ЮЖНО-УРАЛЬСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

УТВЕРЖДАЮ:
Директор института
Высшая школа экономики и управления

ЮУрГУ
Электронный документ, подписанный ПЭП, хранится в системе электронного документооборота Южно-Уральского государственного университета
СВЕДЕНИЯ О ВЛАДЕЛЬЦЕ ПЭП
Кому выдан: Карпушкина А. В.
Пользователь: karpushkinaav
Дата подписания: 22.12.2021

А. В. Карпушкина

РАБОЧАЯ ПРОГРАММА

дисциплины 1.О.12.03 Объектно-ориентированное программирование
для направления 09.03.03 Прикладная информатика
уровень Бакалавриат
форма обучения очная
кафедра-разработчик Цифровая экономика и информационные технологии

Рабочая программа составлена в соответствии с ФГОС ВО по направлению подготовки 09.03.03 Прикладная информатика, утверждённым приказом Минобрнауки от 19.09.2017 № 922

Зав.кафедрой разработчика,
д.экон.н., доц.

ЮУрГУ
Электронный документ, подписанный ПЭП, хранится в системе электронного документооборота Южно-Уральского государственного университета
СВЕДЕНИЯ О ВЛАДЕЛЬЦЕ ПЭП
Кому выдан: Худякова Т. А.
Пользователь: khudiakovata
Дата подписания: 21.12.2021

Т. А. Худякова

Разработчик программы,
старший преподаватель

ЮУрГУ
Электронный документ, подписанный ПЭП, хранится в системе электронного документооборота Южно-Уральского государственного университета
СВЕДЕНИЯ О ВЛАДЕЛЬЦЕ ПЭП
Кому выдан: Конова Е. А.
Пользователь: konovaea
Дата подписания: 21.12.2021

Е. А. Конова

СОГЛАСОВАНО

Руководитель направления
к.техн.н., доц.

ЮУрГУ
Электронный документ, подписанный ПЭП, хранится в системе электронного документооборота Южно-Уральского государственного университета
СВЕДЕНИЯ О ВЛАДЕЛЬЦЕ ПЭП
Кому выдан: Прохорова И. А.
Пользователь: prokhorovaia
Дата подписания: 21.12.2021

И. А. Прохорова

Челябинск

1. Цели и задачи дисциплины

Основная цель изучения дисциплины – овладение теоретическими основами и инструментами объектно-ориентированного проектирования и программирования, формирование практических навыков по программированию решения экономических, вычислительных и других задач с использованием объектно-ориентированного подхода. Одной из составляющих является изучение инструментов и методов визуального программирования и практическое овладение объектной моделью среды программирования Visual Studio.Net. В результате успешного усвоения курса студент должен овладеть специальной подготовкой в предметной области и в области информационных технологий. Задача изучения дисциплины – реализация требований, установленных в квалификационной характеристике, а именно: 1) изучить теоретические основы объектно-ориентированного программирования; 2) овладеть инструментами и методами языка C++, реализующими объектный подход; 3) изучить инструменты визуального программирования и объектную модель среды программирования Visual Studio.Net; 4) получить навыки проектирования задач на основе объектного подхода и опыт разработки Windows приложений. В результате изучения дисциплины формируется теоретическая и методологическая база для цикла общепрофесиональных дисциплин, в которых изучается программирование и информационные технологии.

Краткое содержание дисциплины

В дисциплине «Объектно-ориентированное программирование » изучаются следующие разделы: • теоретические основы объектно-ориентированного проектирования и программирования; • инструменты и методы реализации объектной модели в языке программирования C++; • Net.Framework как интегрированная компонента ОС; • инструменты и методы визуального программирования; • объектная модель среды программирования Visual Studio.Net; • практическая разработка Windows приложений. В качестве языка изучения выбран язык Visual C++/CLI, в качестве среды разработчика Microsoft Visual Studio 2013, 2015.

2. Компетенции обучающегося, формируемые в результате освоения дисциплины

Планируемые результаты освоения ОП ВО (компетенции)	Планируемые результаты обучения по дисциплине
ОПК-2 Способен понимать принципы работы современных информационных технологий и программных средств, в том числе отечественного производства, и использовать их при решении задач профессиональной деятельности	Знает: Теоретические основы объектно-ориентированного проектирования и программирования, библиотеки классов, основные возможности современных интегрированных сред разработки программного обеспечения на объектно-ориентированных языках программирования, возможности компиляторов программных проектов под различные операционные системы, наборы инструкций для системных утилит автоматической сборки программного обеспечения и установки программных пакетов объектно-ориентированных библиотек и

	фреймворков Умеет: Использовать функциональные возможности современных интегрированных сред разработки программного обеспечения на объектно-ориентированных языках программирования для разработки прикладных программ, использовать утилиты автоматической сборки и развертывания программ в операционных системах Имеет практический опыт: Разработки программ на современных объектно-ориентированных языках, отладки и тестирования программного обеспечения с использованием современных интегрированных сред разработки.
ОПК-7 Способен разрабатывать алгоритмы и программы, пригодные для практического применения	Знает: Методы разработки алгоритмов и программ в рамках объектно-ориентированной парадигмы программирования на современном языке высокого уровня; принципы объектно-ориентированной парадигмы: абстрагирование, инкапсуляция, наследование, полиморфизм; основные синтаксические конструкции объектно-ориентированного языка программирования: классы, поля, свойства, методы, выражения, события; методы обобщенного программирования; методы оценки сложности алгоритмов; функциональные возможности стандартной библиотеки языка и фреймворка Умеет: Разрабатывать алгоритмы и программ в рамках объектно-ориентированной парадигмы на современном языке программирования высокого уровня с применением основных синтаксических конструкций и функциональных возможностей стандартной библиотеки языка и фреймворка Имеет практический опыт: Разработки алгоритмов и программ, отладки, поиска и устранения ошибок программного кода, оценки сложности алгоритмов, использования возможностей стандартной библиотеки, сторонних библиотек программного кода и фреймворков

3. Место дисциплины в структуре ОП ВО

Перечень предшествующих дисциплин, видов работ учебного плана	Перечень последующих дисциплин, видов работ
1.О.12.01 Основы программирования, 1.О.18 Пакеты прикладных программ, 1.О.12.02 Программирование на языках высокого уровня, 1.О.09 Информатика, Учебная практика, ознакомительная практика (2 семестр)	1.О.14 Операционные системы, 1.О.15 Вычислительные системы, сети и телекоммуникации, 1.О.13 Базы данных, Учебная практика, научно-исследовательская работа (получение первичных навыков научно-исследовательской работы) (4 семестр)

Требования к «входным» знаниям, умениям, навыкам студента, необходимым при освоении данной дисциплины и приобретенным в результате освоения предшествующих дисциплин:

Дисциплина	Требования
1.О.18 Пакеты прикладных программ	Знает: Виды технической документации предметной области, Классификацию и назначение различных категорий пакетов прикладных программ; состав и структуру пакетов; виды интерфейсов; возможности интеграции выбранных пакетов с другими программами. Умеет: Соотносить требования стандартов по оформлению документации с настройками объектов текстового документа., Выбирать пакеты программ в соответствии с типом задачи и имеющихся ресурсов и условий использования; создавать документы и шаблоны в среде выбранных пакетов Имеет практический опыт: Разработки шаблонов текстовых документов в соответствии с требованиями стандартов., Работы с пакетами прикладных программ для решения задач профессиональной области
1.О.12.01 Основы программирования	Знает: Основные конструкции языка программирования высокого уровня, основные компоненты современной среды программирования, Среды программирования для создания программ на языках высокого уровня, Основные структуры данных и алгоритмы их обработки Умеет: Проектировать программу, кодировать программу, осуществлять тестирование программы, а также отлаживать программу с использованием инструментов среды программирования, Устанавливать среду программирования, создавать и отлаживать программы в среде программирования, Разрабатывать алгоритмы и создавать программы на основе концепции структурного программирования Имеет практический опыт: Работы с современной средой программирования, проектирования и решения простых задач, Установки и использования среды программирования для решения профессиональных задач, Разработки алгоритмов и создания программ, а также использования встроенных структур данных языка программирования высокого уровня
1.О.12.02 Программирование на языках высокого уровня	Знает: Методы разработки алгоритмов и программ в рамках парадигмы структурного программирования на языке высокого уровня; основные синтаксические конструкции языка программирования высокого уровня: операторы, выражения, блоки, ветвления, циклы; методы оценки сложности алгоритмов; функциональные возможности стандартной библиотеки языка

	высокого уровня, Возможности современных языков программирования, парадигмы программирования, библиотеки алгоритмов и классов, основные возможности современных интегрированных сред разработки программного обеспечения на языках высокого уровня, возможности компиляторов и компоновщиков под различные операционные системы, наборы инструкций для системных утилит автоматической сборки программ. Умеет: Разрабатывать алгоритмы и программ в рамках парадигмы структурного программирования на языке программирования высокого уровня с применением основных синтаксических конструкций и функциональных возможностей стандартной библиотеки языка высокого уровня, Использовать функциональные возможности современных интегрированных сред разработки программного обеспечения на языках высокого уровня для разработки прикладных программ, использовать утилиты автоматической сборки и развертывания программ в операционных системах. Имеет практический опыт: Разработки алгоритмов и программ, отладки, поиска и устранения ошибок программного кода, оценки сложности алгоритмов, использования возможностей стандартной библиотеки, сторонних библиотек программного кода, Работы с основными современными интегрированными средами разработки программного обеспечения на языках высокого уровня, разработки, отладки и тестирования разработанных программ.
1.О.09 Информатика	Знает: Базовые понятия информационной безопасности, классификацию угроз, требования к формированию паролей, Возможности современного программного обеспечения для подготовки текстовой документации., Состав, назначение функциональных компонентов и программного обеспечения персонального компьютера, в том числе отечественного производства Умеет: Выбирать необходимую защиту данных для текстовых документов и файлов электронных таблиц, Использовать возможности программного обеспечения для настройки оформления в соответствии с нормативными требованиями., Использовать программные и аппаратные средства персонального компьютера, применять типовые программные средства сервисного назначения, выбирать современные информационные технологии и программные средства, в том числе отечественного производства при решении задач профессиональной деятельности Имеет практический опыт: Применения современных программных средств для наглядного представления и структурирования информации

	с учетом требований информационной безопасности, Использования стандартов, норм и правил наглядного представления структурированной информации, Применения современных информационных технологий и программных средств при решении задач профессиональной деятельности
Учебная практика, ознакомительная практика (2 семестр)	Знает: Основные приемы эффективного управления собственным временем., Методы математического анализа и моделирования, теоретического и экспериментального исследования в профессиональной деятельности, Принципы работы современных информационных технологий и программных средств., Современные справочные ресурсы в профессиональной деятельности., Требования к организации рабочего места при использовании вычислительной техники. , Основные технологии межличностной и групповой коммуникации в деловом взаимодействии Умеет: Планировать своё время на основе анализа сложности и объема поставленных задач., Применять знания математических и естественно-научных дисциплин при разработке алгоритмов решения практических задач., Умеет выбирать программные средства и технологии для реализации практических задач с учетом имеющихся ресурсов., Осуществлять поиск необходимой информации, использовать информационные ресурсы при решении типовых задач программирования., Анализировать условия работы и организовывать рабочее место., Устанавливать и поддерживать контакты, обеспечивающие успешную работу в коллективе; применять основные методы и нормы социального взаимодействия для реализации своей роли и взаимодействия внутри команды Имеет практический опыт: Распределения задач и составления плана работы на заданный промежуток времени., Составления алгоритмов с применением базовых понятий математики., Использования доступных технологий и программных средств для решения поставленных задач., Работы со справочными ресурсами при выполнении заданий практики., Создания и поддержания в повседневной жизни и в профессиональной деятельности безопасных условий жизнедеятельности , Простейшими методами и приемами социального взаимодействия и работы в команде.

4. Объём и виды учебной работы

Общая трудоемкость дисциплины составляет 4 з.е., 144 ч., 74,5 ч. контактной работы

Вид учебной работы	Всего часов	Распределение по семестрам в часах
		Номер семестра
		3
Общая трудоёмкость дисциплины	144	144
Аудиторные занятия:	64	64
Лекции (Л)	32	32
Практические занятия, семинары и (или) другие виды аудиторных занятий (ПЗ)	32	32
Лабораторные работы (ЛР)	0	0
Самостоятельная работа (СРС)	69,5	69,5
с применением дистанционных образовательных технологий	0	
Выполнение заданий на самостоятельную разработку	19,5	19.5
Изучение теоретического материала, подготовка к практическим занятиям	12	12
Подготовка к экзамену	38	38
Консультации и промежуточная аттестация	10,5	10,5
Вид контроля (зачет, диф.зачет, экзамен)	-	экзамен

5. Содержание дисциплины

№ раздела	Наименование разделов дисциплины	Объем аудиторных занятий по видам в часах			
		Всего	Л	ПЗ	ЛР
1	Теоретические основы объектно-ориентированного проектирования и программирования.	4	2	2	0
2	Принципы объектно-ориентированного программирования. Инструменты реализации объектной модели в языке C++. Абстрактные типы данных – классы. Инкапсуляция, наследование, полиморфизм.	28	14	14	0
3	Введение в Microsoft .NET Framework. Язык C++/CLI. Разработка приложений на основе шаблона Windows Form.	8	4	4	0
4	Net Framework: основы визуального программирования. Элементы управления: свойства, события. Технология разработки Windows приложения и требования к интерфейсу.	12	6	6	0
5	Библиотека классовNet Framework. Базовые объекты: array, String, object и другие.	12	6	6	0

5.1. Лекции

№ лекции	№ раздела	Наименование или краткое содержание лекционного занятия	Кол-во часов
1	1	Теоретические основы объектно-ориентированного проектирования и программирования. Эволюция в области технологий программирования. Основные подходы к разработке программ – модульный и объектный стили.	2

		Принципы объектного подхода к проектированию и разработке программ: – абстракция, инкапсуляция, наследование, полиморфизм. Расширение базовых понятий – типизация, параллелизм, сохраняемость. Объектные модели задач предметной области	
2	2	Основы объектно-ориентированного проектирования и программирования. Знакомство с инструментами реализации объектно-ориентированного подхода: абстрактные типы данных – классы. Инкапсуляция: описание объектного типа данных, переменные объектного типа. Статические и динамические объекты. Порождение и уничтожение объектов – конструкторы и деструкторы.	2
3	2	Перегрузка операций. Конструктор копирования.	2
4	2	Наследование: общая концепция, реализация механизмов наследования. Виды наследования, построение иерархий.	2
5	2	Наследование: контейнерное наследование: реализация. Примеры использования.	2
6	2	Полиморфизм: виртуальные функции и абстрактные классы, примеры реализации.	2
7	2	Параметрический полиморфизм: шаблоны функций и классов. Краткое введение в обобщенное программирование.	2
8	2	Библиотека STL как пример библиотеки шаблонов. Использование классов STL для решения прикладных задач.	2
9	3	Платформа Microsoft .NET Framework: роль и назначение, составные части платформы .Net, общая концепция языков и спецификация CLI, общая система типов .Net. Управляемые и неуправляемые приложения: знакомство с языковыми средствами C++/CLI, сборка мусора (garbage collected) и дескрипторы объектов. Пространства имен.	2
10	3	Разработка приложений на основе шаблона Windows Form. Основы визуального программирования. Знакомство с основными элементами управления как объектами. Свойства и события основных элементов, кодирование обработчиков событий.	2
11	4	Технология разработки SDI приложения – виды окон, разработка основных элементов интерфейса – меню, панели инструментов, строка статуса.	2
12	4	Требования к интерфейсу Windows приложения. Использование нескольких окон, окна диалога, стандартные диалоги.	2
13	4	Работа с файлами данных. Элементы управления для отображения данных. Класс Convert.	2
14	5	Библиотека классов Net Framework. Класс array как базовый класс для реализации структур последовательного хранения данных. Разработка интерфейса массива на Grid'e.	2
15	5	Библиотека классов Net Framework. Класс string для обработки строк, элементы визуализации текста. Конвертирование данных к текстовому представлению.	2
16	5	Обзор объектной модели Net Framework. Коллекции. Классы интерфейсы.	2

5.2. Практические занятия, семинары

№ занятия	№ раздела	Наименование или краткое содержание практического занятия, семинара	Кол-во часов
1	1	Класс как абстрактный тип данных. Спецификация классов, порождение статических и динамических объектов класса, обращение к данным и методам объекта класса. Конструкторы и деструкторы.	2
2	2	Класс как абстрактный тип данных. Инкапсуляция. Перегрузка операций.	2

3	2	Инкапсуляция. Реализация абстрактного типа данных «массив», некоторые алгоритмы управления массивом, реализация конструктора и деструктора массива.	2
4	2	Наследование. Реализация механизма наследования. Особенности конструкторов производных классов.	2
5	2	Наследование. Реализация механизма наследования, виды наследования – множественное, косвенное наследование.	2
6	2	Наследование. Реализация механизма контейнерного наследования.	2
7	2	Реализация идеи полиморфизма. Виртуальные функции и механизм динамического связывания. Абстрактные классы	2
8	2	Параметрический полиморфизм. Шаблоны функций и классов.	2
9	3	Основы визуального программирования. Знакомство с основными элементами управления как объектами. Свойства и события основных элементов, кодирование обработчиков событий. Программное управление поведением компонентов и объектов, разработка обработчиков событий. Обработка событий клавиатуры и мыши	2
10	3	Основы визуального программирования. Место объекта пользователя в проекте. Разработка управляемого приложения с использованием объекта пользователя и интерфейсов на основе элементов управления. Разработка интерфейса объекта на примере реализации анимации.	2
11	4	Основы визуального программирования. Разработка интерфейса объекта на примере реализации анимации. Использование различных элементов управления для изменения состояния объекта.	2
12	4	Требования к интерфейсу Windows приложения. Разработка основных элементов интерфейса: меню, панели инструментов, строка состояния.	2
13	4	Изучение базовых типов данных: класс array как шаблон последовательных структур хранения данных. Разработка интерфейса массива на Grid'e.	2
14	5	Изучение базовых типов данных: класс string для обработки строк, элементы визуализации текста. Конвертирование данных к текстовому представлению.	2
15	5	Работа с файлами данных. Элементы управления для отображения данных. Конвертирование данных: класс Convert.	2
16	5	Контрольный срез	2

5.3. Лабораторные работы

Не предусмотрены

5.4. Самостоятельная работа студента

Выполнение СРС			
Подвид СРС	Список литературы (с указанием разделов, глав, страниц) / ссылка на ресурс	Семестр	Кол-во часов
Выполнение заданий на самостоятельную разработку	ПУМД, по разделам, ЭУМД, по разделам	3	19,5
Изучение теоретического материала, подготовка к практическим занятиям	ПУМД, по разделам.	3	12
Подготовка к экзамену	ПуМД и ЭУМД, по разделам	3	38

6. Текущий контроль успеваемости, промежуточная аттестация

Контроль качества освоения образовательной программы осуществляется в соответствии с Положением о балльно-рейтинговой системе оценивания результатов учебной деятельности обучающихся.

6.1. Контрольные мероприятия (КМ)

№ КМ	Се-местр	Вид контроля	Название контрольного мероприятия	Вес	Макс. балл	Порядок начисления баллов	Учи-тыва-ется в ПА
1	3	Текущий контроль	Практическая работа 01. Инкапсуляция	1	6	Баллы начисляются за часть задания, в котором необходимо выполнить задание на самостоятельную разработку. Соблюдение принципов объектного подхода - 3 б., владение инструментами языка - 3б. Код должен быть отлажен и протестирован.	экзамен
2	3	Текущий контроль	Практическая работа 04. Наследование	1	1	Баллы начисляются за часть задания, в котором необходимо выполнить задание на самостоятельную разработку. Соблюдение принципов объектного подхода - 3 б., владение инструментами языка - 3б. Код должен быть отлажен и протестирован.	экзамен
3	3	Текущий контроль	Практическая работа 06. Визуальная разработка	1	7	Баллы начисляются за часть задания, в котором необходимо выполнить задание на самостоятельную разработку. Соблюдение принципов объектного подхода - 3 б., владение инструментами языка - 2б, разработка формы - 2б.. Код должен быть отлажен и протестирован.	экзамен
4	3	Текущий контроль	Практическая работа 07. Визуальная разработка: элементы управления на форме.	1	7	Соблюдение принципов объектного подхода - 3 б., владение инструментами языка - 4б. Код должен быть отлажен и протестирован.	экзамен
5	3	Текущий контроль	Практическая работа 08. Контейнерные классы на основе шаблона array	1	7	Баллы начисляются за часть задания, в котором необходимо выполнить задание на самостоятельную разработку. Соблюдение принципов объектного подхода - 3 б., владение инструментами языка - 4б. Код должен быть отлажен и протестирован.	экзамен
6	3	Текущий контроль	Практическая работа 09. Работа с текстом: тип String	1	7	Баллы начисляются за часть задания, в котором необходимо выполнить задание на самостоятельную разработку. Соблюдение принципов объектного подхода - 3 б., владение инструментами языка - 4б. Код должен быть отлажен и протестирован.	экзамен
7	3	Текущий	Практическая	1	7	Баллы начисляются за часть задания, в	экзамен

		контроль	работа 09. Полиморфизм. Абстрактные классы			котором необходимо выполнить задание на самостоятельную разработку. Соблюдение принципов объектного подхода - 3 б., владение инструментами языка - 4б. Код должен быть отлажен и протестирован.	
8	3	Текущий контроль	Практическая работа 10. Интерфейсы как класс.	1	7	Баллы начисляются за часть задания, в котором необходимо выполнить задание на самостоятельную разработку. Соблюдение принципов объектного подхода - 3 б., владение инструментами языка - 4б. Код должен быть отлажен и протестирован.	экзамен
9	3	Текущий контроль	Практическая работа 11. Прикладные задачи: объектная модель.	1	8	Необходимо выполнить задание на самостоятельную разработку. Соблюдение принципов объектного подхода при построении модели - 3 б., владение инструментами языка - 3б, владение инструментами разработки интерфейса - 2б.	экзамен
10	3	Проме-жуточная аттестация	Промежуточная аттестация по дисциплине ООП. Тип КМ - экзамен.	-	40	Тестирование по теоретической части курса содержит 40 вопросов, время выполнения - 60 мин. Практическая часть содержит задание на самостоятельную разработку, подобное тому, что дано в "Практическая работа 11". Время выполнения - 90 мин. При оценивании результатов мероприятия используется балльно-рейтинговая система оценивания результатов учебной деятельности обучающихся (утверждена приказом ректора от 24.05.2019 г. № 179). По результатам тестирования высший балл 40. По по решению задачи высший балл 20 : оценивается качество объектной модели - 5б, разработка объектов в соответствии с принципами ООП - 5., владение инструментами языка - 5б., владение инструментами визуальной разработки - 5б. Экзамен считается завершенным, если по совокупности баллов студент набрал не мене 60 % общего рейтинга обучающегося, в ином случае студент направляется на пересдачу. Баллы конвертируются в традиционную оценку по следующей шкале: Отлично: величина рейтинга обучающегося по дисциплине 85…100% Хорошо: величина рейтинга обучающегося по дисциплине 75…84 % Удовлетворительно: величина	экзамен

						рейтинга обучающегося по дисциплине 60…74 % Неудовлетворительно: величина рейтинга обучающегося по дисциплине 0…59 %	

6.2. Процедура проведения, критерии оценивания

Вид промежуточной аттестации	Процедура проведения	Критерии оценивания
экзамен	Тестирование на проверку сформированности компетенций по теоретической части дисциплины. Выполнение интегрированного практического задания на проверку сформированности умений и навыков построения объектных моделей, знания инструментов языка, умения кодировать, отлаживать код.	В соответствии с пп. 2.5, 2.6 Положения

6.3. Оценочные материалы

Компетенции	Результаты обучения	№ КМ									
		1	2	3	4	5	6	7	8	9	10
ОПК-2	Знает: Теоретические основы объектно-ориентированного проектирования и программирования, библиотеки классов, основные возможности современных интегрированных сред разработки программного обеспечения на объектно-ориентированных языках программирования, возможности компиляторов программных проектов под различные операционные системы, наборы инструкций для системных утилит автоматической сборки программного обеспечения и установки программных пакетов объектно-ориентированных библиотек и фреймворков	+	+	+	+	+	+	+	+	+	+
ОПК-2	Умеет: Использовать функциональные возможности современных интегрированных сред разработки программного обеспечения на объектно-ориентированных языках программирования для разработки прикладных программ, использовать утилиты автоматической сборки и развертывания программ в операционных системах	+	+	+	+	+	+	+	+	+	+
ОПК-2	Имеет практический опыт: Разработки программ на современных объектно-ориентированных языках, отладки и тестирования программного обеспечения с использованием современных интегрированных сред разработки.			+	+		+	+	+	+	+
ОПК-7	Знает: Методы разработки алгоритмов и программ в рамках объектно-ориентированной парадигмы программирования на современном языке высокого уровня; принципы объектно-ориентированной парадигмы: абстрагирование, инкапсуляция, наследование, полиморфизм; основные синтаксические конструкции объектно-ориентированного языка программирования: классы, поля, свойства, методы, выражения, события; методы обобщенного программирования; методы оценки сложности алгоритмов; функциональные возможности стандартной библиотеки языка и фреймворка	+	+	+	+	+	+	+	+	+	+
ОПК-7	Умеет: Разрабатывать алгоритмы и программ в рамках объектно-ориентированной парадигмы на современном языке программирования высокого уровня с применением основных	+	+	+	+	+	+		+	+	+

	синтаксических конструкций и функциональных возможностей стандартной библиотеки языка и фреймворка									
ОПК-7	Имеет практический опыт: Разработки алгоритмов и программ, отладки, поиска и устранения ошибок программного кода, оценки сложности алгоритмов, использования возможностей стандартной библиотеки, сторонних библиотек программного кода и фреймворков			+		+	+	+	+	+

Фонды оценочных средств по каждому контрольному мероприятию находятся в приложениях.

7. Учебно-методическое и информационное обеспечение дисциплины

Печатная учебно-методическая документация

а) основная литература:

1. Подбельский, В. В. Язык Си++ [Текст] учеб. пособие для вузов по направлениям "Приклад. математика" и "Вычисл. машины, комплексы, системы и сети" В. В. Подбельский. - 5-е изд. - М.: Финансы и статистика, 2004. - 559 с.

б) дополнительная литература:

1. Павловская, Т. А. C/C++: Программирование на языке высокого уровня учебник для вузов по направлению "Информатика и вычисл. техника" Текст Т. А. Павловская. - СПб. и др.: Питер, 2008. - 460 с.
2. Павловская, Т. А. C++ : Объектно-ориентированное программирование. Практикум Текст учеб. пособие для вузов по направлению "Информатика и вычисл. техника" Т. А. Павловская, Ю. А. Щупак. - СПб. и др.: Питер, 2008. - 264 с. ил.

в) отечественные и зарубежные журналы по дисциплине, имеющиеся в библиотеке:

1. 1. Программирование. Научн. журн. Российской академии наук. ISSN 0132-3474.
2. 3. RSDN Magazine.
3. Электронная версия http://rsdn.ru/.

г) методические указания для студентов по освоению дисциплины:

1. Программирование в объектах C++

из них: учебно-методическое обеспечение самостоятельной работы студента:

1. Программирование в объектах C++

Электронная учебно-методическая документация

№	Вид литературы	Наименование ресурса в электронной форме	Библиографическое описание
1	Основная литература	Электронный каталог ЮУрГУ	Конова Е. А. Объектно-ориентированное программирование с примерами на C++ [Текст] : учеб. пособие по направлениям 09.03.02 "Приклад. информатика" и 09.03.02 "Информ. системы и технологии" / Е. А. Конова ; под ред. Б. М. Суховилова ; Юж.-Урал.

			гос. ун-т, Каф. Информ. технологии в экономике ; Челябинск : Издательский Центр ЮУрГУ , 2019 http://www.lib.susu.ac.ru/ftd?base=SUSU_METHOD&key=000563302
2	Дополнительная литература	Электронный каталог ЮУрГУ	Радченко Г.И. Объектно-ориентированное программирование [Текст] : конспект лекций для направлений 010300 "Фундам. информатика и информ. технологии" и 010400 "Приклад. математика и информатика"/ Г. И. Радченко, Е. А. Захаров; ЮУрГУ, Каф. Систем. программирование ; ЮУрГУ. Челябинск : Издательский Центр ЮУрГУ , 2013 http://www.lib.susu.ac.ru/ftd?base=SUSU_METHOD&key=000515626

Перечень используемого программного обеспечения:

1. Microsoft-Office(бессрочно)
2. Microsoft-Visual Studio(бессрочно)

Перечень используемых профессиональных баз данных и информационных справочных систем:

1. ООО "ГарантУралСервис"-Гарант(бессрочно)

8. Материально-техническое обеспечение дисциплины

Вид занятий	№ ауд.	Основное оборудование, стенды, макеты, компьютерная техника, предустановленное программное обеспечение, используемое для различных видов занятий
Практические занятия и семинары	258 (3б)	Специализированный компьютерный класс каф. ЦЭиИТ, среда разработчика Visual Studio.Net, доступ к справочной системе.
Лекции	229 (3б)	Компьютер, проектор, среда разработчика Visual Studio.Net, доступ к справочной системе.
Контроль самостоятельной работы	258 (3б)	Специализированный компьютерный класс каф. ЦЭиИТ, среда разработчика Visual Studio.Net, доступ к справочной системе.