Міністерство освіти і науки України

Харківський національний університет імені В.Н. Каразіна

Кафедра інформаційних технологій в фізико-енергетичних системах

“ЗАТВЕРДЖУЮ”

Проректор

з наукової та педагогічної роботи



_____ Олександр ГОЛОВКО

“____”_____20____ р.

РОБОЧА ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

ОСНОВИ 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

2023 / 2024 навчальний рік

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

" 28 " серпня 2023 року, протокол № 8/23

РОЗРОБНИКИ ПРОГРАМИ:

Гарячевська Ірина Василівна, к.т.н., доцент кафедри інформаційних технологій в фізико-енергетичних системах.

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

Протокол від " 28 " серпня 2023 року № 8/23

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)

_____ Руслан СУХОВ
(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)

Програму погоджено методичною комісією навчально-наукового інституту комп'ютерної фізики та енергетики

Протокол від " 28 " серпня 2023 року № 8/23

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

_____ Денис ПРОТЕКТОР
(підпис)

ВСТУП

Програма навчальної дисципліни “Основи 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 викладання навчальної дисципліни є знайомство студентів з основами програмування, принципами побудови програм, вивчення інструментальних засобів програмування.

1.2. Основні завдання вивчення дисципліни

Основними завданнями вивчення дисципліни є оволодіння основними концепціями програмування мовою С++, вміння самостійно створювати програми для вирішення фізичних та математичних задач в галузі енергетики.

1.3. Кількість кредитів 3

1.4. Загальна кількість годин 90

1.5. Характеристика навчальної дисципліни	
Обов’язкова	
Денна форма навчання	Заочна (дистанційна) форма навчання
Рік підготовки	
2-й	-й
Семестр	
3-й	-й
Лекції	
16 год.	год.
Практичні, семінарські заняття	
год.	год.
Лабораторні заняття	
32 год.	год.
Самостійна робота	
42 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:

- основні концепції програмування мовою С++.

В результаті вивчення навчальної дисципліни студент повинен вміти:

- 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;
- класифікувати, аналізувати та інтерпретувати науковотехнічну інформацію в галузі прикладної фізики;
- презентувати результати досліджень і розробок фахівцям і нефахівцям, аргументувати власну позицію;
- розробляти програмне та програмно-апаратне забезпечення в фізико-енергетичних системах.

2. Тематичний план навчальної дисципліни

Розділ 2. Динамічні структури даних.

Тема 9. Вказівники та посилання в С++.

Тема 10. Функції.

Тема 11. Динамічні масиви. Операції new і delete. Передача динамічного масиву в функцію.

Тема 12. Символьні дані. Об'ява символьних змінних. Ввод-вивід строк. Клас string.

Тема 13. Обробка даних. Алгоритми сортування та пошуку в С++. Алгоритми роботи на графах.

Тема 14. Робота з типізованими файлами. Клас fstream.

Тема 15. Динамічні структури даних. Список. Черга. Стек.

Тема 16. Класи. Загальні принципи побудови класу. Підключення класів до головної програми. Графіка в С++. Інтерфейс графічних пристроїв GDI.

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.	інд.	с. р.		л	п	лаб.	інд.	с. р.
1	2	3	4	5	6	7	8	9	10	11	12	13
Розділ 2. Динамічні структури даних.												
Тема 9. Вказівники та посилання в С++.	11	2		4		5						
Тема 10. Функції.	11	2		4		5						
Тема 11. Динамічні масиви. Операції new і delete. Передача динамічного масиву в функцію.	11	2		4		5						
Тема 12. Символьні дані. Об'ява символьних змінних. Ввод-вивід строк. Клас string.	11	2		4		5						
Тема 13. Обробка даних. Алгоритми сортування та пошуку в С++. Алгоритми роботи на графах.	11	2		4		5						
Тема 14. Робота з типізованими файлами. Клас fstream.	11	2		4		5						
Тема 15. Динамічні структури даних. Список. Черга. Стек.	11	2		4		5						
Тема 16. Класи. Загальні принципи побудови класу. Підключення класів до головної програми. Графіка в С++. Інтерфейс графічних пристроїв GDI.	13	2		4		7						
Разом за Розділом 2	90	16		32		42						
Усього годин	90	16		32		42						

4. Теми семінарських (практичних, лабораторних) занять

№ з/п	Назва теми	Кількість годин
II семестр		
1	*Вказівники та посилання в С++*	4
2	*Функції.*	4
3	*Динамічні масиви. Операції new і delete. Передача динамічного масиву в функцію.*	4
4	*Символьні дані. Об'ява символьних змінних. Ввод-вивід строк. Клас string.*	4
5	*Обробка даних. Алгоритми сортування та пошуку в С++. Алгоритми роботи на графах.*	4
6	*Робота з типізованими файлами. Клас fstream.*	4
7	*Динамічні структури даних. Список. Черга. Стек.*	4
8	*Класи. Загальні принципи побудови класу. Підключення класів до головної програми. Графіка в С++. Інтерфейс графічних пристроїв GDI.*	4
Разом за II семестр		32
Усього годин		32

5. Завдання для самостійної роботи

№ з/п	Назва теми	Кількість годин
II семестр		
1	*Навчитися працювати з вказівниками та посиланнями в С++.*	5
2	*Оволодіти навичками роботи з функціями.*	5
3	*Оволодіти навичками роботи з динамічними масивами та з операторами new і delete.*	5
4	*Навчитися працювати з символьними даними в С++. Оволодіти навичками роботи з класом string.*	5
5	*Реалізувати алгоритми сортування та пошуку в С++. Реалізувати алгоритми роботи на графах.*	5
6	*Опанувати роботу з типізованими файлами та класом fstream.*	5
7	*Навчитися працювати з динамічними структурами даних.*	5
8	*Вивчити загальні принципи побудови класу. Навчитися підключати класи до головної програми. Опанувати роботу з графікою в С++ із використанням інтерфейсу графічних пристроїв GDI.*	7
Разом за II семестр		42
Усього годин		42

Самостійна робота студента включає виконання завдання (окремий варіант для кожного студента), оформлення звіту за результатами виконаного завдання, та захист самостійної роботи.

6. Індивідуальні завдання

Не передбачено.

7. Методи навчання

Методи навчання, що застосовуються при викладанні навчальної дисципліни «Основи 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) Розрахунково-графічна робота. За навчальним планом передбачена одна розрахунково-графічна робота. Максимальна кількість балів, яку може отримати студент за написання розрахунково-графічної роботи, складає 12 балів. Написання розрахунково-графічної роботи полягає в складанні алгоритму роботи програми у вигляді блок-схеми з подальшою реалізацією алгоритму роботи програми мовою програмування С++. Нарахування балів здійснюється за наступними критеріями:

12 балів нараховується, коли:

- блок-схема та алгоритм роботи програми складені логічно правильно;
- правильно оформлені вхідні та вихідні дані;
- немає помилок у використанні структурних елементів блок-схеми та алгоритму роботи програми;
- студент без помилок читає блок-схему та алгоритм роботи програми;
- початковий код програми не має синтаксичних та логічних помилок;
- студент вільно орієнтується в початковому коді програми.

9 балів нараховується, коли:

- блок-схема та алгоритм роботи програми складені логічно правильно, але були допущені 1-2 помилки або 2-3 недоліка;
- початковий код програми складено логічно правильно, але були допущені 1-2 синтаксичні помилки або 2-3 недоліка.

6 балів нараховується, коли:

- допущені помилки в алгоритмі роботи програми;
- неправильно використовуються структурні елементи блок-схеми;
- у поясненні алгоритму роботи програми та блок-схеми студент відчував труднощі, які були виправлені за допомогою викладача;
- початковий код програми має синтаксичні та логічні помилки;
- у поясненні початкового коду програми студент відчував труднощі, які були виправлені за допомогою викладача.

3 балів нараховується, коли:

- допущені істотні помилки в оформленні алгоритму роботи програми та блок-схеми;
- студент не володіє основними правилами оформлення алгоритму роботи програми та блок-схеми;
- допущені грубі помилки в алгоритмі роботи програми, які студент не може виправити навіть за допомогою додаткових запитань викладача;

- початковий код програми має істотні синтаксичні та логічні помилки, які студент не може виправити навіть за допомогою додаткових запитань викладача.

0 балів нараховується, коли:

- студент показує повне незнання алгоритмічних конструкцій і структурних елементів блок-схеми;
- студент показує повне незнання синтаксису мови програмування.

2) Захист самостійних робіт. За навчальним планом передбачено виконання 8 самостійних робіт. Кожна самостійна робота оцінюється від 0 до 3 балів. Нарахування балів здійснюється за оформлений звіт, початковий код програми, відповіді на питання по коду програми, проходження тестів без збоїв в роботі програми. За семестр студент може набрати максимум 24 бали за виконання 8 самостійних робіт.

3) Виконання лабораторних робіт. За навчальним планом передбачено виконання 8 лабораторних робіт. За виконання кожної лабораторної роботи студент отримує від 0 до 2 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 16 балів.

4) Виконання поточних задач. За виконання поточних задач по кожній темі студент отримує від 0 до 1 балів. Нарахування балів здійснюється за оформлений звіт, початкові коди програм, відповіді на питання по коду програм, проходження тестів без збоїв в роботі програм. За виконання усіх поточних задач студент за семестр може набрати максимум 8 балів.

9. Схема нарахування балів

Підсумковий семестровий контроль в формі заліку.

Мінімальна кількість балів для допуску до підсумкового контролю – 20 балів.

Критерії оцінювання навчальних досягнень

Назва роботи	Оцінка
Лабораторна робота № 1	2
Лабораторна робота № 2	2
Лабораторна робота № 3	2
Лабораторна робота № 4	2
Лабораторна робота № 5	2
Лабораторна робота № 6	2
Лабораторна робота № 7	2
Лабораторна робота № 8	2
Поточні задачі № 1	1
Поточні задачі № 2	1
Поточні задачі № 3	1
Поточні задачі № 4	1
Поточні задачі № 5	1
Поточні задачі № 6	1
Поточні задачі № 7	1
Поточні задачі № 8	1
Самостійна робота № 1	3
Самостійна робота № 2	3

Самостійна робота № 3	3
Самостійна робота № 4	3
Самостійна робота № 5	3
Самостійна робота № 6	3
Самостійна робота № 7	3
Самостійна робота № 8	3
Розрахунково-графічна робота	12
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. Рекомендоване методичне забезпечення

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://en.cppreference.com/w/