

МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ

Учебно-методическое объединение по образованию
в области информатики и радиоэлектроники

УТВЕРЖДАЮ

Первый заместитель Министра образования
Республики Беларусь

_____ А.Г.Баханович

Регистрационный № ТД-_____/тип.

ОБЪЕКТНО-ОРИЕНТИРОВАННОЕ ПРОГРАММИРОВАНИЕ

Типовая учебная программа по учебной дисциплине
для специальности

1-40 04 01 Информатика и технологии программирования

СОГЛАСОВАНО

Председатель Учебно-методического
объединения по образованию в
области информатики и
радиоэлектроники

_____ В.А. Богуш

СОГЛАСОВАНО

Начальник Главного управления
профессионального образования
Министерства образования
Республики Беларусь

СОГЛАСОВАНО

Проректор по научно-методической
работе Государственного учреждения
образования «Республиканский
институт высшей школы»

_____ И.В. Титович

Эксперт-нормоконтролер

Минск 2023

СОСТАВИТЕЛЬ:

А.В.Галковский, старший преподаватель кафедры информатики учреждения образования «Белорусский государственный университет информатики и радиоэлектроники»;

А.В.Жвакина, доцент кафедра информатики учреждения образования «Белорусский государственный университет информатики и радиоэлектроники», кандидат технических наук, доцент

РЕЦЕНЗЕНТЫ:

Кафедра управления информационными ресурсами Академии управления при Президенте Республики Беларусь (протокол № 2 от 08.02.2023);

И.П.Стацук, доцент кафедры цифровых систем и технологий Института бизнеса Белорусского государственного университета, кандидат технических наук, доцент

РЕКОМЕНДОВАНА К УТВЕРЖДЕНИЮ В КАЧЕСТВЕ ТИПОВОЙ:

Кафедрой информатики учреждения образования «Белорусский государственный университет информатики и радиоэлектроники» (протокол № 4 от 14.12.2022);

Научно-методическим советом учреждения образования «Белорусский государственный университет информатики и радиоэлектроники» (протокол № ___ от _____);

Научно-методическим советом по разработке программного обеспечения и информационно-коммуникационным технологиям Учебно-методического объединения по образованию в области информатики и радиоэлектроники (протокол № 6 от 13.02.2023)

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

ХАРАКТЕРИСТИКА УЧЕБНОЙ ДИСЦИПЛИНЫ

Типовая учебная программа по учебной дисциплине «Объектно-ориентированное программирование» разработана для студентов учреждений высшего образования, обучающихся по специальности 1-40 04 01 Информатика и технологии программирования в соответствии с требованиями образовательного стандарта высшего образования I степени и типового учебного плана вышеуказанной специальности.

Учебная дисциплина «Объектно-ориентированное программирование» знакомит студентов с концепциями объектно-ориентированного подхода, а также реализацией их в современных языках программирования и применением при разработке программ на этих языках. Знания и навыки, полученные при изучении курса, общепрофессиональными, формируют базовый уровень знаний инженера-системного программиста и необходимы для освоения других учебных дисциплин учебного плана специальности 1-40 04 01 Информатика и технологии программирования.

Воспитательное значение учебной дисциплины «Объектно-ориентированное программирование» заключается в формировании у студентов правильной культуры и стиля программирования и научного мировоззрения; развитии исследовательских умений, аналитических способностей, креативности, необходимых для решения научных и практических задач; развитии познавательных способностей и активности: творческой инициативы, самостоятельности, ответственности и организованности. Изучение данной учебной дисциплины способствует созданию условий для формирования интеллектуально развитой личности обучающегося, которой присущи стремление к профессиональному совершенствованию, активному участию в экономической и социально-культурной жизни страны, гражданская ответственность и патриотизм.

ЦЕЛЬ, ЗАДАЧИ УЧЕБНОЙ ДИСЦИПЛИНЫ

Цели учебной дисциплины: получение теоретических знаний и практических навыков в области современных технологий объектно-ориентированной разработки программ с использованием современных стандартов, подготовка к пониманию основных принципов объектно-ориентированного проектирования программных продуктов, получение информации о современных методах и средствах программирования; приобретение навыков практического применения полученных знаний для решения инженерных и научно-исследовательских задач.

Задачи учебной дисциплины:

освоение теоретических основ современных стандартов в объектно-ориентированном программировании;

изучение принципов проектирования и функционирования программных систем, основанных на объектах;

изучение основных концепций, применяемых в объектном подходе; отражение их в синтаксисе современных языков программирования;

изучение методов и приемов программирования, языковых средств объектно-ориентированного проектирования;

изучение интегрированных средств компоновки и отладки сложных проектов;

изучение концепций объектно-ориентированных языков высокого уровня C++, Java, Python и C#;

оформление представления о проектировании средних и больших проектов.

Базовыми учебными дисциплинами для учебной дисциплины «Объектно-ориентированное программирование» являются: «Основы алгоритмизации и программирования», «Программирование», «Конструирование программ». В свою очередь дисциплина «Объектно-ориентированное программирование» является базой для таких учебных дисциплин как «Операционные среды и системное программирование», «Технологии разработки программного обеспечения» (учебная дисциплина компонента учреждения высшего образования).

ТРЕБОВАНИЯ К УРОВНЮ ОСВОЕНИЯ СОДЕРЖАНИЯ УЧЕБНОЙ ДИСЦИПЛИНЫ

В результате изучения учебной дисциплины «Объектно-ориентированное программирование» формируются следующие компетенции:

универсальные:

владеть основами исследовательской деятельности, осуществлять поиск, анализ и синтез информации;

обладать навыками самосовершенствования и саморазвития в профессиональной деятельности;

проявлять инициативу и адаптироваться к изменениям в профессиональной деятельности;

базовая профессиональная:

применять концепции объектно-ориентированного программирования в различных языках программирования, применять методы и приемы построения объектных моделей реальных сущностей и процессов.

В результате изучения учебной дисциплины обучающийся должен:

знать:

технологии объектно-ориентированного программирования (ООП);

реализацию конструкций объектно-ориентированного программирования средствами процедурного программирования;

механизмы вызова виртуальных и интерфейсных методов;

принципы модульного и расширяемого программирования;

концепции объектно-ориентированного программирования в различных языках программирования;

объектно-ориентированные библиотеки, предназначенные для построения пользовательских интерфейсов;

технологии визуального программирования;

уметь:

разрабатывать программы с объектно-ориентированным дизайном;

разрабатывать надежные программы с защитой от исключительных ситуаций;

разрабатывать программы с использованием современных объектно-ориентированных библиотек;

разрабатывать программы в современных системах визуального программирования;

владеть:

методами и приемами построения объектных моделей реальных сущностей и процессов.

Типовая учебная программа рассчитана на 216 учебных часов, из них – 84 аудиторных. Примерное распределение аудиторных часов по видам занятий: лекции – 52 часа, лабораторные занятия – 32 часа.

ПРИМЕРНЫЙ ТЕМАТИЧЕСКИЙ ПЛАН

Наименование раздела, темы	Всего аудиторных часов	Лекции	Лабораторные занятия
Тема 1. Введение	2	2	-
Тема 2. Проблема сложности программных систем	4	4	-
Тема 3. Обзор современных объектно-ориентированных языков программирования	2	2	-
Тема 4. Язык UML. Использование диаграмм для описания программного проекта	4	4	-
Тема 5. Объектная модель. Реализация концепции абстракции в языках программирования	8	4	4
Тема 6. Применение объектов-контейнеров в объектно-ориентированном программировании	8	4	4
Тема 7. Реализация концепции иерархии для объектов и типов	6	4	2
Тема 8. Концепция инкапсуляции	8	4	4
Тема 9. Концепция полиморфизма	8	4	4
Тема 10. Типизация в языках программирования	6	4	2
Тема 11. Модульность в объектно-ориентированных языках программирования	6	4	2
Тема 12. Многопоточность в ООП	6	4	2
Тема 13. Постоянство объектов	8	4	4
Тема 14. Паттерны проектирования	8	4	4
Итого:	84	52	32

СОДЕРЖАНИЕ УЧЕБНОЙ ДИСЦИПЛИНЫ

Тема 1. ВВЕДЕНИЕ

Предмет курса, его задачи, структура. История развития языков программирования. Возникновение объектно-ориентированных языков. Понятие объекта и фундаментальные характеристики ООП.

Тема 2. ПРОБЛЕМА СЛОЖНОСТИ ПРОГРАММНЫХ СИСТЕМ

Жизненный цикл программного обеспечения (ПО). Понятия надежности и сложности ПО. Уменьшение сложности программной системы. Методы проектирования. Понятие жизненного цикла ПО: схема «водопада» и спиральная схема проектирования.

Тема 3. ОБЗОР СОВРЕМЕННЫХ ОБЪЕКТНО-ОРИЕНТИРОВАННЫХ ЯЗЫКОВ ПРОГРАММИРОВАНИЯ

Обзор и сравнительная характеристика языков программирования C++, C#, Python и Java.

Тема 4. ЯЗЫК UML. ИСПОЛЬЗОВАНИЕ ДИАГРАММ ДЛЯ ОПИСАНИЯ ПРОГРАММНОГО ПРОЕКТА

Язык UML как стандарт для описания проектов и документирования. Use-case диаграммы, диаграммы классов, временные диаграммы, диаграммы объектов, диаграммы состояний.

Тема 5. ОБЪЕКТНАЯ МОДЕЛЬ. РЕАЛИЗАЦИЯ КОНЦЕПЦИИ АБСТРАКЦИИ В ЯЗЫКАХ ПРОГРАММИРОВАНИЯ

Описание классов, абстрактных классов и интерфейсов. Объявление и определение методов класса. Создание и уничтожение объектов. Жизненный цикл объекта. Посылка сообщений.

Тема 6. ПРИМЕНЕНИЕ ОБЪЕКТОВ-КОНТЕЙНЕРОВ В ОБЪЕКТНО-ОРИЕНТИРОВАННОМ ПРОГРАММИРОВАНИИ

Назначение и роль объектов-контейнеров в приложениях. Особенности и реализация в языках C++, C# и Java.

Тема 7. РЕАЛИЗАЦИЯ КОНЦЕПЦИИ ИЕРАРХИИ ДЛЯ ОБЪЕКТОВ И ТИПОВ

Проектирование иерархии классов и объектов. Наследование свойств. Модификаторы доступа при наследовании и множественное наследование в C++. Особенности реализации наследования в языках C#, Python и Java. Виртуальные базовые классы в C++.

Тема 8. КОНЦЕПЦИЯ ИНКАПСУЛЯЦИИ

Соккрытие и защита данных. Интерфейс и реализация в описаниях классов. Реализация в языках C++, C#, Python и Java.

Тема 9. КОНЦЕПЦИЯ ПОЛИМОРФИЗМА

Перекрытие и перегрузка методов. Реализация в языках C++, C#, Python. Динамическая диспетчеризация в Java.

Тема 10. ТИПИЗАЦИЯ В ЯЗЫКАХ ПРОГРАММИРОВАНИЯ

Языки со слабой и сильной типизацией, статической и динамической типизацией. Динамическая типизация в языке программирования Python. Приведение типов. Определение типов объектов во время выполнения. Отражение (reflection) в языках C# и Java.

Тема 11. МОДУЛЬНОСТЬ В ОБЪЕКТОНО-ОРИЕНТИРОВАННЫХ ЯЗЫКАХ ПРОГРАММИРОВАНИЯ

Физическое и логическое разделение кода в проекте. Пакеты в Java, модули Python и пространства имен в C++, C#.

Тема 12. МНОГОПОТОЧНОСТЬ В ООП

Реализация концепции многопоточности в языках программирования. Стандартные классы и интерфейсы, отвечающие за реализацию потоков для отдельных объектов.

Тема 13. ПОСТОЯНСТВО ОБЪЕКТОВ

Сохранение состояния объектов. Поддержка сохранения объектов в библиотеках языков C++, C#, Python и Java.

Тема 14. ПАТТЕРНЫ ПРОЕКТИРОВАНИЯ.

Применение паттернов (шаблонов) проектирования при разработке объектно-ориентированных систем. Примеры паттернов «Одиночка», «Стратегия», «Состояние». Особенности и условия применения.

ИНФОРМАЦИОННО-МЕТОДИЧЕСКАЯ ЧАСТЬ

ЛИТЕРАТУРА

ОСНОВНАЯ

1. Объектно-ориентированный анализ и проектирование с примерами приложений / Г. Буч [и др.] – 3-е изд. – Москва : Вильямс, 2015. – 560 с.
2. Вайсфельд, М. Объектно-ориентированное мышление / М. Вайсфельд. – Санкт-Петербург : Питер, 2014. – 304 с.
3. Шилдт, Г. Java. Полное руководство / Г. Шилдт. – 12-е изд. – Санкт-Петербург : Диалектика, 2023. – 1344 с.
4. Парлог, Н. Система модулей Java / Н. Парлог. – Санкт-Петербург : Питер, 2021. – 464 с.
5. Гримм, Р. Параллельное программирование на современном C++ / Р. Гримм. – Москва : ДМК Пресс, 2022. – 616 с.
6. Лой, М. Програмируем на Java / М. Лой, П. Нимайер, Д. Лук. – 5-е межд. изд. – Санкт-Петербург : Питер, 2023. – 544 с.
7. Бизли, Д. Python. Исчерпывающее руководство / Д. Бизли. – Санкт-Петербург : Питер, 2023. – 368 с.
8. Седжвик, Р. Computer Science: основы программирования на Java, ООП, алгоритмы и структуры данных / Р. Седжвик, К. Уэйн. – Санкт-Петербург : Питер, 2018. – 1072 с.
9. Программирование на языке C++ в среде Qt Creator / Е. Р. Алексеев [и др.]. – Москва : ALT Linux, 2015. – 448 с.
10. Уильямс, Э. C++. Практика многопоточного программирования / Э. Уильямс. – Санкт-Петербург : Питер, 2020. – 801 с.
11. Прайс, М. C# 8 и .NET Core. Разработка и оптимизация / М. Прайс. – Санкт-Петербург : Питер, 2021. – 832 с.
12. Клири, С. Конкурентность в C#. Асинхронное, параллельное и многопоточное программирование / С. Клири. – 2-е межд. изд. – Санкт-Петербург : Питер, 2020. – 304 с.
13. Гамма, Э. Приемы объектно-ориентированного проектирования. Паттерны проектирования / Э. Гамма. – Санкт-Петербург : Питер, 2011. – 344 с.
14. Ларман, К. Применение UML 2.0 и шаблонов проектирования. Введение в объектно-ориентированный анализ и проектирование / К. Ларман. – 3-е изд. – Москва : Вильямс, 2012. – 620 с.

ДОПОЛНИТЕЛЬНАЯ

15. Задорожный, С. Объектно-ориентированное программирование на языке Python / С. Задорожный, Е. Фадеев. – Москва : МГУ, 2022. – 49 с.
16. Гайсарян, С. С. Объектно-ориентированные технологии проектирования прикладных программных систем [Электронный ресурс] / Центр Информационных технологий. – Режим доступа: http://citforum.ru/programming/oop_rsis/. – Дата доступа: 31.01.2023.

17. Миллет, С. Предметно-ориентированное проектирование: паттерны, принципы и методы / С. Миллет, Н. Тьюн. – Санкт-Петербург : Питер, 2017. – 367 с.
18. Эванс, Э. Предметно-ориентированное проектирование. Структуризация сложных программных систем / Э. Эванс. – Москва : Вильямс, 2010. – 832 с.
19. Dusty, Phillips. Python 3 Object-oriented Programming / P. Dusty. – Second Edition. – Packt Publishing, 2015. – 460 p.
20. Катупития, Я. Управление электронными устройствами на C++. Разработка практических приложений / Я. Катупития, К. Бенгли. – Москва : ДМК Пресс, 2016. – 442 с.
21. Фаулер, М. Asyncio и конкурентное программирование на Python / М. Фаулер. – Москва : ДМК Пресс, 2022. – 398 с.
22. Уорбэртон, Р. Лямбда-выражения в Java 8. Функциональное программирование – в массы / Р. Уорбэртон. – Москва : ДМК Пресс, 2014. – 192 с.
23. Хьюз, Дж. Структурный подход к программированию / Дж. Хьюз, Дж. Мичтом. – Москва : Мир, 1980. – 278 с.
24. Майерс, Г. Надежность программного обеспечения / Г. Майерс. – Москва : Мир, 1980. – 360 с.

МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ ПО ОРГАНИЗАЦИИ И ВЫПОЛНЕНИЮ САМОСТОЯТЕЛЬНОЙ РАБОТЫ СТУДЕНТОВ

При изучении учебной дисциплины рекомендуется использовать следующие формы самостоятельной работы:

- изучение технической документации;
- выполнение лабораторных работ;
- решение индивидуальных домашних заданий.

ПЕРЕЧЕНЬ РЕКОМЕНДУЕМЫХ СРЕДСТВ ДИАГНОСТИКИ КОМПЕТЕНЦИЙ СТУДЕНТА

Типовым учебным планом по специальности 1-40 04 01 Информатика и технологии программирования в качестве формы промежуточной аттестации по учебной дисциплине «Объектно-ориентированное программирование» рекомендуется экзамен и курсовой проект. Оценка учебных достижений студента производится по десятибалльной шкале.

Для текущего контроля по учебной дисциплине и диагностики компетенций студентов могут использоваться следующие формы:

- собеседования;
- контрольные опросы;
- тестирование;
- отчеты по аудиторным лабораторным работам с их устной защитой.

РЕКОМЕНДУЕМЫЕ МЕТОДЫ (ТЕХНОЛОГИИ) ОБУЧЕНИЯ

Основные рекомендуемые методы (технологии) обучения, отвечающие целям и задачам учебной дисциплины:

элементы проблемного обучения (проблемное изложение, частично-поисковый метод), реализуемые на лекционных и лабораторных занятиях;

элементы контролируемого обучения (контрольные опросы, контрольные работы), реализуемые на лабораторных занятиях, а также в ходе самостоятельной работы студентов.

РЕКОМЕНДАЦИИ ПО КУРСОВОМУ ПРОЕКТИРОВАНИЮ

Курсовые проекты выполняются в любой среде программирования на языках C++, Java или C#. Цель курсового проектирования: глубокое самостоятельное изучение пройденного материала.

Результатом выполнения курсового проекта является самостоятельно разработанное студентами программное обеспечение и пояснительная записка.

ПРИМЕРНЫЙ ПЕРЕЧЕНЬ ТЕМ КУРСОВЫХ ПРОЕКТОВ

1. Обучающая программа с тестированием знаний.
2. Редактор 3D с возможностью масштабирования и вращения трехмерных объектов.
3. Генератор уровней для игровой программы.
4. Инженерный калькулятор.
5. Текстовый редактор.
6. Графический редактор.
7. Игровая программа «Тетрис».
8. Игровая программа «Жизнь».
9. Игровая программа «Сапер».
10. Планетарий (программа для отображения звездного неба, звезд и планет).
11. Сетевая игра «Battle Sity Offline».
12. Приложение под Android «Книга рецептов».
13. Игра «Bomberman».
14. Логическая игра «Games of mind».
15. 2D-стратегия в реальном времени.
16. Эмулятор солнечной системы.
17. Интернет-магазин.
18. Игра в стиле шутер.
19. Игра «Street Battle Racing».
20. Приложение «Аудио-плеер».
21. Игра «3D Бильярд».
22. Игра под мобильную платформу iPhone.
23. Игра под мобильную платформу Android.

24. Игра под мобильную платформу Bada.
25. Игра под мобильную платформу Symbian.
26. Игра под мобильную платформу WinPhone7.
27. Программа djvu-reader для платформы Android.
28. Игра-аркада в стиле «Марио».

ПРИМЕРНЫЙ ПЕРЕЧЕНЬ ТЕМ ЛАБОРАТОРНЫХ ЗАНЯТИЙ

1. Создание и манипулирование объектом пользовательского типа. Обеспечение защиты данных.
2. Работа с контейнерами объектов.
3. Создание прототипа программы, содержащей 10-15 классов объектов.
4. Развитие прототипа программы, усложнение поведения классов.
5. Обеспечение сложного взаимодействия объектов.
6. Концепция постоянства объектов. Сохранение состояния программы.
7. Применение паттернов проектирования в проекте.

ПРИМЕРНЫЙ ПЕРЕЧЕНЬ КОМПЬЮТЕРНЫХ ПРОГРАММ

1. Персональный компьютер (ноутбук) на основе микропроцессора с архитектурой x64.
2. Среда программирования языков высокого уровня (VisualStudio / IntelliJ IDEA).
3. Программное обеспечение MicrosoftOffice или LibreOffice.