

**Министерство науки и высшего образования Российской Федерации**  
**Федеральное государственное бюджетное образовательное учреждение**  
**высшего образования**  
**«Владимирский государственный университет**  
**имени Александра Григорьевича и Николая Григорьевича Столетовых»**  
**(ВлГУ)**

Институт прикладной математики, физики и информатики  
(Наименование института)

УТВЕРЖДАЮ:

Директор института

К.С. Хорьков

2021 г.



**РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ**  
**ОБЪЕКТНО-ОРИЕНТИРОВАННОЕ ПРОГРАММИРОВАНИЕ**  
(наименование дисциплины)

**направление подготовки / специальность**  
**02.03.02 «Фундаментальная информатика и информационные**  
**технологии»**

(код и наименование направления подготовки (специальности))

**направленность (профиль) подготовки**

**Мобильные и Интернет-технологии**

(направленность (профиль) подготовки)

г. Владимир

2021 г.

## 1. ЦЕЛИ ОСВОЕНИЯ ДИСЦИПЛИНЫ

Целью освоения дисциплины «Объектно-ориентированное программирование» является получение студентами знаний в области объектно-ориентированного программирования, навыков работы в визуальной среде разработки.

Задачи:

- изучение студентами основных алгоритмических структур;
- изучение алгоритмов обработки данных, в том числе алгоритмов поиска и упорядочивания;
- получение студентами навыков оценки сложности алгоритма;
- изучение студентами парадигм программирования;
- изучение синтаксических основ современного языка программирования высокого уровня C++;
- изучение способов машинного представления данных,
- получение навыков тестирования программного кода.

## 2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ОПОП

Дисциплина «Объектно-ориентированное программирование» относится к дисциплинам обязательной части блока Б1 дисциплины (модули) учебного плана.

## 3. ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ ОБУЧЕНИЯ ПО ДИСЦИПЛИНЕ

Планируемые результаты обучения по дисциплине, соотнесенные с планируемыми результатами освоения ОПОП (компетенциями и индикаторами достижения компетенций)<sup>1)</sup>

Формируемые компетенции (код, содержание компетенции)	Планируемые результаты обучения по дисциплине, в соответствии с индикатором достижения компетенции		Наименование оценочного средства
	Индикатор достижения компетенции (код, содержание индикатора)	Результаты обучения по дисциплине	
ОПК-2. Способен применять компьютерные/суперкомпьютерные методы, современное программное обеспечение, в том числе отечественного происхождения, для решения задач профессиональной деятельности	ОПК-2.1. Знает основные положения, концепции и терминологию в области программирования, знаком с содержанием Единого реестра российских программ. ОПК-2.2. Умеет осуществлять обоснованный выбор компьютерных/суперкомпьютерных методов и необходимого программного обеспечения при решении задач профессиональной деятельности. ОПК-2.3. Владеет навыками применения компьютерных/суперкомпьютерных методов и программного обеспечения при решении конкретных задач	<b>Знать:</b> основные положения и концепции в области программирования, архитектуру языков программирования, основную терминологию в области программного обеспечения. <b>Уметь:</b> осуществлять обоснованный выбор компьютерных методов решения и необходимого программного обеспечения при решении задач профессиональной деятельности. <b>Владеть:</b> навыками применения компьютерные/суперкомпьютерные методы методов и программного обеспечения при решении конкретных задач.	Отчёты по лабораторным работам, Контрольные вопросы к лабораторным работам, вопросы к промежуточной и текущей аттестации.
ОПК-3. Способен к разработке алгоритмических и программных решений в области системного и прикладного	ОПК-3.1. Знает методы теории алгоритмов, системного и прикладного программирования, принципы и методологии тестирования программного обеспечения, принципы математического моделирования,	<b>Знает:</b> методы теории алгоритмов <b>Умеет:</b> разрабатывать алгоритмические решения, представлять решение в виде блок-схемы и на языке программирования	Отчёты по лабораторным работам, Контрольные вопросы к лабораторным работам,

<p>программирования, математических, информационных и имитационных моделей, созданию информационных ресурсов глобальных сетей, образовательного контента, прикладных баз данных, тестов и средств тестирования систем и средств на соответствие стандартам и исходным требованиям</p>	<p>типичные (универсальные) математические (включая информационные и имитационные) модели, формулы, теоремы и методы, используемые в широком наборе областей применения прикладной математики. ОПК-3.2. Умеет определять и составлять информационные ресурсы глобальных сетей, образовательного контента, средств тестирования систем, осуществлять обоснованный выбор адекватных поставленной задаче базовых математических моделей, модифицировать базовые и (или) разрабатывать оригинальные математические модели в соответствии со спецификой поставленной задачи моделирования. ОПК-3.3. Владеет навыками разработки программного обеспечения, а также выполнения математического моделирования от анализа постановки задачи до анализа результатов.</p>	<p>высокого уровня. <b>Владеет:</b> Навыками составления блок-схем, написания кода по алгоритмическому решению, представленному в виде юлок схемы.</p>	<p>вопросы к промежуточной и текущей аттестации.</p>
<p>ОПК-2. Способен применять компьютерные/суперкомпьютерные методы, современное программное обеспечение, в том числе отечественного происхождения, для решения задач профессиональной деятельности</p>	<p>ОПК-2.1. Знает основные положения, концепции и терминологию в области программирования, знаком с содержанием Единого реестра российских программ. ОПК-2.2. Умеет осуществлять обоснованный выбор компьютерных/суперкомпьютерных методов и необходимого программного обеспечения при решении задач профессиональной деятельности. ОПК-2.3. Владеет навыками применения компьютерных/суперкомпьютерных методов и программного обеспечения при решении конкретных задач</p>	<p><b>Знает:</b> Знает модели жизненного цикла информационных систем <b>Умеет:</b> осуществлять выбор базовых математических моделей соответствующих поставленной задаче. <b>Владеет:</b> навыками документирования процесса создания информационных систем.</p>	<p>Отчёты по лабораторным работам, Контрольные вопросы к лабораторным работам, вопросы к промежуточной и текущей аттестации.</p>
<p>ПК-1. Способен разрабатывать требования и проектировать программное обеспечение</p>	<p>ПК-1.1. Знает методики разработки требований к системе, методы классического системного анализа, стандарты оформления технических заданий, методологии разработки программного обеспечения и технологии программирования, языки формализации функциональных спецификаций, принципы построения архитектуры программного обеспечения и виды архитектуры программного обеспечения, основные концепции и атрибуты качества программного обеспечения. ПК-1.2. Умеет выбирать методики разработки требований к системе и шаблоны документов требований к</p>	<p><b>Знает:</b> возможности современных перспективных средств разработки программных продуктов <b>Умеет:</b> обоснованно выбирать средства и варианты реализации программного обеспечения. <b>Владеет:</b> навыками планирования работ по разработке требований к системе, анализа проблемной ситуации, навыками оценки и согласования сроков выполнения поставленных</p>	<p>Отчёты по лабораторным работам, Контрольные вопросы к лабораторным работам, вопросы к промежуточной и текущей аттестации.</p>

	<p>системе, определять источники информации для требований к системе, выполнять тестирование системы с целью проверки её реализации на соответствие требованиям, формулировать и оформлять запросы на изменение требований, проводить анализ исполнения требований, вырабатывать варианты их реализации, проводить оценку и обоснование рекомендуемых решений, выбирать и использовать средства и варианты реализации программного обеспечения.</p> <p>ПК-1.3. Владеть навыками планирования работ по разработке требований к системе, анализа проблемной ситуации, согласования целей создания системы с заинтересованными лицами, оформления технического задания на систему, представления концепции, технического задания и изменений в них заинтересованным лицам, оценки возможностей, времени и трудоемкости реализации требований к программному обеспечению, разработки и согласования технических спецификаций на программное обеспечение, формирования и предоставления отчетности в соответствии с установленными регламентами, проектирования структур данных, баз данных, программных интерфейсов.</p>	задач.	
ПК-2. Способен анализировать требования к программному средству	<p>ПК-2.1. Знает методы анализа и тестирования требований к программному средству, методы анализа, проектирования и разработки программного обеспечения.</p> <p>ПК-2.2. Умеет тестировать требования к программному средству, оформлять документацию по тестированию, анализировать требования на соответствие принятым стандартам и методам проектирования, использовать современные CASE-средства.</p> <p>ПК-2.3. Владеет навыками проверки осуществимости функционирования и сопровождения программного средства, определения возможности введения изменений и дополнений требований к программному средству.</p>	<p><b>Знает:</b> методы анализа, проектирования и разработки программного обеспечения.</p> <p><b>Умеет:</b> использовать современные CASE-средства.</p> <p><b>Владеет:</b> навыками проверки функционирования программного продукта.</p>	<p>Отчёты по лабораторным работам, Контрольные вопросы к лабораторным работам, вопросы к промежуточной и текущей аттестации.</p>

#### 4. ОБЪЕМ И СТРУКТУРА ДИСЦИПЛИНЫ

Трудоемкость дисциплины составляет   6   зачетных единиц,   216   часов

**Тематический план  
форма обучения – очная, ускоренное обучение**

№ п/п	Наименование тем и/или разделов/тем дисциплины	Семестр	Неделя семестра	Контактная работа обучающихся с педагогическим работником				Самостоятельная работа	Формы текущего контроля успеваемости, форма промежуточной аттестации (по семестрам)
				Лекции	Практические занятия	Лабораторные работы	в форме практической подготовки		
1	<b>Раздел 1.</b> Принципы объектно-ориентированного программирования	2	1-12	24	-	24	10	102	Зачёт (переаттестация) Рейтинг-контроль №1 Рейтинг-контроль №2
2	<b>Раздел 2.</b> Работа в визуальной среде проектирования.	2	13-18	12	-	10	10	30	Рейтинг-контроль №3
3	Подведение итогов курса	2	18	-	-	-	-	12	Рейтинг-контроль №3
Всего за _2_ семестр:		-	-	36	-	36	-	144	Зачёт, Зачёт с оценкой
Наличие в дисциплине КП/КР		-	-	-	-	-	-	-	-
Итого по дисциплине		-	-	36	-	36	-	144	Зачёт с оценкой

**Содержание лекционных занятий по дисциплине**

**Раздел 1.** Принципы объектно-ориентированного программирования

- Тема 1 Инкапсуляция.
- Тема 2 Дружественные функции.
- Тема 3 Наследование.
- Тема 4 Полиморфизм.
- Тема 5 Шаблоны классов.
- Тема 6 Классы контейнеры.

**Раздел 2.** Работа в визуальной среде проектирования

- Тема 1 Разработка визуального приложения. Работа с формой.  
*Компоненты. Свойства. Методы. События*
- Тема 2 Работа с графикой.
- Тема 3 Организация меню в приложении.
- Тема 4 Диалоги.

**Содержание лабораторных занятий по дисциплине**

**Раздел 1.** Принципы объектно-ориентированного программирования

- Тема 1 Инкапсуляция.  
Инкапсуляция, конструкторы, деструкторы, указатель this.
- Тема 2 Дружественные функции.  
Перегрузка функций, как методов и как функций.
- Тема 3 Наследование.  
Раннее связывание. Одиночное и множественное наследование.
- Тема 4 Полиморфизм.  
Позднее связывание, таблица виртуальных методов.
- Тема 5 Шаблоны классов.  
Изучения возможностей описания и использования шаблонов классов.
- Тема 6 Классы контейнеры.  
Изучение способов создания контейнерных классов.

## **Раздел 2. Работа в визуальной среде проектирования.**

Тема 1 Разработка визуального приложения. Работа с формой.

Компоненты. Свойства. Методы. События.

Тема 2 Работа с графикой.

Знакомство с принципами отображения графики в языке программирования C#.

Тема 3 Организация меню в приложении.

Организации в приложении главного и контекстного меню, продолжение изучения компонент отображения текстовой информации.

Тема 4 Диалоги.

Закрепление навыка написания обработчиков событий.

Знакомство с принципами организации в приложении главного и контекстного меню, продолжение изучения компонент отображения текстовой информации.

Организация хранения информации в файлах и работы с данными, хранимыми в файле.

## **5. ОЦЕНОЧНЫЕ СРЕДСТВА ДЛЯ ТЕКУЩЕГО КОНТРОЛЯ УСПЕВАЕМОСТИ, ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ ПО ИТОГАМ ОСВОЕНИЯ ДИСЦИПЛИНЫ И УЧЕБНО-МЕТОДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ САМОСТОЯТЕЛЬНОЙ РАБОТЫ СТУДЕНТОВ**

### **5.1. Текущий контроль успеваемости**

#### **Текущий контроль успеваемости студентов**

##### **Список примерных вопросов для рейтинг-контроля 1**

- 1) В каких случаях прибегают к механизму наследования?
- 2) В каком случае наследование называют одиночным?
- 3) В каком случае наследование называют множественным?
- 4) Происходит ли наследование конструкторов?
- 5) Какому правилу подчиняется вызов конструкторов при создании объекта класса потомка?
- 6) Какому правилу подчиняется вызов деструкторов при удалении из памяти объекта класса потомка?
- 7) Что такое ключ наследования?
- 8) Какой модификатор доступа будет у члена класса наследника, если в родительском классе он был объявлен с модификатором `private`, а наследование произведено с ключом `public`?
- 9) Какой модификатор доступа будет у члена класса наследника, если в родительском классе он был объявлен с модификатором `protected`, а наследование произведено с ключом `public`?
- 10) Какой модификатор доступа будет у члена класса наследника, если в родительском классе он был объявлен с модификатором `public`, а наследование произведено с ключом `private`?
- 11) Описать базовый класс и класс наследник. В `main` функции должна быть возможность создания и удаления объектов каждого из классов, вывода на экран значений полей.

#### **Текущий контроль успеваемости студентов**

##### **Список примерных вопросов для рейтинг-контроля 2**

- 1) Какую функцию называют виртуальной?
- 2) Поясните механизм раннего и позднего связывания.
- 3) Что такое таблица виртуальных методов?

- 4) Каким образом осуществляется связь с таблицей виртуальных методов?
  - 5) Каким образом определяется адрес нужного метода в таблице виртуальных методов?
  - 6) В чём преимущества позднего связывания?
  - 7) В чём недостатки позднего связывания?
  - 8) В каком случае виртуальный метод родительского класса останется виртуальным в классе наследнике?
  - 9) Какие методы нет смысла делать виртуальными?
  - 10) При вызове виртуального метода через указатель, на объект, метод будет вызван по типу указателя или указуемого?
  - 11) Что такое шаблон класса?
  - 12) Для чего используют шаблоны классов?
  - 13) Какие классы называют контейнерами?
  - 14) Какие есть два подхода к организации контейнерных классов?
  - 15) Каким образом создать объект класса с необходимым типом данных, описанного как шаблон?
  - 16) Каким образом можно определить размер памяти, который необходимо выделить под объект, если управляющая часть, совмещена с хранилищем?
  - 17) Если контейнерный класс, организован по принципу стека через массив, каким образом осуществляется доступ к элементам?
  - 18) Каким образом осуществляется проверка контейнера на пустоту?
  - 19) Каким образом осуществляется проверка контейнера на полное заполнение?
  - 20) Как получить доступ к первому элементу хранилища, если хранилище совмещено с управляющей частью объекта?
  - 21) Что такое шаблон класса?
  - 22) Для чего используют шаблоны классов?
  - 23) Какие классы называют контейнерами?
  - 24) Какие есть два подхода к организации контейнерных классов?
  - 25) Каким образом создать объект класса с необходимым типом данных, описанного как шаблон?
- Каким образом можно определить размер памяти, который необходимо выделить под объект, если управляющая часть, совмещена с хранилищем?
- 26) Если контейнерный класс, организован по принципу стека через массив, каким образом осуществляется доступ к элементам?
  - 27) Каким образом осуществляется проверка контейнера на пустоту?
  - 28) Каким образом осуществляется проверка контейнера на полное заполнение?
  - 29) Как получить доступ к первому элементу хранилища, если хранилище совмещено с управляющей частью объекта?

### Список примерных вопросов для рейтинг-контроля 3

- 1) Какие два способа изменения значения свойства компонента существуют?
- 2) Каково назначение компонента label?
- 3) Каково назначение свойства Text?
- 4) Каково назначение компонент контейнеров?
- 5) Перечислите известные вам компоненты контейнеры?
- 6) Каким образом можно организовать добавление в программу автоматически сгенерированного кода обработчика (заголовок и пустые фигурные скобки)?
- 7) Какие компоненты можно использовать, для организации получения ответа от пользователя да/нет?
- 8) Поясните назначение компонента Button.
- 9) Назовите основное событие компонента из предыдущего вопроса?

10) Какие компоненты позволяют организовать выбор пользователем одного или нескольких значений из списка?

## **5.2. Промежуточная аттестация по итогам освоения дисциплины (зачет с оценкой) Список примерных вопросов для переаттестации (зачёт) курса**

- 1) Что такое объект?
  - 2) Что такое класс?
  - 3) Для чего используют модификаторы доступа?
  - 4) Какие вы знаете модификаторы доступа?
  - 5) Напишите синтаксис создания объекта.
  - 6) Что такое конструктор?
  - 7) Какие виды конструкторов существуют?
  - 8) Сколько конструкторов может быть описано в классе?
  - 9) Что такое метод?
  - 10) Что такое дружественные функции?
  - 11) Что такое указатель `this`?
  - 12) Может ли программист перенастроить указатель `this`?
  - 13) Возможен ли доступ к указателю `this` из `main` функции?
  - 14) Что такое деструктор?
  - 15) Напишите синтаксис перегрузки оператора.
  - 16) Сколько и почему именно столько параметров нужно передать в метод класса, если он является перегруженным бинарным оператором?
  - 17) Сколько и почему именно столько параметров нужно передать в метод класса, если он является перегруженным унарным оператором?
  - 18) Сколько и почему именно столько параметров нужно передать дружественной функции если она является перегруженным бинарным оператором?
  - 19) Сколько и почему именно столько параметров нужно передать дружественной функции класса, если она является перегруженным унарным оператором?
- Что общего и в чём отличия дружественной функции и метода класса?

## **Промежуточная аттестация по итогам освоения 1 семестра дисциплины**

### **Примерный список вопросов к зачёту**

- 1) Описать иерархию классов: учащийся, ученик, студент, работающий студент, метод вывода информации об объекте сделать полиморфным.
- 2) На основе класса `Shape` (фигура) описать класс `Ellipsoid` (эллипсоид). Эллипсоид будет характеризоваться данными `x`, `y`, `z` (наследуются от `Shape`) и `a`, `b`, `c` – полуоси. В классе `Ellipsoid` переопределить функцию для расчета объема. Объем эллипсоида вычисляется по формуле  $V = 4\pi abc/3$ .
- 3) Создать абстрактный базовый класс с виртуальной функцией `Площадь`. Создать произвольные классы `Прямоугольник`, `Круг`, `Прямоугольный треугольник`, `Трапеция` со своими функциями площади. Для проверки определить массив ссылок на абстрактный класс, которым присваивается адреса различных объектов.
- 4) Разработать иерархию классов. В иерархии должно быть не менее двух уровней и не менее четырех классов (вместе с базовым классом). Каждый класс-потомок должен иметь хотя бы один собственный атрибут, и как минимум один собственный метод. Предусмотреть не менее одного виртуального метода, например, метод вывода информации об объекте на экран. В каждом классе-потомке должен быть описан собственный конструктор. В основной программе надо привести небольшой демонстрационный пример, показывающий возможности работы с классами-потомками: создание экземпляра класса-потомка, доступ к атрибутам и методам класса-предка, порядок вызова конструкторов и т.д.



5) Создать абстрактный базовый класс объектов на декартовой плоскости CShape (фигура). Разработать и реализовать иерархию классов конкретных фигур: CPoint, CCircle и т.д., не менее четырёх классов наследников. При этом: во всех классах должен быть реализован метод Square(), вычисляющий и возвращающий площадь объект и метод Show(), который отображает на экране информацию в текстовом виде о данном экземпляре (не менее трех характеристик; у точки есть имя и координаты, у окружности ещё есть радиус и площадь, у ломаной - длина и так далее.

6) Создать базовый класс Array с виртуальными методами сложения и поэлементной обработки массива. Разработать производные классы AndArray и OnArray. В первом классе операция сложения реализуется как пересечение множеств, а поэлементная обработка представляет собой извлечение квадратного корня. Во втором классе операция сложения реализуется как объединение, а поэлементная обработка-вычисление логарифма.

7) Создать иерархию классов Шахматная фигура – абстрактный класс, содержащий поле – цвет. Создать производные классы все фигуры, содержащие свое название и координаты позиции на доске. Определить конструктор копирования, оператор присваивания через соответствующие функции базового класса. Продемонстрировать работу классов.

8) Создать класс 3D фигура, и производные классы шар, конус, цилиндр и куб. В классе должны присутствовать виртуальные методы вычисления объёма и вывода информации об объекте на экран. Объявить массив указателей на базовый класс, настроить элементы массива на разнотипные объекты. Используя эти указатели продемонстрировать работу упомянутых выше методов.

9) Создать абстрактный класс - млекопитающие. Определить производные классы животные и люди. У животных определить производные классы собак и коров. Определить виртуальные функции описания человека, собаки и коровы.

10) Создать абстрактный класс Norm с виртуальной функцией вычисления нормы и модуля. Определить производные классы Complex, Vector3D с собственными функциями вычисления нормы и модуля. (Модуль для комплексного числа вычисляется как корень из суммы квадратов действительной и мнимой частей; норма для комплексных чисел вычисляется как модуль в квадрате. Модуль вектора вычисляется как корень квадратный из суммы квадратов координат; норма вектора вычисляется как максимальное из абсолютных значений координат).

### 5.3. Самостоятельная работа обучающегося.

Самостоятельная работа при освоении дисциплины «Объектно-ориентированное программирование» заключается в изучении содержания тем курса по конспектам, учебникам и дополнительной литературе, подготовке к лабораторным и практическим занятиям, оформлении отчёта по лабораторным работам подготовке к работам рейтинг-контроля, к экзамену. В ходе подготовки к защите лабораторных работ, работам рейтинг-контроля, необходимо самостоятельное решение достаточного количества заданий по всем разделам курса.

Фонд оценочных материалов (ФОМ) для проведения аттестации уровня сформированной компетенций, обучающихся по дисциплине «Объектно-ориентированное программирование» оформляется отдельным документом.

## 6. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

### 6.1. Книгообеспеченность

Наименование литературы: автор, название, вид издания, издательство	Год издания	КНИГООБЕСПЕЧЕННОСТЬ Наличие в электронном каталоге ЭБС
---	-------------	---

Основная литература*		
Математическая логика и теория алгоритмов: Учебник / Пруцков А.В., Волкова Л.Л. - М.:КУРС, НИЦ ИНФРА-М, 2016. - 152 с.: 60x90 1/16. - (Бакалавриат) (Переплёт 7БЦ) ISBN 978-5-906818-74-4	2016	<a href="http://znanium.com/catalog.php?bookinfo=558694">http://znanium.com/catalog.php?bookinfo=558694</a>
«Стек, или Путешествие туда и обратно», Алексей Вторников, 2017, Издательство: ДМК-Пресс, 142 с., ISBN: 978-5-97060-517-2	2017	<a href="https://e.lanbook.com/book/97356">https://e.lanbook.com/book/97356</a>
Павловская Т. А., С#. Программирование на языке высокого уровня. Учебник для вузов. — СПб.: Питер, 2009. — 432 с: ил. ISBN 978-5-91180-174-8	2009	<a href="https://www.iprbookshop.ru/366.html">https://www.iprbookshop.ru/366.html</a>
Дополнительная литература		
Городняя, Л. В. Парадигма программирования : учебное пособие для вузов / Л. В. Городняя. — 2-е изд., стер. — Санкт-Петербург : Лань, 2021. — 232 с. — ISBN 978-5-8114-6680-1	2021	<a href="https://e.lanbook.com/book/151660">https://e.lanbook.com/book/151660</a>
Системы виртуальной, дополненной и смешанной реальности : учебное пособие / А. А. Смолин, Д. Д. Жданов, И. С. Потемин [и др.]. — Санкт-Петербург : НИУ ИТМО, 2018. — 59 с.	2018	<a href="https://e.lanbook.com/book/136468">https://e.lanbook.com/book/136468</a>
Шапиро Л., Стокман Дж. Компьютерное зрение Издательство "Лаборатория знаний" (ранее "БИНОМ. Лаборатория знаний") Системы дополненной реальности учебное пособие 763 стр. 2020	2020	<a href="https://e.lanbook.com/book/173806">https://e.lanbook.com/book/173806</a>

## 6.2. Периодические издания

<https://elibrary.ru/>

1. Вестник компьютерных и информационных технологий
2. Вестник компьютерных и информационных технологий, ISSN: 1810-7206.
3. Computerworld Россия, ISSN: 1560-5213.
4. Мир ПК, ISSN: 0235-3520.

## 6.3. Интернет-ресурсы

Центр дистанционного образования <https://cs.cdo.vlsu.ru/>

ЭБС Консультант студента <https://www.studentlibrary.ru/>

## 7. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

Для реализации данной дисциплины имеются специальные помещения для проведения занятий лекционного типа, лабораторных занятий, текущего контроля и промежуточной аттестации, а также помещения для самостоятельной работы.

Лабораторные занятия проводятся в компьютерном классе (100-3, 1226-3, 5116, в-3 или аналогичной аудитории в зависимости от сетки расписания).

Рабочую программу составил Шишкина М.В., ст. преподаватель каф. ФиПМ

Рецензент Генеральный директор «ФС Сервис»

Д.С. Квасов

Программа рассмотрена и одобрена на заседании кафедры ФиПМ

Протокол №1 от 30.08.2021 года

Заведующий кафедрой

С.М. Аракелян

(ФИО, подпись)

Рабочая программа рассмотрена и одобрена

на заседании учебно-методической комиссии направления 02.03.02 «Фундаментальная информатика и информационные технологии»

Протокол №1 от 30.08.2021 года

Председатель комиссии

С.М. Аракелян

**ЛИСТ ПЕРЕУТВЕРЖДЕНИЯ  
РАБОЧЕЙ ПРОГРАММЫ ДИСЦИПЛИНЫ**

Рабочая программа одобрена на 20<sup>22</sup> / 20<sup>23</sup> учебный года

Протокол заседания кафедры № 1 от 30.08.2022 года

Заведующий кафедрой \_\_\_\_\_

*С.В. Абрамен*

Рабочая программа одобрена на 20\_\_\_\_ / 20\_\_\_\_ учебный года

Протокол заседания кафедры № \_\_\_\_\_ от \_\_\_\_\_ года

Заведующий кафедрой \_\_\_\_\_

Рабочая программа одобрена на 20\_\_\_\_ / 20\_\_\_\_ учебный года

Протокол заседания кафедры № \_\_\_\_\_ от \_\_\_\_\_ года

Заведующий кафедрой \_\_\_\_\_