

Міністерство освіти і науки України
Львівський національний університет імені Івана Франка
Кафедра радіофізики та комп'ютерних технологій

“ЗАТВЕРДЖУЮ”

Декан факультету електроніки та
комп'ютерних технологій

_____ Фургала Ю.М.
“ _____ ” _____ 2019 року

«ОБ’ЄКТНО-ОРІЄНТОВАНЕ ПРОГРАМУВАННЯ»

ПРОГРАМА

навчальної дисципліни підготовки бакалаврів

галузі знань	12	– «Інформаційні технології»
спеціальності	122	– «Комп'ютерні науки»
спеціалізації		– «Інформаційні технології»

(шифр за ОПП _____)

Львів – 2019

РОЗРОБЛЕНО ТА ВНЕСЕНО:

Львівським національним університетом імені Івана Франка

РОЗРОБНИК ПРОГРАМИ:

Хвищун І.О. канд. техн. наук, доцент кафедри радіофізики та комп'ютерних технологій

Затверджено на засіданні Вченої ради факультету електроніки та комп'ютерних технологій

“_30_” серпня 2019 року, протокол № 7/10

Голова Вченої ради

Фургала Ю.М.

Обговорено та рекомендовано до затвердження Навчально-методичною радою факультету електроніки та комп'ютерних технологій

“_29_” серпня 2019 року, протокол №_4

Голова Навчально-методичної ради
факультету електроніки та комп'ютерних технологій

Лучечко А.П.

Робочу програму схвалено на засіданні кафедри радіофізики та комп'ютерних технологій

“28” серпня 2019 року, протокол № ____

Завідувач кафедри
радіофізики та комп'ютерних технологій

Болеста І.М.

ВСТУП

Програма вивчення навчальної дисципліни «**Об'єктно-орієнтоване програмування**» складена відповідно до освітньо-професійної програми підготовки здобувачів першого (бакалаврського) рівня вищої освіти спеціальності 122 – «Комп'ютерні науки» спеціалізації – «Інформаційні технології», затвердженої наказом Міністерства освіти і науки № 485 від 26 травня 2010 року.

Міждисциплінарні зв'язки: навчальна дисципліна «**Об'єктно-орієнтоване програмування**» вивчається у логічному взаємозв'язку з іншими навчальними дисциплінами, що обумовлює необхідність постійного обліку та реалізації викладачами існуючих міжпредметних зв'язків з суміжними навчальними курсами, зокрема, такими як «Алгоритмізація та програмування», «Математичний аналіз», «Дискретна математика». Знання отримані під час вивчення дисципліни, сприятимуть формуванню у бакалаврів комп'ютерних наук розуміння та навиків використання ідей об'єктно-орієнтованого програмування (ООП) у розробці сучасних великих програмних проєктів, які призначені для розв'язання різноманітних науково-технічних задач.

Предметом вивчення навчальної дисципліни є освоєння ідей розробки великих програмних проєктів на основі використання принципів ООП, ознайомлення із показниками якості програмних продуктів, освоєння засобів для побудови графічних інтерфейсів користувача в таких сучасних об'єктно-орієнтованих середовищах розробки інформаційних систем, як Embarcadero RAD Studio Delphi 2010, Lazarus і Visual Studio 2017.

Програма навчальної дисципліни складається з таких модулів:

МОДУЛЬ 1. Вивчення способів реалізації принципів ООП мовою програмування Delphi

Змістовий модуль 1. Вивчення роботи програм у захищеному режимі 32-розрядного Intel-сумісного мікропроцесора та принципів ООП в Lazarus, Delphi.

МОДУЛЬ 2. Вивчення реалізації принципів ООП мовою програмування C++

Змістовий модуль 2. Вивчення принципів ООП в середовищі Visual C++ 2012.

1. МЕТА ТА ЗАВДАННЯ НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

1.1. Мета викладання навчальної дисципліни «Об'єктно-орієнтоване програмування» полягає в одержанні студентами знань з основ об'єктно-орієнтованого програмування алгоритмічними мовами Delphi, C++ в середовищах Embarcadero RAD Studio Delphi 2010 і Visual Studio 2017. Поглиблення знань про вбудовані, структуровані типи даних та алгоритмічні конструкції цих мов. Вивчення базових принципів ООП: абстрагування, ієрахічність, інкапсулювання, успадкування та поліморфізм. Освоєння способів побудови графічних інтерфейсів користувача (меню, панелі інструментів, шаблони діалогових вікон) в сучасних об'єктно-орієнтованих середовищах розробки інформаційних систем.

1.2. Основні завдання вивчення дисципліни: навчити студентів розуміти відмінності технології ООП від імперативно-модульної технології, опанувати принципи створення програмних систем у вигляді дискретних об'єктів, які поєднують у собі структури даних і поведінку об'єкта з ними, розуміти способи класифікації об'єктів і побудови ієрахічних структур класів, а також ідеї об'єктноорієнтованого проектування та аналізу програмних систем

1.3. Згідно з вимогами освітньо-професійної програми студенти повинні:

знати:

правила роботи в середовищі сучасних операційних систем та їхні файлові системи;
 правила використання об'єктно-орієнтованих мов програмування: Delphi, C++;
 принципи побудови об'єктної моделі;
 поняття класу, об'єкту, поля, методу, властивості та події.
 складові частини Win32-проектів;
 принципи роботи в інтегрованих середовищах середовищах Embarcadero RAD Studio Delphi 2010 і Visual Studio 2012;
 різницю між компонентною і каркасною моделлю побудови Windows-додатку;

вміти:

застосовувати об'єктно-орієнтований підхід для проектування складних програмних систем;
 здійснювати об'єктно-орієнтований аналіз, проектування та програмування складних програмних систем;
 інсталювати та налаштовувати середовища Embarcadero RAD Studio Delphi 2010 і Visual Studio 2012;
 створювати та налагоджувати проекти в названих середовищах;
 програмувати об'єктноорієнтованими мовами Delphi та C++ алгоритми обробки та відображення текстових, числових та графічних даних;
 створювати власні компоненти та бібліотеки DLL.

На вивчення навчальної дисципліни відводиться 128 годин, 4 кредити ЄКТС.

Форма навчання	Семестр	Всього кредитів/годин	Розподіл навчального часу за видами занять ¹					Семестрова атестація
			Лекції	Практичні заняття	Семінарські заняття	Лабораторні роботи	СРС	
Денна	2	4/120	32	-	-	32	56	екзамен

II. ІНФОРМАЦІЙНИЙ ОБСЯГ НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

1. Вимоги до створення ІТ-проектів
2. Реалізація принципів ООП в Delphi.
3. Методи у класах Delphi.
4. Успадкування. Поліморфізм.
5. Методи розв'язування систем нелінійних рівнянь (СНР).
6. Створення власної компоненти в Delphi.
7. Обробка переривань та виключних ситуацій.
8. Робота з файлами у Lazarus, Delphi.
2. Вивчення реалізації принципів ООП мовою програмування С++
9. Особливості ООП в мові С++.
10. Конструктори класу.
11. Конструктор копіювання. Деструктор.
12. Поліморфізм.
13. Успадкування.
14. Створення DLL.
15. Дружні класи і функції
16. Структури і об'єднання мови С++. Обробка виключних ситуацій.

III. ПЕРЕЛІК ЛАБОРАТОРНИХ РОБІТ

- 1 Ряд Фур'є у Embarcadero Delphi 10.2
- 2 Використання API-ф-цій у Delphi
- 3 OpenGL в Delphi
- 4 Перший проект ООП Sphere
- 5 Успадкування на основі класу Sphere
- 6 Успадкування у віконному режимі
- 7 Створення власної компоненти в Embarcadero Delphi 2010
- 8 Розробка власного проекту "Програмування ряду Фур'є по технології ООП"
- 9 Побудова класу Sphere у VC++ 2012
- 10 Успадкування на основі класу Sphere у VC++ 2012
- 11 Створення DLL у VC++ 2012
- 12 Успадкування VC++ 2012 (Managed)
- 13 Створення текстового редактора із панеллю інструментів у VC++2012

IV. РЕКОМЕНДОВАНА ЛІТЕРАТУРА

1. *Вайсфельд М.* Объектно-ориентированное мышление. – СПб.: Питер, 2014. – 304 с. (Серия «Библиотека программиста»).
2. *Буч Г.* Объектно-ориентированный анализ и проектирование с примерами на C++. – М.: Бином, 1998. – 560 с.
3. *Лафоре Р.* Объектно-ориентированное программирование в C++. Классика Computer Science. 4-е изд.– СПб.: Питер, 2005. – 924 с.
4. *Кравець П.О.* Об'єктно-орієнтоване програмування. Навч. посібн. – Львів: Видавн. Львівської політехніки, 2012. – 634 с.
5. *Казимир В.В.* Об'єктно-орієнтоване програмування . Навч. посібник. – К.: Видавн. Дім “Слово”, 2008. – 192 с.
6. *Просиз Дж.* Программирование для Microsoft .NET. – М.: Изд.-торг. дом “Русская Редакция”, 2003. – 704 с.
7. *Алексеев Е.Р. и др.* Free Pascal и Lazarus. Учебник по программированию. – М.: Альт Линукс, 2010. – 440 с.
8. *Рубанцев В.* Большой самоучитель Delphi XE3. Я+Р. 2013. – 1266 с.
9. *Осинов Д. Л.* Delphi. Программирование для Windows, OS X, iOS и Android. – СПб.: БХВ-Петербург, 2014. – 464 с.
10. *Осинов Д. Л.* Delphi XE2. – СПб.: БХВ-Петербург, 2012. – 912 с
11. *Фленов М. Е.* Библия Delphi. – СПб.: БХВ-Петербург, 2004. – 880 с.
12. *Павловская Т.А.* C/C++. Программирование на языке высокого уровня.– СПб.: Питер, 2007. – 461 с.
13. *Павловская Т.А., Щупак Ю.А.* C++. Объектно-ориентированное программирование: Практикум. – СПб.: Питер, 2005. – 265 с.
14. *Синтес А.* Освой самостоятельно объектно-ориентированное программирование за 21 день/ Пер. с англ. – М.: Изд. дом "Вильямс", 2002. – 672 с.
15. *Страуструп Б.* Язык программирования C++. – СПб.: Бином, 1999. – 991 с.
16. *Шилдт Г.* Искусство программирования на C++– СПб.: БХВ-Петербург, 2005. – 496 с.
17. *Шилдт Г.* Полный справочник по C++, 4-е изд./ Пер. с англ. – М.: Изд. дом "Вильямс", 2006. – 800 с.
18. *Шилдт Г.* C++: Базовый курс, 3-е изд./ Пер. с англ. – М.: Изд. дом "Вильямс", 2005. – 624с

V. ФОРМА ПІДСУМКОВОГО КОНТРОЛЮ УСПІШНОСТІ НАВЧАННЯ

Підсумкову оцінку якості засвоєння навчальної програми з дисципліни «**Об'єктно-орієнтоване програмування**» визначають за результатами екзамену, порядок проведення якого встановлює робоча навчальна програма.

Рівень засвоєння навчального матеріалу дисципліни визначають, використовуючи рейтингову систему оцінювання. Положення про рейтингову систему оцінювання знань розробляють та затверджується на засіданні кафедри з урахуванням особливостей професійної підготовки та розподілу навчального часу за видами занять. Це положення входить до складу робочої навчальної програми.

VI. Засоби діагностики успішності навчання

Оцінка якості засвоєння навчальної програми включає поточний контроль успішності, два модульних контролі та складання екзамену. Для поточного контролю засвоєння студентами навчального матеріалу передбачається виконання та захист 15-ти лабораторних робіт, перелік яких наводиться в робочій навчальній програмі.

Для модульних контролів засвоєння студентами навчального матеріалу передбачається виконання двох модульних контрольних робіт, порядок проведення та зміст яких наводяться в робочій навчальній програмі.

Для організації індивідуальної роботи студентів передбачається глибше опанування студентами лекційного матеріалу з використанням рекомендованих літературних джерел, матеріалів з Інтернету, виконання додаткових завдань, які є у методичних вказівках до виконання лабораторних робіт.