

Міністерство освіти і науки України

Харківський національний університет імені В.Н. Каразіна

Кафедра інформаційних технологій в фізико-енергетичних системах



Робоча програма навчальної дисципліни

Основи програмування

спеціальність: 105 Прикладна фізика та наноматеріали

освітня програма: "Прикладна фізика енергетичних систем"

спеціалізація : фізика нетрадиційних енерготехнологій та фізичні аспекти екології
теплофізика, молекулярна фізика і енергоефективність
інформаційні технології обробки даних в енергетичних системах

освітня програма: "Комп'ютерна фізика"

факультет фізико-енергетичний

Програму рекомендовано до затвердження вченою радою фізико-енергетичного факультету

“21 ” червня 2018 року, протокол № 6 /18

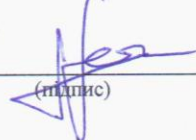
РОЗРОБНИКИ ПРОГРАМИ: (вказати авторів, їхні наукові ступені, вчені звання та посади)

Гарячевська.І В., к.т.н., доц. кафедри інформаційних технологій в фізико-енергетичних системах

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

Протокол від “20” червня 2018 року № 6-2/18


Завідувач кафедри інформаційних технологій в фізико-енергетичних системах


(підпис)

Нємченко К.Е.

Програму погоджено методичною комісією
Фізико-енергетичного факультету

Протокол від “20 ” червня 2018 року, протокол № 6 /18
Голова методичної комісії фізико-енергетичного факультету


(підпис)

Лісіна О.Ю.

ВСТУП

Програма навчальної дисципліни “Основи програмування” складена відповідно до освітньо-професійної (освітньо-наукової) програми підготовки бакалаврів

спеціальність: 105 Прикладна фізика та наноматеріали

освітня програма: “Прикладна фізика енергетичних систем”

спеціалізація: фізика нетрадиційних енерготехнологій та фізичні аспекти екології
теплофізика, молекулярна фізика і енергоефективність
інформаційні технології обробки даних в енергетичних системах

освітня програма: “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 C++, вміння самостійно створювати програми для вирішення фізичних та математичних задач в галузі енергетики.

1.3. Кількість кредитів 7

1.4. Загальна кількість годин 210

1.5. Характеристика навчальної дисципліни	
Нормативна	
Денна форма навчання	Заочна (дистанційна) форма навчання
Рік підготовки	
1-й	-й
Семестр	
1-й	-й
Лекції	
16 год.	год.
Практичні, семінарські заняття	
год.	год.
Лабораторні заняття	
32 год.	год.
Самостійна робота	
42 год.	год.
Індивідуальні завдання	
год.	

Рік підготовки	
1-й	-й
Семестр	
2-й	-й
Лекції	
32 год.	год.
Практичні, семінарські заняття	
год.	год.
Лабораторні заняття	
32 год.	год.
Самостійна робота	
56 год.	год.
Індивідуальні завдання	
год.	

1.6. Заплановані результати навчання

Згідно з вимогами освітньо-професійної (освітньо-наукової) програми студенти повинні досягти таких результатів навчання: В результаті вивчення навчальної дисципліни студент повинен

знати: основні концепції програмування мовою C++ та принципи побудови програм

вміти: самостійно створювати програми мовою C++ для вирішення фізичних та математичних задач в галузі енергетики

2. Тематичний план навчальної дисципліни

Розділ 1. Основи програмування.

Тема 1. Структура програми.

Зміст: Загальні правила написання програми мовою C++. Обробка програми комп'ютером. Системи числення.

Тема 2. Типи даних.

Зміст: Типи даних, розмір та об'ява. Властивості змінної, правила формування імені змінної.

Тема 3. Операції.

Зміст: Арифметичні та логічні операції.

Тема 4. Оператори.

Зміст: Оператори введення та виведення. Оператори розгалуження та вибору. Оператори циклу. Відмінності циклів.

Тема 5. Обробка масивів.

Зміст: Об'ява масиву. Введення та виведення елементів масиву. Алгоритми сортування та пошуку.

Тема 6. Показчики та посилання.

Зміст: Об'ява та ініціалізація показників. Об'ява та ініціалізація посилань. Оператори new та delete. Об'ява динамічного масиву.

Тема 7. Матриці.

Зміст: Об'ява матриці. Введення та виведення елементів матриці. Алгоритми.

Розділ 2. Динамічні структури даних.

Тема 1. Функції.

Зміст: Об'ява функцій. Виклик функцій. Формальні та фактичні параметри. Передача параметрів в функцію. Рекурсивні функції. Перевантаження функцій.

Тема 2. Технологія роботи з файлами.

Зміст: Створення файлів. Читання та запис інформації в файл.

Тема 3. Структури.

Зміст: Правила об'яви структури. Робота з полями структури. Об'ява структурної змінної. Вкладеність структур.

Тема 4. Список, стек та черга.

Зміст: Однозв'язний список. Додавання та видалення елементів зі списку. Пошук елемента в списку. Стек. Черга.

Тема 5. Алгоритми роботи на графах.

Зміст: Алгоритм Дейкстри, Флойда-Уоршела, жадібний алгоритм.

Тема 6. Класи.

Зміст: Загальні принципи побудови класу. Підключення класів до головної програми.

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 1. Основи програмування.												
Тема 1. Структура програми.	12	2		4		6						
Тема 2. Типи даних.	12	2		4		6						
Тема 3. Операції.	12	2		4		6						
Тема 4. Оператори.	12	2		4		6						
Тема 5. Обробка масивів.	12	2		4		6						
Тема 6. Показники та посилання.	15	3		6		6						
Тема 7. Матриці.	15	3		6		6						
Разом за Розділом 1	90	16		32		42						
Розділ 2. Динамічні структури даних.												
Тема 1. Функції.	19	5		5		9						
Тема 2. Технологія роботи з файлами.	19	5		5		9						
Тема 3. Структури.	19	5		5		9						
Тема 4. Список, стек та черга.	22	6		6		10						
Тема 5. Алгоритми роботи на графах.	19	5		5		9						

1	2	3	4	5	6	7	8	9	10	11	12	13
Тема 6. Класи.	22	6		6		10						
Разом за Розділом 2	120	32		32		56						
Усього годин	210	48		64		98						

4. Теми семінарських (практичних, лабораторних) занять

№ з/п	Назва теми	Кількість годин
І семестр		
1	Структура програми.	4
2	Типи даних.	4
3	Операції.	4
4	Оператори.	4
5	Обробка масивів.	4
6	Показчики та посилання.	6
7	Матриці.	6
Разом за І семестр		32
II семестр		
1	Функції.	5
2	Технологія роботи з файлами.	5
3	Структури.	5
4	Список, стек та черга.	6
5	Алгоритми роботи на графах.	5
6	Класи.	6
Разом за II семестр		32
Усього годин		64

5. Завдання для самостійної роботи

№ з/п	Назва теми	Кількість годин
І семестр		
1	Структура програми.	6
2	Типи даних.	6
3	Операції.	6
4	Оператори.	6
5	Обробка масивів.	6
6	Показчики та посилання.	6
7	Матриці.	6
Разом за І семестр		42
II семестр		
1	Функції.	9
2	Технологія роботи з файлами.	9
3	Структури.	9
4	Список, стек та черга.	10
5	Алгоритми роботи на графах.	9
6	Класи.	10
Разом за II семестр		56
Усього годин		98

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

6. Методи контролю

1) Розрахунково-графічна робота. За навчальним планом у кожному семестрі передбачена одна розрахунково-графічна робота. Максимальна кількість балів, яку може отримати студент за написання розрахунково-графічної роботи, складає 20 балів. Написання розрахунково-графічної роботи полягає в складанні алгоритму роботи програми до одного з самостійних завдань (окремий варіант для кожного студента) у вигляді блок-схеми з подальшим захистом написаної роботи. Нарахування балів здійснюється за наступними критеріями:

20 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.

15 балів нараховується, коли:

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

10 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;

5 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.

0 балів нараховується, коли:

- студент показує повне незнання алгоритмічних конструкцій і структурних елементів блок-схеми.

2) Захист самостійних робіт. У кожному семестрі передбачено виконання 6 самостійних робіт. Кожна самостійна робота оцінюється від 0 до 10 балів. Нарахування балів здійснюється за оформлений звіт, робочу програму, відповіді на питання по коду програми, проходження тестів без збоїв в роботі програми. За семестр студент може набрати максимум 60 балів за виконання 6 самостійних робіт.

3) Виконання лабораторних робіт. У кожному семестрі передбачено виконання 6 лабораторних робіт. За виконання кожної лабораторної роботи студент отримує від 0 до 4 балів. За виконання усіх лабораторних робіт студент за семестр може набрати максимум 20 балів.

7. Схема нарахування балів

Підсумковий семестровий контроль в формі заліку без виконання залікової роботи.

I семестр

Поточний контроль, самостійна робота, індивідуальні завдання							
Розділ 1							Розрахунково-графічна робота
T1	T2	T3	T4	T5	T6	T7	Сума
11	11	11	11	12	12	12	20
							100

T1, T2, ... T7 – теми розділів.

II семестр

Поточний контроль, самостійна робота, індивідуальні завдання							
Розділ 2							Розрахунково-графічна робота
T1	T2	T3	T4	T5	T6		Сума
13	13	13	13	14	14		20
							100

T1, T2, ... T6 – теми розділів.

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

8. Рекомендоване методичне забезпечення

Основна література

- 1) Бьёрн Страуструп Программирование: принципы и практика использования C++
- 2) Бондарев В.М. — Программирование на C++(2005) ISBN 966-8530-50-0
- 3) Глушаков С.В., Коваль А.В., Смирнов С.В. — Язык программирования C++. Учебный курс(2001) ISBN 966-03-1282-2
- 4) Голуб А. — Правила программирования на С и C++(2001) ISBN: 5-7503-0090-0
- 5) Дейтел Х., Дейтел П. — Как программировать на C++ (3rd) (2001)
- 6) Дирк Хенкеманс, Марк Ли — Программирование на C++ ISBN: 5-93286-050-2
- 7) Кубенский А.А. — Структуры и алгоритмы обработки данных (2004)

Допоміжна література

- 1) *Бьєрн Страуструп* — The Design and Evolution of C++ (Дизайн и эволюция C++)
- 2) *Скотт Майерс* — Effective C++ (Эффективное использование C++)
- 3) *Герб Саттер* — Exceptional C++ (Решение сложных задач на C++)
- 4) *Герб Саттер, Андрей Александреску* — C++ Coding Standards (Стандарты программирования на C++).
- 5) *Джеймс Коплиен* — Advanced C++ Programming Styles and Idioms (Программирование на C++)

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

- 1) Довідник по мові C++ <https://docs.microsoft.com/en-us/cpp/cpp/cpp-language-reference?view=vs-2019>