

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



Пантелеймонов А.В.

25 червня 2019 р.

Робоча програма навчальної дисципліни

**ОБ'ЄКТНО – ОРІЄНТОВАНЕ ПРОГРАМУВАННЯ**

спеціальність: 105 Прикладна фізика та наноматеріали  
освітня програма: «Прикладна фізика енергетичних систем»  
факультет фізико-енергетичний

2019 / 2020 навчальний рік

Програму рекомендовано до затвердження Вченою радою фізико-енергетичного факультету

“25” червня 2019 року, протокол № 6/19

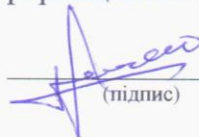
РОЗРОБНИКИ ПРОГРАМИ:

доктор фізико-математичних наук, професор Немченко Костянтин Едуардович

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

Протокол від “24” червня 2019 року № 6/19

Завідувач кафедри інформаційних технологій в фізико-енергетичних системах



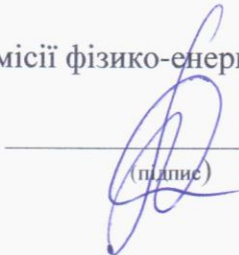
(підпис)

Немченко К.Е.

Програму погоджено методичною комісією фізико-енергетичного факультету

Протокол від “25” червня 2019 року № 6/19

Голова методичної комісії фізико-енергетичного факультету



(підпис)

Лісіна О.Ю.

## ВСТУП

Програма навчальної дисципліни «Об'єктно – орієнтоване програмування» складена відповідно до освітньо-професійної програми підготовки бакалавр

спеціальність: 105 Прикладна фізика та наноматеріали

освітня програма: «Прикладна фізика енергетичних систем»

### 1. Опис навчальної дисципліни

#### 1.1. Мета викладання навчальної дисципліни

**Метою** курсу «Об'єктно – орієнтоване програмування» 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 мовами C++, Java, Python 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	
Нормативна	
Денна форма навчання	Заочна (дистанційна) форма навчання
Рік підготовки	
3-й	-й
Семестр	
5-й	-й
Лекції	
48 год.	год.
Практичні, семінарські заняття	
	год.
Лабораторні заняття	
32 год.	год.
Самостійна робота	
40 год.	год.
Індивідуальні завдання	
0 год.	

#### 1.6. Заплановані результати навчання

Згідно з вимогами освітньо-професійної програми студенти повинні досягти таких результатів навчання:

**знати:** теоретичні засади об'єктно орієнтованого програмування на прикладі мов C++, Java, Python.

**вміти:** вміти використовувати на практиці теоретичні знання для кодування мовами C++, Java, Python.

## 2. Тематичний план навчальної дисципліни

**Розділ 1. Основи об'єктно орієнтованого програмування.**

**Тема 1.** Основні ідеї об'єктно орієнтованого програмування.

**Тема 2.** Структурний та об'єктно-орієнтований підходи у програмуванні. Поняття класу. Дані-члени. Методи-члени. Статичні члени класу.

**Тема 3.** Модифікатори доступу: приватні, захищені, відкриті члени класу. Інкапсуляція даних класу.

**Тема 4.** Конструктор, деструктор. Конструктор та деструктор за замовченням. Вказівники та посилання на об'єкти класів. Конструктор копіювання.

**Тема 5.** Оператори new та delete. Вказівник this. Ключове слово const при визначенні вказівника та функції-члена.

**Тема 6.** Спадкування, його види. Передача аргументів в конструктори базових класів.

**Тема 7.** Перевизначення методів у похідних класах. Поняття поліморфізму. Раннє та пізнє зв'язування.

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

**Тема 9.** Друзі класу. Друзі-функції та друзі-класи. Вибір між оператором-членом та оператором-другом.

**Розділ 2. Практичне використання концепцій об'єктно орієнтованого програмування.**

**Тема 10.** Перевантаження операторів +, -, /, \*, =, +=, -=, \*=, /=, [ ], ( ), -, \*, постфіксного і префіксного ++ та --).

**Тема 11.** Перевантаження операторів потокового введення/виведення. Форматування виводу. Обробка виключень.

**Тема 12.** Ієрархія потокових класів. Класи istream, ostream. Аналіз та управління станом потоку.

**Тема 13.** Стандартні консольні потоки введення/виведення. Введення даних з використанням cin. Методи об'єкта cin. Введення одного символу та строки. Виведення даних з використанням cout. Методи об'єкта cout.

**Тема 14.** Використання файлів для введення/виведення даних. Класи ofstream та ifstream. Бінарні та текстові файли.

**Тема 15.** 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 16.** Класи алгоритмів. Алгоритми, що не змінюють послідовність. Алгоритми, що змінюють послідовність.

## 3. Структура навчальної дисципліни

Назви розділів	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 об'єктно орієнтованого програмування.</b>												
Тема 1	8	1		2		5						
Тема 2	8	1		2		5						
Тема 3	8	1		2		5						
Тема 4	8	1		2		5						
Тема 5	8	1		2		5						
Тема 6	8	1		2		5						
Тема 7	8	1		2		5						

Тема 8	8	1		2		5						
Тема 9	7	1		2		4						
Разом за розділом 1	71	9		18		44						
<i>Розділ 2. Практичне використання концепцій об'єктно орієнтованого програмування.</i>												
Тема 1	7	1		2		4						
Тема 2	7	1		2		4						
Тема 3	7	1		2		4						
Тема 4	7	1		2		4						
Тема 5	7	1		2		4						
Тема 6	7	1		2		4						
Тема 7	7	1		2		4						
Разом за розділом 2	49	7		14		28						
<b>Усього годин</b>	120	16		32		72						

#### 4. Теми лабораторних занять

№ з/п	Назва теми	Кількість годин
1	<b>Тема 1.</b> Основні ідеї об'єктно орієнтованого програмування.	2
2	<b>Тема 2.</b> Структурний та об'єктно-орієнтований підходи у програмуванні. Поняття класу. Дані-члени. Методи-члени. Статичні члени класу.	2
3	<b>Тема 3.</b> Модифікатори доступу: приватні, захищені, відкриті члени класу. Інкапсуляція даних класу.	2
4	<b>Тема 4.</b> Конструктор, деструктор. Конструктор та деструктор за замовченням. Вказівники та посилання на об'єкти класів. Конструктор копіювання.	2
5	<b>Тема 5.</b> Оператори new та delete. Вказівник this. Ключове слово const при визначенні вказівника та функції-члена.	2
6	<b>Тема 6.</b> Спадкування, його види. Передача аргументів в конструктори базових класів.	2
7	<b>Тема 7.</b> Перевизначення методів у похідних класах. Поняття поліморфізму. Раннє та пізнє зв'язування.	2
8	<b>Тема 8.</b> Віртуальні функції. Поняття абстрактного класу та чисті віртуальні функції. Віртуальні деструктори. Множинне спадкування, його види та правила застосування.	2
9	<b>Тема 9.</b> Друзі класу. Друзі-функції та друзі-класи. Вибір між оператором-членом та оператором-другом.	2
10	<b>Тема 10.</b> Перевантаження операторів +, -, /, *, =, +=, -=, *=, /=, [ ], ( ), ->, *, постфіксного и префіксного ++ та --).	2
11	<b>Тема 11.</b> Перевантаження операторів потокового введення/виведення. Форматування виводу. Обробка виключень.	2
12	<b>Тема 12.</b> Ієрархія потокових класів. Класи istream, ostream. Аналіз та управління станом потоку.	2
13	<b>Тема 13.</b> Стандартні консольні потоки введення/виведення. Введення даних з використанням cin. Методи об'єкта cin. Введення одного символу та строки. Виведення даних з використанням cout. Методи об'єкта cout.	2
14	<b>Тема 14.</b> Використання файлів для введення/виведення даних. Класи	2

	ofstream та ifstream. Бінарні та текстові файли.	
15	<b>Тема 15.</b> 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.	2
16	<b>Тема 16.</b> Класи алгоритмів. Алгоритми, що не змінюють послідовність. Алгоритми, що змінюють послідовність.	2
	Усього	32

## 5. Завдання для самостійної роботи

№ з/п	Назва теми	Кількість годин
1	Ознайомитись з основними ідеями об'єктно орієнтованого програмування.	5
2	Дослідження структурного та об'єктно-орієнтованого підходів у програмуванні.	5
3	Вивчення модифікаторів доступу: приватні, захищені, відкриті члени класу.	5
4	Вивчення понять конструкторів та деструкторів.	5
5	Дослідження понять операторів new та delete.	5
6	Спадкування, його види. Передача аргументів в конструктори базових класів.	5
7	Перевизначення методів у похідних класах.	5
8	Віртуальні функції. Поняття абстрактного класу та чисті віртуальні функції.	5
9	Друзі класу. Друзі-функції та друзі-класи. Вибір між оператором-членом та оператором-другом.	4
10	Перевантаження операторів +, -, /, *, =, +=, -=, *=, /=, [], (), ->, *, постфіксного и префіксного ++ та --).	4
11	Перевантаження операторів потокового введення/виведення. Форматування виводу.	4
12	Ієрархія потокових класів.	4
13	Стандартні консольні потоки введення/виведення. Введення даних з використанням cin.	4
14	Використання файлів для введення/виведення даних.	4
15	Поняття шаблону. Шаблонні функції. Шаблонні класи.	4
16	Класи алгоритмів. Алгоритми, що не змінюють послідовність. Алгоритми, що змінюють послідовність.	4
	Усього	72

## 6. Індивідуальні завдання

Розрахункова робота або контрольна робота не передбачена навчальним планом

## 7. Методи контролю

На лабораторних заняттях – експрес-опитування, виконання лабораторних робіт. Оцінка виставляється за результатами поточного контролю.

Навчальна програма нормативної дисципліни «Об'єктно – орієнтоване програмування» відповідає чинним нормативним документам, рекомендованим Міністерством освіти і науки України як навчальна програма для студентів фізичного спрямування для вищих навчальних закладів. Навчальна дисципліна «Об'єктно – орієнтоване програмування» є складовою циклу професійної підготовки фахівців освітньо–кваліфікаційного рівня «бакалавр».

Зміст програми, об'єм учбових питань дисципліни «Об'єктно – орієнтоване програмування» визначаються потребою загальнонаукової, загально-інженерної та технічної підготовки.

Вивчення дисципліни «Об'єктно – орієнтоване програмування» передбачає викладання лекцій, проведення практичних занять, виконання лабораторних робіт. Підсумковий контроль знань здійснюється на екзамені.

На лекціях викладається теоретичний матеріал, який ілюструється типовими прикладами і задачами за профілем підготовки фахівців з прикладної фізики та наноматеріалів. Викладання лекційного матеріалу має закінчений характер, здійснюється у доступній і наочній формі, містить проблемні ситуації.

Основною метою лабораторних занять є розвиток навичок практичного застосування і закріплення теоретичного матеріалу. При вирішенні задач рекомендується користуватися стандартними прийомами і методиками. На кожному практичному занятті частину учбового часу доцільно використовувати для самостійного розв'язку задач, контролюючи при цьому правильність обраного методу і ходу рішення.

Важливим фактором засвоєння курсу «Об'єктно – орієнтоване програмування» й оволодіння її методами є самостійна робота студентів. Для самостійного відпрацювання розділів і тем дисципліни пропонуються лабораторні роботи, при перевірці виконання яких здійснюється ефективний контроль за рівнем засвоєння матеріалу.

На всіх заняттях підкреслюється роль і значення предмету на конкретних прикладах і задачах прикладного характеру, висвітлюється її зв'язок з іншими дисциплінами і майбутньою професією.

## 8. Схема нарахування балів

Поточний контроль, самостійна робота	
Розділ 1	
Теми розділів	Кількість балів
Тема 1	6
Тема 2	6
Тема 3	6
Тема 4	6
Тема 5	6
Тема 6	6
Тема 7	6
Тема 8	6
Тема 9	6
Разом за розділом 1	54
Розділ 2	
Тема 10	6
Тема 11	6
Тема 12	6
Тема 13	6
Тема 14	6
Тема 15	6
Тема 16	6
Разом за розділом 2	36

Залікова робота	10
<b>Сума</b>	<b>100</b>

### Шкала оцінювання

Сума балів за всі види навчальної діяльності протягом семестру	Оцінка	
	для екзамену	для заліку
90 – 100	відмінно	зараховано
70-89	добре	
50-69	задовільно	
1-49	незадовільно	не зараховано

### 9. Рекомендована література

#### Основна література

1. Т.А. Павловская, Ю.А.Щупак. С++. Объектно-ориентированное программирование. Практикум. – СПб, Питер, 2005.

#### Допоміжна література

1. А.Фридман, Л.Кландер, М.Михаэлис, Х.Шилдт. С/С++. Алгоритмы и приемы программирования. – Москва. ООО «Бином-Пресс», 2003 г. – 560 с.
2. Культин Н.Б. С/С++ в задачах и примерах. – СПб: «БХВ–Петербург», 2005. – 288 с.
3. Глушаков С.В., Коваль А.В., Смирнов С.В. Практикум по С++. – Харьков, Фолио, 2006. – 525 с.
4. С. Прата. Язык программирования С++. Лекции и упражнения. – М.: ООО «ДиаСофтЮП», 2005. – 1104 с.
5. Рэй Лишнер. STL. Карманный справочник. Руководство по использованию. – СПб, Питер, 2005.
6. Липпман С., Лажойе Ж. Язык программирования С++. Вводный курс. 3-е изд. – СПб.-М.: Невский проспект – ДМК Пресс, 2001.

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

1. Мережа Internet.
2. Бібліотеки ХНУ імені В.Н.Каразіна.