис ПРОТЕКТОР
(підпис)

Програму погоджено з гарантом освітньо-професійної програми (керівником проектної групи) Прикладна фізика нетрадиційної енергетики

Гарант освітньо-професійної програми (керівник проектної групи)

_____ Ілля МАРУЩЕНКО
(підпис)

Протокол від " 28 " серпня 2024 року № 8/24

Голова методичної комісії навчально-наукового інституту комп'ютерної фізики та енергетики

_____ Денис ПРОТЕКТОР
(підпис)

ВСТУП

Програма навчальної дисципліни “Об’єктно-орієнтоване програмування. Частина 2” складена відповідно до освітньо-професійної (освітньо-наукової) програми підготовки бакалаврів

спеціальність: 105 Прикладна фізика та наноматеріали

освітня програма: «Прикладна фізика енергетичних систем»
«Прикладна фізика нетрадиційної енергетики»
«Комп’ютерна фізика»

1. Опис навчальної дисципліни

1.1. Мета викладання навчальної дисципліни

Метою курсу «Об’єктно-орієнтоване програмування. Частина 2» є вивчення сучасної теорії об’єктно орієнтованого програмування з використанням актуальних інтегрованих серед розробки програмних продуктів та набуття практичних навичок програмування.

1.2. Основні завдання вивчення дисципліни

Основними завданнями вивчення дисципліни є набуття студентами знань і навичок програмування на мові C++, необхідних для автоматизованої обробки великих об’ємів даних та комп’ютерного моделювання.

1.3. Кількість кредитів 4

1.4. Загальна кількість годин 120

1.5. Характеристика навчальної дисципліни	
Обов’язкова	
Денна форма навчання	Заочна (дистанційна) форма навчання
Рік підготовки	
3-й	-й
Семестр	
5-й	-й
Лекції	
16 год.	год.
Практичні, семінарські заняття	
год.	год.
Лабораторні заняття	
32 год.	год.
Самостійна робота	
72 год.	год.
у тому числі індивідуальні завдання	
год.	

1.6. Заплановані результати навчання

Згідно з вимогами стандарту вищої освіти за спеціальністю 105 «Прикладна фізика та наноматеріали» галузі знань 10 «Природничі науки» для першого (бакалаврського) рівня вищої освіти студенти повинні досягти таких результатів навчання:

В результаті вивчення навчальної дисципліни студент повинен знати:

- теоретичні засади об’єктно-орієнтованого програмування на мові С++.

В результаті вивчення навчальної дисципліни студент повинен вміти:

- застосовувати парадигму об’єктно-орієнтованого програмування на мові С++ для автоматизованої обробки великих об’ємів даних та комп’ютерного моделювання.
- застосовувати ефективні технології, інструменти та методи експериментального дослідження властивостей речовин і матеріалів, включаючи наноматеріали, при розв’язанні практичних проблем прикладної фізики.
- застосовувати фізичні, математичні та комп'ютерні моделі для дослідження фізичних явищ, розробки приладів і наукоємних технологій.
- вибирати ефективні методи та інструментальні засоби проведення досліджень у галузі прикладної фізики.
- відшуковувати необхідну науково-технічну інформацію в науковій літературі, електронних базах, інших джерелах, оцінювати надійність та релевантність інформації.
- класифікувати, аналізувати та інтерпретувати науковотехнічну інформацію в галузі прикладної фізики.

2. Тематичний план навчальної дисципліни

Розділ 2. Практичне використання концепцій об’єктно орієнтованого програмування.

Тема 8. Віртуальні функції. Поняття абстрактного класу та чисті віртуальні функції. Віртуальні деструктори. Множинне спадкування, його види та правила застосування.

Тема 9. Перевантаження операторів +, -, /, *, =, +=, -=, *=, /=, [ ], ( ), ->, *, постфіксного і префіксного ++ та --.

Тема 10. Перевантаження операторів потокового вводу/виводу. Форматування виводу. Обробка виключень.

Тема 11. Ієрархія потокових класів. Класи istream, ostream. Аналіз та управління станом потока.

Тема 12. Стандартні консольні потоки вводу/виводу. Ввод даних з використанням cin. Методи об’єкта cin. Ввод одного символу та рядка. Вивід даних з використанням cout. Методи об’єкта cout.

Тема 13. Використання файлів для вводу/виводу даних. Класи ofstream та ifstream. Бінарні та текстові файли.

Тема 14. Поняття шаблону. Шаблонні функції. Шаблонні класи. Статичні члени класу та шаблони. Стандартна бібліотека шаблонів. Контейнерні класи. Ітератори. Послідовні контейнери: vector, list, stack, queue. Асоціативні контейнери: map, multimap, set, multiset.

Тема 15. Класи алгоритмів. Алгоритми, що не змінюють послідовність. Алгоритми, що змінюють послідовність.

3. Структура навчальної дисципліни

Назви розділів і тем	Кількість годин											
	денна форма						заочна форма					
	усього	у тому числі					усього	у тому числі				
		л	п	лаб.	сем.	с. р.		л	п	лаб.	сем.	с. р.
1	2	3	4	5	6	7	8	9	10	11	12	13
Розділ 2. Практичне використання концепцій об’єктно орієнтованого програмування.												
Тема 8. Віртуальні функції. Поняття абстрактного класу та чисті віртуальні функції. Віртуальні деструктори. Множинне спадкування, його види та правила застосування.	15	2		4		9						
*Тема 9. Перевантаження операторів +, -, /, *, =, +=, -=, *=, /=, [ ], ( ), ->, *, постфіксного і префіксного ++ та -- .*	15	2		4		9						
Тема 10. Перевантаження операторів потокового вводу/виводу. Форматування виводу. Обробка виключень.	15	2		4		9						
Тема 11. Ієрархія потокових класів. Класи istream, ostream. Аналіз та управління станом потока.	15	2		4		9						
Тема 12. Стандартні консольні потоки вводу/виводу. Ввод даних з використанням cin. Методи об’єкта cin. Ввод одного символу та рядка. Вивід даних з використанням cout.	15	2		4		9						

Методи об'єкта cout.												
Тема 13. Використання файлів для вводу/виводу даних. Класи ofstream та ifstream. Бінарні та текстові файли.	15	2		4		9						
Тема 14. Поняття шаблону. Шаблонні функції. Шаблонні класи. Статичні члени класу та шаблони. Стандартна бібліотека шаблонів. Контейнерні класи. Ітератори. Послідовні контейнери: vector, list, stack, queue. Асоціативні контейнери: map, multimap, set, multiset.	15	2		4		9						
Тема 15. Класи алгоритмів. Алгоритми, що не змінюють послідовність. Алгоритми, що змінюють послідовність.	15	2		4		9						
Разом за Розділом 2	120	16		32		72						
Усього годин	120	16		32		72						

4. Теми лабораторних занять

№ з/п	Назва теми	Кількість годин
Розділ 2		
1	*Віртуальні функції. Поняття абстрактного класу та чисті віртуальні функції. Віртуальні деструктори. Множинне спадкування, його види та правила застосування.*	4
2	*Перевантаження операторів +, -, /, *, =, +=, -=, *=, /=, [ ], ( ), ->, *, постфіксного і префіксного ++ та --.*	4
3	*Перевантаження операторів потокового вводу/виводу. Форматування виводу. Обробка виключень.*	4
4	*Ієрархія потокових класів. Класи istream, ostream. Аналіз та управління станом потока.*	4

5	*Стандартні консольні потоки вводу/виводу. Ввод даних з використанням cin. Методи об'єкта cin. Ввод одного символу та рядка. Вивід даних з використанням cout. Методи об'єкта cout.*	4
6	*Використання файлів для вводу/виводу даних. Класи ofstream та ifstream. Бінарні та текстові файли.*	4
7	*Поняття шаблону. Шаблонні функції. Шаблонні класи. Статичні члени класу та шаблони. Стандартна бібліотека шаблонів. Контейнерні класи. Ітератори. Послідовні контейнери: vector, list, stack, queue. Асоціативні контейнери: map, multimap, set, multiset.*	4
8	*Класи алгоритмів. Алгоритми, що не змінюють послідовність. Алгоритми, що змінюють послідовність.*	4
Разом за Розділом 2		32
Усього годин		32

5. Завдання для самостійної роботи

№ з/п	Назва теми	Кількість годин
Розділ 2		
1	*Віртуальні функції. Поняття абстрактного класу та чисті віртуальні функції. Віртуальні деструктори. Множинне спадкування, його види та правила застосування.*	9
2	*Перевантаження операторів +, -, /, *, =, +=, -=, *=, /=, [ ], ( ), ->, *, постфіксного і префіксного ++ та --.*	9
3	*Перевантаження операторів потокового вводу/виводу. Форматування виводу. Обробка виключень.*	9
4	*Ієрархія потокових класів. Класи istream, ostream. Аналіз та управління станом потока.*	9
5	*Стандартні консольні потоки вводу/виводу. Ввод даних з використанням cin. Методи об'єкта cin. Ввод одного символу та рядка. Вивід даних з використанням cout. Методи об'єкта cout.*	9
6	*Використання файлів для вводу/виводу даних. Класи ofstream та ifstream. Бінарні та текстові файли.*	9
7	*Поняття шаблону. Шаблонні функції. Шаблонні класи. Статичні члени класу та шаблони. Стандартна бібліотека шаблонів. Контейнерні класи. Ітератори. Послідовні контейнери: vector, list, stack, queue. Асоціативні контейнери: map, multimap, set, multiset.*	9
8	*Класи алгоритмів. Алгоритми, що не змінюють послідовність. Алгоритми, що змінюють послідовність.*	9
Разом за Розділом 2		72
Усього годин		72

6. Індивідуальні завдання

Не передбачено.

7. Методи навчання

Методи навчання, що застосовуються при викладанні навчальної дисципліни «Об'єктно-орієнтоване програмування. Частина 2»:

1) Пояснювально-ілюстративний метод. Викладання лекційного матеріалу дисципліни у вигляді презентацій за допомогою мультимедійного обладнання.

2) Репродуктивний метод. Відтворення студентами набутих теоретичних знань при виконанні лабораторних робіт.

3) Дослідницький метод. Виконання студентами самостійних робіт.

8. Методи контрою

1) Написання контрольної роботи. За навчальним планом передбачено дві контрольні роботи. Максимальна кількість балів, яку може отримати студент за написання однієї контрольної роботи, складає 10 балів. За написання усіх контрольних робіт студент за семестр може набрати максимум 20 балів.

2) Виконання лабораторних робіт. В навчальній дисципліні передбачено виконання 4 лабораторних робіт. За виконання кожної лабораторної роботи студент отримує від 0 до 10 балів. Нарахування балів здійснюється за оформлений звіт, початковий код програми, відповіді на питання по коду програми, проходження тестів без збоїв в роботі програми. За виконання усіх лабораторних робіт студент за семестр може набрати максимум 40 балів.

3) Написання залікової роботи. Максимальна кількість балів, яку студент може отримати за написання залікової роботи складає 40 балів. Мінімальна кількість балів для допуску до підсумкового контролю – 20 балів.

9. Схема нарахування балів

Підсумковий семестровий контроль в формі заліку.

Критерії оцінювання навчальних досягнень

Назва роботи	**Оцінка**
Лабораторна робота № 1	10
Лабораторна робота № 2	10
Лабораторна робота № 3	10
Лабораторна робота № 4	10
Контрольна робота № 1	10
Контрольна робота № 2	10
Залік	40
Сума	**100**

Шкала оцінювання

Сума балів за всі види навчальної діяльності протягом семестру	Оцінка	
	для чотирирівневої шкали оцінювання	для дворівневої шкали оцінювання
90-100	відмінно	зараховано
70-89	добре	
50-69	задовільно	
1-49	незадовільно	не зараховано

10. Рекомендована література

Основна література

1) *Deitel P., Deitel H.* C++ How to Program, 10th Edition. (2016). ISBN-13: 978-0134448237
2) *Stanley B. Lippman.* C++ Primer: Completely Rewritten for the New C++11 Standard. (2012). ISBN-13: 978-0321714114
3) *Scott Meyers.* Effective Modern C++: 42 Specific Ways to Improve Your Use of C++11 and C++14. (2014). ISBN-13: 978-1491903995
4) *Bjarne Stroustrup.* The C++ Programming Language, 4th Edition. (2013). ISBN-13: 978-0321563842

Допоміжна література

1) *Andrew Koenig.* Accelerated C++: Practical Programming by Example. (2000). ISBN-13: 978-0201703535
2) *Bjarne Stroustrup.* Programming: Principles and Practice Using C++. (2008). ISBN-13: 978-0321543721
3) *Bjarne Stroustrup.* A Tour of C++, 2nd Edition. (2018). ISBN-13: 978-0134997834
4) *Herbert Schildt.* C++ The Complete Reference, 5th Edition. (2012). ISBN-13: 978-0071634809

11. Посилання на інформаційні ресурси в Інтернеті, відео-лекції, інше методичне забезпечення

1) Довідник по мові C++ https://learn.microsoft.com/en-us/cpp/cpp/?view=msvc-170