

Міністерство освіти і науки України

Харківський національний університет імені В.Н. Каразіна  
Кафедра інформаційних технологій в фізико-енергетичних системах

“ЗАТВЕРДЖУЮ”

Проректор  
з науково-педагогічної роботи

  
Антон ПАНТЕЛЕЙМОНОВ  
“” 20\_\_ р.  


## РОБОЧА ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

### ОСНОВИ ПРОГРАМУВАННЯ

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

2020 / 2021 навчальний рік

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

“ 14 ” грудня 2020 року, протокол № 12/20

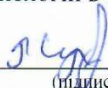
РОЗРОБНИКИ ПРОГРАМИ:

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

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

Протокол від “ 2 ” грудня 2020 року № 12/20

Завідувач кафедри інформаційних технологій в фізико-енергетичних системах

  
(підпис) \_\_\_\_\_ Руслан СУХОВ

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

Гарант освітньо-професійної програми (керівник проектної групи)

  
(підпис) \_\_\_\_\_ Світлана РОГОВА


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

Гарант освітньо-професійної програми (керівник проектної групи)

  
(підпис) \_\_\_\_\_ Руслан СУХОВ

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

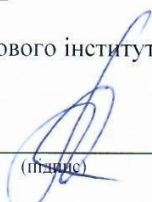
Гарант освітньо-професійної програми (керівник проектної групи)

  
(підпис) \_\_\_\_\_ Ілля МАРУЩЕНКО

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

Протокол від “ 14 ” грудня 2020 року № 12/20

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

  
(підпис) \_\_\_\_\_ Ольга ЛІСІНА

## 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»  
«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 C++, вміння самостійно створювати програми для вирішення фізичних та математичних задач в галузі енергетики.

#### 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	
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.
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:

- технології та методи експериментального дослідження властивостей речовин і матеріалів, включаючи наноматеріали, та застосовувати їх при розв'язанні практичних проблем прикладної фізики.

В результаті вивчення навчальної дисципліни студент повинен вміти:

- знаходити науково-технічну інформацію з різних джерел з використанням сучасних інформаційних технологій;
- класифікувати, аналізувати та інтерпретувати науково-технічну інформацію в галузі прикладної фізики;
- застосовувати фізичні, математичні та комп'ютерні моделі для дослідження фізичних явищ, розробки приладів і наукоємних технологій.

## 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 показчиків. 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.

### 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
<b>Розділ 1. Основи програмування</b>												
<i>Тема 1. Структура програми.</i>	16	4		4		8						
<i>Тема 2. Типи даних.</i>	16	4		4		8						
<i>Тема 3. Операції.</i>	16	4		4		8						
<i>Тема 4. Оператори.</i>	18	5		5		8						
<i>Тема 5. Обробка масивів.</i>	18	5		5		8						
<i>Тема 6. Показчики та посилання.</i>	18	5		5		8						
<i>Тема 7. Матриці.</i>	18	5		5		8						
<b>Разом за Розділом 1</b>	120	32		32		56						
<b>Усього годин</b>	120	32		32		56						

### 4. Теми семінарських (практичних, лабораторних) занять

№ з/п	Назва теми	Кількість годин
1	Структура програми.	4
2	Типи даних.	4
3	Операції.	4
4	Оператори.	5
5	Обробка масивів.	5
6	Показчики та посилання.	5
7	Матриці.	5
	Разом	32

### 5. Завдання для самостійної роботи

№ з/п	Види, зміст самостійної роботи	Кількість годин
1	Оволодіти загальними правилами написання програми мовою C++.	8
2	Ознайомитися з типами даних, їх розмірами та об'явою.	8
3	Ознайомитися з арифметичними та логічними операціями.	8
4	Навчитися користуватися операторами вводу та виводу, операторами розгалуження та циклу.	8
5	Оволодіти навичками роботи з масивами.	8
6	Навчитися користуватися показчиками та посиланнями.	8
7	Оволодіти навичками роботи з матрицями.	8
	Разом	56

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 «Інтегровані середовища розробки»:

- 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 контролю**

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 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. За навчальним планом передбачено виконання 6 самостійних робіт. Кожна самостійна робота оцінюється від 0 до 7 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 42 бали за виконання 6 самостійних робіт.

3) Виконання лабораторних робіт. За навчальним планом передбачено виконання 6 лабораторних робіт. За виконання кожної лабораторної роботи студент отримує від 0 до 4 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 20 балів.

4) Виконання поточних задач. За виконання поточних задач по кожній темі студент отримує від 0 до 3 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 18 балів.

## 9. Схема нарахування балів

Підсумковий семестровий контроль в формі заліку без виконання залікової роботи.

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

T1, T2, ... T7 – теми розділів.

## Критерії оцінювання навчальних досягнень

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

Самостійна робота № 3	7
Самостійна робота № 4	7
Самостійна робота № 5	7
Самостійна робота № 6	7
Розрахунково-графічна робота	20
<b>Сума</b>	<b>100</b>

### 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) *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 C 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) *Бьёрн 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++)

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

- 1) Довідник по мові C++ <https://docs.microsoft.com/ru-ru/cpp/cpp/?view=msvc-160>