

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
федеральное государственное бюджетное образовательное учреждение
высшего образования
«Тольяттинский государственный университет»

Б1.В.ДВ.04.02
(индекс дисциплины)

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

Разработка приложений на платформе Net (Дотнет) 2

(наименование дисциплины)

по направлению подготовки

01.03.02 Прикладная математика и информатика

направленность (профиль)

Компьютерные технологии и математическое моделирование

Форма обучения: очная

Год набора: 2023

Общая трудоемкость: 4 ЗЕ

Распределение часов дисциплины по семестрам

Семестр	7	Итого
Форма контроля	зачет	
Вид занятий		
Лекции	16	16
Лабораторные		
Практические	48	48
Руководство: курсовые работы (проекты) / РГР		
Промежуточная аттестация	0.25	0.25
Контактная работа	64.25	64.25
Самостоятельная работа	79.75	79.75
Контроль		
Итого	144	144

Рабочую программу составил(и):

доцент, к.п.н., Ерофеева Е.А.

(должность, ученое звание, степень, Фамилия И.О.)

(должность, ученое звание, степень, Фамилия И.О.)

Рецензирование рабочей программы дисциплины:

☐

Отсутствует

☐

Рецензент

(должность, ученое звание, степень, Фамилия И.О.)

Рабочая программа дисциплины составлена на основании ФГОС ВО и учебного плана направления подготовки (специальности)

01.03.02 Прикладная математика и информатика

Срок действия рабочей программы дисциплины до «31» августа 2027 г.

УТВЕРЖДЕНО

На заседании кафедры

«Прикладная математика и информатика»

(протокол заседания № 1 от «30» августа 2022 г.).

1. Цель освоения дисциплины

Цель освоения дисциплины – сформировать у студентов практические навыки по разработке программных приложений на платформе .Net для решения прикладных задач с применением современных методов и технологий программирования, обучить работе с научно-технической литературой и технической документацией по разработке и тестированию приложений.

2. Место дисциплины в структуре ОПОП ВО

Дисциплины, учебные курсы, на освоении которых базируется данная дисциплина: Программирование на языках высокого уровня; Алгоритмы и структуры данных; Архитектура компьютеров и операционные системы; Объектно-ориентированное программирование; Разработка приложений на платформе Net (Дотнет) 1.

Дисциплины, учебные курсы, для которых необходимы знания, умения, навыки, приобретаемые в результате изучения данной дисциплины: подготовка выпускной квалификационной работы.

3. Планируемые результаты обучения

Формируемые и контролируемые компетенции (код и наименование)	Индикаторы достижения компетенций (код и наименование)	Планируемые результаты обучения
ПК-5 Способен осуществлять выбор языка программирования и моделировать решение для реализации программного обеспечения	ПК-5.1 Знает принципы выбора языка программирования и моделирования решений для реализации программного обеспечения	Знать: технологии моделирования программного обеспечения Уметь: применять технологии моделирования программного обеспечения Владеть :навыками моделирования программного обеспечения
	ПК-5.2 Умеет использовать знания по выбору языка программирования и моделированию решений для реализации программного обеспечения	Знать: технологию разработки программного обеспечения на языках программирования Уметь: выбирать и моделировать решения по разработке программного обеспечения на языках программирования Владеть: навыками реализации программного обеспечения на языках программирования
	ПК-5.3 Владеет навыками выбора языка программирования и моделирования решений для реализации программного обеспечения	Знать: технологии моделирования решения для реализации программного обеспечения на языках программирования Уметь: моделировать решения для реализации программного обеспечения на языках программирования Владеть: инструментом моделирования

Формируемые и контролируемые компетенции (код и наименование)	Индикаторы достижения компетенций (код и наименование)	Планируемые результаты обучения
		решения для реализации программного обеспечения на языках программирования

4. Структура и содержание дисциплины

Модуль (раздел)	Вид учебной работы	Наименование тем занятий (учебной работы)	Семестр	Объем, ч.	Баллы	Интерактив, ч.	Формы текущего контроля (наименование оценочного средства)
1	Лекция 1	Коллекции и обобщенные типы	7	4		-	тест
1	Практическое занятие 1.1	Коллекции и обобщенные типы	7	4		-	
1	Практическое занятие 1.2	Коллекции и обобщенные типы	7	4		-	
1	Практическое занятие 1.3	Коллекции и обобщенные типы	7	4		-	
1	Практическое занятие 1.4	Коллекции и обобщенные типы	7	4		-	
1	Лекция 2	Введение в LINQ	7	4		-	
1	Практическое занятие 2.1	Введение в LINQ	7	4		-	
1	Практическое занятие 2.2	Введение в LINQ	7	4		-	
1	Практическое занятие 2.3	Введение в LINQ	7	4		-	
1	Практическое занятие 2.4	Введение в LINQ	7	4		-	
2	Лекция 3	Основы разработки программного обеспечения	7	2		-	
2	Практическое занятие 3.1	Основы разработки программного обеспечения	7	2		-	
2	Лекция 4	Юнит тесты	7	2		-	
2	Практическое занятие 4.1	Юнит тесты	7	2		-	

2	Практическое занятие 4.2	Юнит тесты	7	2		-	
2	Практическое занятие 4.3	Юнит тесты	7	2		-	
2	Лекция 5	Типичные ошибки при разработке программного обеспечения	7	2		-	
2	Практическое занятие 5.1	Типичные ошибки при разработке программного обеспечения	7	2		-	
2	Лекция 6	Работа с XML	7	2		-	
2	Практическое занятие 6.1	Работа с XML	7	2		-	
2	Практическое занятие 6.2	Работа с XML	7	2		-	
2	Практическое занятие 6.3	Работа с XML	7	2		-	
2	Практическое занятие 6.4	Работа с XML	7	2		-	
	Сам	Самостоятельное изучение материала	7	79,75			
	ПА		7	0,25			
Итого:				144			

5. Образовательные технологии

В рамках изучения дисциплины «Разработка приложений на платформе Net (Дотнет) 2» предусмотрено использование следующих образовательных технологий:

- технология дистанционного обучения: лекции, практические занятия, самостоятельная работа, реализуемые с применением информационно-телекоммуникационных сетей при опосредованном (на расстоянии) взаимодействии студентов и преподавателя.

6. Методические указания по освоению дисциплины

Дистанционное обучение предполагает самостоятельное изучение учебных дисциплин с использованием электронных учебно-методических комплексов, размещенных в системе обучения, консультации преподавателя при подготовке к тестированию и по его итогам, при подготовке к зачетам и экзаменам, контрольных и курсовых работ, а также участие в электронных семинарах и практических занятиях.

Самостоятельная работа студентов проводится с целью углубления и расширения теоретических знаний; развития познавательных способностей и активности студентов; самостоятельности, ответственности и организованности, творческой инициативы; формирования самостоятельности мышления, способности к саморазвитию, самосовершенствованию и самореализации.

Контроль выполненной самостоятельной работы осуществляется индивидуально, при защите рефератов, курсовых работ, творческих проектов, с использованием информационно-телекоммуникационных технологий.

6.1. Рекомендации по подготовке к лекционным занятиям

Изучение дисциплины требует систематического и последовательного накопления знаний, следовательно, пропуски отдельных тем не позволяют глубоко освоить предмет.

В ходе лекционных занятий студентам необходимо изучить наиболее значимые и актуальные темы и вопросы учебной дисциплины. Помимо лекционного материала студентам также рекомендуется самостоятельно проработать каждую тему с использованием дополнительной учебной литературы, указанной в библиографии курса (дисциплины). Студент может дополнить список использованной литературы современными источниками, не представленными в списке рекомендованной литературы, и в дальнейшем использовать собственные подготовленные учебные материалы при написании курсовых и выпускных квалификационных работ.

После изучения лекционного материала студент переходит к тестовому материалу, который состоит из тестов текущего контроля. Тесты текущего контроля размещены в конце каждой темы. К текущему тестированию студенту рекомендуется готовиться по вопросам для самоподготовки. Текущее тестирование, прежде всего, является одним из элементов самоконтроля и закрепления студентом пройденного учебного материала.

6.2. Рекомендации по подготовке к практическим занятиям

Практические занятия у дистанционных студентов могут проходить либо в виде тестирования, либо в виде практикума по решению задач.

Студентам следует:

- при подготовке к практическим занятиям следует обязательно использовать не только лекции, учебную литературу, но и другие источники;
- во время выполнения заданий студент может задать преподавателю вопросы по материалу, вызвавшему затруднения в его понимании и освоении при решении задач, заданных для самостоятельного решения, используя возможности форума, открытого в курсе системы обучения.

Доводить задания практической работы до окончательного решения, прикрепить выполненные задания в курсе системы обучения, в случае затруднений обращаться к преподавателю.

Форум – средство общения пользователей в сети с использованием специального программного обеспечения, позволяющее его участникам общаться между собой не в режиме реального времени. Сообщения, отправленные на форум, могут храниться в нём неограниченно долго, и ответ на форуме может быть дан в любое время, удобное его участнику, а не в тот же день, когда появился обсуждаемый вопрос. Посредством форума предоставляется возможность в системе дистанционного образования коллективного общения и обсуждения.

Для того чтобы практические занятия приносили максимальную пользу, необходимо помнить, что упражнение и решение задач проводятся по рассмотренному на лекциях материалу и связаны, как правило, с детальным разбором отдельных вопросов лекционного курса.

При этих условиях студент не только хорошо усвоит материал, но и научится применять его на практике, а также получит дополнительный стимул для активной проработки лекции.

6.3. Рекомендации по подготовке к зачету

Подготовка к зачету способствует закреплению, углублению и обобщению знаний, получаемых, в процессе обучения, а также применению их к решению практических задач. Готовясь к зачету, студент ликвидирует имеющиеся пробелы в знаниях, углубляет, систематизирует и упорядочивает свои знания. На зачете студент демонстрирует то, что он приобрел в процессе обучения по конкретной учебной дисциплине.

После изучения лекционного материала студент переходит к тестовому материалу, который состоит из тестов промежуточной аттестации (зачет, экзамен).

Перед тестированием в формате переписки студент имеет возможность получить консультацию преподавателя по наиболее сложным для него вопросам, а по итогам тестирования – оценку преподавателя и анализ уровня усвоения материала темы.

Тесты промежуточной аттестации произвольно формируются из вопросов по всем темам учебной дисциплины. Это позволяет преподавателю получить объективную оценку уровня знаний, умений и навыков, освоенных студентом.

7. Оценочные средства

7.1. Паспорт оценочных средств

Семестр	Код контролируемой компетенции (или ее части)	Наименование оценочного средства
7	ПК-5	Вопросы для собеседования
		Комплект отчетов по практическим работам

7.2. Типовые задания или иные материалы, необходимые для текущего контроля

7.2.1. Комплект отчетов по практическим работам (примеры)

Практическая работа №2 «Коллекции и обобщенные типы».

Задания к практической работе №2 «Бинарное дерево поиска».

- Разработать тип «бинарное дерево поиска» для хранения результатов тестов, выполненных студентами, и их предоставления в упорядоченном виде.
- Предусмотреть возможность хранения данных любого типа, поддерживающего сравнения.
- Информация о студенте может содержать поля, хранящие имя студента, название теста, дату его прохождения и оценку теста для данного студента.
 - Реализовать обобщенный стандартный интерфейс, чтобы информацию о студенте можно было сравнить с информацией о другом студенте.
- Предусмотреть следующие конструкторы «дерева»:
 - Который принимает объект стандартного обобщённого типа для сравнения двух объектов одного типа, и применял его для сравнения добавляемых в дерево элементов.
 - Который принимает объект типа стандартной обобщенной коллекции (не список!), и добавлял элементы коллекции в дерево (пример смотрите в «исходниках» конструкторов типа List<T>).
 - Который принимает и коллекцию элементов для добавления в дерево, и объект с помощью, которого будет выполнять сравнения.
- ❖ Итого: если указан «компаратор», то он используется для сравнения элементов, иначе – используется «сравнение, реализованное внутри элемента».
- Созданный класс «дерево» должен реализовывать обобщённый интерфейс перечислителя. При реализации нужно применить «итератор» (реализовать с помощью yield, см. важно п.3)).
- В «дереве» реализовать метод, который возвращает объект для итерации по дереву в обратном порядке.
Например, GetReversedEnumerator()(реализовать с помощью yield, см. важно п.3))
- В качестве UI-интерфейса использовать консольное приложение с интерфейсом командной строки, WinForms или WPF-приложение. В приложении нужно:
 - Применить созданное «дерево» для хранения чисел из целочисленного массива, при этом в дерево они сразу должны храниться в порядке убывания.
 - Добавить ещё несколько чисел в дерево.
 - Вывести все числа «дерева» в том порядке, в котором они хранятся в дереве.
 - Применить созданное «дерево» для хранения результатов прохождения тестов

студентами, при этом элементы в дереве **сразу** должны храниться в порядке убывания «оценок (баллов)» по тесту.

- Вывести информацию о студентах в том порядке, в котором они хранятся в дереве.
- Применить созданное «дерево» для хранения строк:
 - Строки добавить из списка строк;

Вывести строки в обратном порядке, используя метод «дерева», который возвращает «перечислитель» для прохода «дерева» в обратном порядке.

Практическая работа №3 «Введение в LINQ ».

Задания к практической работе №3:

- Разработать тип, реализующий алгоритм Евклида для вычисления НОД двух целых чисел (http://en.wikipedia.org/wiki/Euclidean_algorithm).
- Добавить к разработанному типу дополнительную функциональность в виде перегруженных методов вычисления НОД для трех, четырех или пяти целых чисел.
- Добавить к разработанному типу метод, реализующий алгоритм Стейна (бинарный алгоритм Евклида) для расчета НОД двух целых чисел (http://en.wikipedia.org/wiki/Binary_GCD_algorithm).
- Сделать «обертку для метода», которая принимает выходной параметр, содержащий значение времени, необходимое для выполнения расчетов.
- Сделать «метод-обертку» для метода реализующего алгоритм Евклида вычисления НОД двух параметров, добавив аналогичную функциональность.
- Создать unit-тесты для тестирования разработанных методов.
- В качестве UI-интерфейса использовать консольное приложение с интерфейсом командной строки, WinForms или WPF-приложение.

Практическая работа №4 «XML и платформа Microsoft .NET Framework».

Задания к практической работе №4:

- Разработать класс «треугольник» для работы с треугольником, заданным на плоскости сторонами a, b, c.
- Предусмотреть возможность проверки существования треугольника, реализовать функции подсчета его площади и периметра.
- В качестве UI-интерфейса использовать консольное приложение с интерфейсом командной строки, WinForms или WPF-приложение.
- Создать unit-тесты для тестирования методов разработанного типа.

Практическая работа №5 «Основы XML».

Задания к практической работе №5:

- Разработать класс «вектор» для работы с трехмерными векторами.
- Перегрузить для класса операции, допустимые для работы с трехмерными векторами.
- Создать unit-тесты для тестирования разработанных методов.
- В качестве UI-интерфейса использовать консольное приложение с интерфейсом командной строки, WinForms или WPF-приложение.
- Разработать класс «многочлен» для работы с многочленами от одной переменной.
- Перегрузить для класса операции, допустимые для работы с многочленами.
- Создать unit-тесты для тестирования разработанных методов.
- В качестве UI-интерфейса использовать консольное приложение с интерфейсом командной строки, WinForms или WPF-приложение.

Содержание отчета

I. Титульный лист.

- II. Название и цель работы.
- III. Результаты выполнения работы
- IV. Заключение
- V. Приложения

Требования к оформлению

Отчёт по практическому занятию выполняется на страницах формата А4 в электронном виде.

При оформлении отчёта используется сквозная нумерация страниц, считая титульный лист первой страницей. Номер страницы на титульном листе не ставится. Номера страницы ставятся по центру сверху.

При оформлении отчёта соблюдать следующие требования:

- Для заголовков: полужирный шрифт, 14 пт, центрированный.
- Для основного текста: нежирный шрифт, 14 пт, выравнивание по ширине.
- Во всех случаях тип шрифта – Times New Roman, отступ абзаца 1.25 см, полуторный междустрочный интервал.
- Поля: левое – 2 см, правое, верхнее и нижнее – 1 см.

Процедура оценивания

Оценка выполненного практического занятия проводится по следующим критериям:

1. Наличие всей существенной информации по работе
2. Точность и полнота предоставляемых сведений
3. Непротиворечивость приводимой информации
4. Правильность интерпретаций и выводов, которые сделаны по результатам работы
5. Степень достижения студентом поставленной цели
6. Обоснованность применяемого решения
7. Грамотность (содержательная) используемых формулировок

Критерии оценки за отчеты по практическим занятиям:

- оценка «зачтено» ставится студенту, который продемонстрировал результаты выполнения практического занятия, соответствующие поставленным задачам, и предоставил отчет, оформленный должным образом и содержащий краткое описание полученных результатов;
- оценка «не зачтено» ставится студенту, который не продемонстрировал результаты выполнения практического занятия или не представил по ней отчет или представленный отчет не соответствует требованиям по оформлению.

7.3. Оценочные средства для промежуточной аттестации по итогам освоения дисциплины

7.3.1. Вопросы к промежуточной аттестации

Семестр 7

№ п/п	Вопросы к зачету
1.	Понятие инкапсуляции как одного из основных принципов объектно-ориентированного программирования.
2.	Применение свойств класса. Внутреннее представление свойств.
3.	Понятие наследования как одного из основных принципов объектно-ориентированного программирования. Примеры реализации.
4.	Понятие полиморфизма как одного из основных принципов объектно-ориентированного программирования. Примеры реализации.
5.	Принципы реализации графического интерфейса программных приложений.
6.	Проектирование базы данных с использованием MS SQL - сервера.
7.	Пространства имен ADO.NET. Типы пространства имен System.Data.
8.	Тип DataView. Возможности класса DataSet.
9.	Статические методы и методы экземпляров.
10.	Статические данные.
11.	Библиотека базовых классов .NET. Сборки. MSIL.
12.	Статические члены System.Object. Упаковка и распаковка.
13.	Базовый класс System.Array. Примеры алгоритмов обработки.
14.	Применение System.StringBuilder. Примеры использования.
15.	Ссылки класса на самого себя, перенаправление вызовов конструктора с использованием this. Определение открытого интерфейса по умолчанию. Примеры реализации.
16.	Реализация инкапсуляции при помощи традиционных методов доступа и изменения.
17.	Тип DataColumn. Первичный ключ таблицы. Тип DataRow. Тип DataTable.
18.	Использование класса DataRelation для моделирования отношений между таблицами.
19.	Статические конструкторы. Статические поля.
20.	Принципы построения приложений обработки баз данных.
21.	Базовый и производные классы. Работа с конструктором базового класса. Примеры реализации.
22.	Работа с конструктором базового класса. Примеры реализации.
23.	Запрет наследования. Применение модели включения-делегирования. Примеры реализации.
24.	Механизм позднего связывания. Примеры реализации.
25.	Абстрактные классы. Абстрактные методы.
26.	Конструкторы и деструкторы.
27.	Философия .NET.
28.	Открытые, закрытые и защищенные методы класса. Примеры реализации.
29.	Использование памяти для структурных (числовые данные, перечисления и структуры) и ссылочных (классы и интерфейсы) типов.
30.	Класс System.Object. Замещение методов System.Object.
31.	Определение пользовательских методов класса.
32.	Модификаторы уровня доступа к методам.
33.	Модификаторы для параметров методов.

34.	В связи с чем язык XML получил широкое распространение?
35.	Что представляет собой язык разметки?
36.	Какими особенностями обладает язык разметки XML?
37.	Что представляет собой платформа Microsoft .NET Framework?
38.	Каково назначение платформы Microsoft .NET Framework?
39.	Из каких компонентов состоит платформа Microsoft .NET Framework?
40.	Каковы функции среды CLR?
41.	Что представляет собой библиотека классов Microsoft .NET Framework?
42.	На основе каких стандартов строится web-служба XML в рамках платформы Microsoft .NET Framework?
43.	Какие API для реализации работы с XML разработаны в рамках платформы Microsoft .NET Framework?
44.	Что представляет собой XML-документ?
45.	Из каких разделов состоит XML-документ?
46.	Какую информацию содержит декларация XML-документа?
47.	Какая информация размещается в прологе XML-документа?
48.	Как строятся элементы XML-документа?
49.	Каковы синтаксические правила создания атрибутов XML документа?
50.	Для чего в XML-документах используются комментарии?
51.	Чем парсируемые данные в XML-документе отличаются от непарсируемых данных?
52.	Что представляют собой сущности XML-документа?
53.	Для чего в XML-документе необходимо пространство имен?
54.	Для чего необходима валидация XML-документа?
55.	Что определяет DTD-схема документа?
56.	Какие инструкции используются в DTD-схеме для описания грамматики XML-документа?
57.	Каковы достоинства и недостатки DTD-схем?
58.	Что представляет собой XML(XDR)-схема?
59.	Какие структурные элементы и характеристики XML-документа описывает XML(XDR)-схема?
60.	В чем заключается разница между простыми и комплексными типами элементов в XML(XDR)-схемах?
61.	Какие типы элементов, описываемых с помощью XML(XDR)-схем, могут содержать атрибуты?
62.	Что определяют примитивы XML(XDR)-схем?
63.	В чем заключается основная разница между DTD и XML(XDR)схемами?
64.	Для чего разработан язык XPath? Как прописываются пути в XPath?
65.	Что определяет ось выборки XPath?
66.	С какой целью в XPath используются предикаты?
67.	Чем определяется контекст выражения в XPath?
68.	Каково основное назначение технологии XSLT?
69.	Как строятся шаблоны преобразований в XSLT?
70.	Какова структура шаблона XSLT?
71.	Как с помощью языка XSLT создаются новые структурные единицы документа (элементы, атрибуты и т.п.)?
72.	Каков алгоритм преобразования XML-документа с помощью языка XSLT?

7.3.2. Критерии и нормы оценки

Семестр	Форма проведения промежуточной аттестации	Критерии и нормы оценки	
7	Зачет	«зачтено»	ставится студенту, если он на поставленные зачетные вопросы дал грамотные (полные или краткие) ответы или допустил небольшие неточности
		«не зачтено»	ставится студенту, если он не дал ответы на поставленные зачетные вопросы или в ответе содержались фундаментальные ошибки

8. Учебно-методическое и информационное обеспечение дисциплины

8.1. Обязательная литература

№ п/п	Авторы, составители	Заглавие (заголовок)	Тип (учебник, учебное пособие, учебно-методическое пособие, практикум, др.)	Год издания	Количество в научной библиотеке / Наименование ЭБС
1	Гуриков С.Р.	Введение в программирование на языке Visual C#: учеб. пособие / С.Р. Гуриков. — М. : ФОРУМ : ИНФРА-М, 2017. — 447 с. — (Высшее образование: Бакалавриат).	учебное пособие	2017	ЭБС Znanium
2	Зыков С. В.	Основы современного программирования [Электронный ресурс] : Разработка гетерогенных систем в Интернет-ориентированной среде : учеб. курс / С. В. Зыков. - Саратов : Ай Пи Эр Медиа, 2017. - 484 с. : ил. - ISBN 978-5-9908055-9-0.	учебное пособие	2017	ЭБС "IPRbooks"
3	Каймин В.А.	Информатика: Учебник / В. А. Каймин - 6-е изд. – Москва : НИЦ ИНФРА-М, 2015. - 285 с.: 60х90 1/16. - (Высшее образование: Бакалавриат)ISBN 978-5-16-010876-6	учебник	2015	ЭБС Znanium
	Снетков В. М.	Практикум прикладного программирования на C# в среде VS.NET 2008 [Электронный ресурс] / В. М. Снетков. - 2-е изд., испр. - Москва : ИНТУИТ, 2016. - 1691 с.	практикум	2016	ЭБС "IPRbooks"
	Тюкачев Н. А.	C#. Программирование 2D и 3D векторной графики [Электронный ресурс] : учеб. пособие / Н. А. Тюкачев, В. Г. Хлебостроев. - Изд. 2-е, испр. и доп. - Санкт-Петербург : Лань, 2017. - 320 с. : ил. - (Учебники для вузов. Специальная литература). - ISBN 978-5-8114-2568-6.	учебное пособие	2017	ЭБС "Лань"

8.2. Дополнительная литература

№ п/п	Авторы, составители	Заглавие (заголовок)	Тип (учебник, учебное пособие, учебно- методическое пособие, практикум, др.)	Год издания	Количество в научной библиотеке / Наименование ЭБС
1	Костюкова Н. И.	Комбинаторные алгоритмы для программистов [Электронный ресурс] : [учеб. пособие] / Н. И. Костюкова. - 2-е изд., испр. - Москва : ИНТУИТ, 2016. - 217 с. : ил. - (Основы информационных технологий). - ISBN 978-5-9556-0069-7.	учебное пособие	2016	ЭБС "IPRbooks"
2	Кирнос В. Н.	Информатика II. Основы алгоритмизации и программирования на языке C++ [Электронный ресурс] : учебно-методическое пособие / В. Н. Кирнос. - Томск : Эль Контент, 2013. - 160 с. : ил.	учебно-методическое пособие	2013	ЭБС "IPRbooks"
3	Туральчук К. А.	Параллельное программирование с помощью языка C# [Электронный ресурс] : [учеб. пособие] / К. А. Туральчук. - 2-е изд., испр. - Москва : ИНТУИТ, 2016. - 189 с. : ил. - (Основы информационных технологий).	практикум	2016	ЭБС "IPRbooks"
4	Хорев П. Б.	Объектно-ориентированное программирование с примерами на C# [Электронный ресурс] : учеб. пособие / П. Б. Хорев. - Москва : Форум : ИНФРА-М, 2016. - 200 с. - (Высшее образование. Бакалавриат). - ISBN 978-5-00091-144-0.	учебное пособие	2016	ЭБС Znanium

8.3. Перечень профессиональных баз данных и информационных справочных систем

1. Бесплатный для студентов, аспирантов, школьников и преподавателей доступ к полным лицензионным версиям инструментов Microsoft для разработки и дизайна <http://www.dreamspark.ru/>- WebofScience[Электронный ресурс] : мультидисциплинарная реферативная база данных. – Philadelphia: ClarivateAnalytics, 2016– . – Режим доступа : apps.webofknowledge.com. – Загл. с экрана. – Яз. рус.,англ.
2. Scopus[Электронный ресурс] : реферативная база данных. – Netherlands: Elsevier, 2004– . – Режим доступа : scopus.com. – Загл. С экрана.– Яз. рус., англ.
3. Elibrary[Электронный ресурс] : научная электронная библиотека. – Москва : НЭБ, 2000– . – Режим доступа : elibrary.ru. – Загл. с экрана. – Яз. рус., англ.

8.4. Перечень программного обеспечения

№ п/п	Наименование ПО	Реквизиты договора (дата, номер, срок действия)
1	DreamSpark в составе: Microsoft Visio; Microsoft Visual Studio; Microsoft Access; Microsoft Project	До 01.07.2020. Продлевается каждые 3 года

8.5. Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине

п/п	Наименование оборудованных учебных кабинетов, лабораторий, мастерских и др. объектов для проведения практических и лабораторных занятий, помещений для самостоятельной работы обучающихся (номер аудитории)	Перечень основного оборудования
1.	Аудитория вебконференций. Учебная аудитория для проведения занятий лекционного типа. Учебная аудитория для проведения занятий семинарского типа. Учебная аудитория для курсового проектирования (выполнения курсовых работ). Учебная аудитория для проведения групповых и индивидуальных консультаций Учебная аудитория для проведения занятий текущего контроля и промежуточной аттестации. (УЛК-807)	Экран телевизионный, ширмы, проектор на штативе. Стол преподавательский, стулья преподавательские. Транспарант-перетяжка, системный блок