

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ  
Федеральное государственное автономное образовательное учреждение высшего образования  
"Казанский (Приволжский) федеральный университет"  
Институт математики и механики им. Н.И. Лобачевского



**УТВЕРЖДАЮ**

Проректор по образовательной деятельности КФУ

\_\_\_\_\_ Д.А. Таюрский

"\_\_" \_\_\_\_\_ 20\_\_ г.

## **Программа дисциплины**

Программирование на C# и платформа. Net

Направление подготовки: 01.03.01 - Математика

Профиль подготовки: Общий профиль

Квалификация выпускника: бакалавр

Форма обучения: очное

Язык обучения: русский

Год начала обучения по образовательной программе: 2017

## Содержание

1. Перечень планируемых результатов обучения по дисциплине (модулю), соотнесенных с планируемыми результатами освоения ОПОП ВО
2. Место дисциплины (модуля) в структуре ОПОП ВО
3. Объем дисциплины (модуля) в зачетных единицах с указанием количества часов, выделенных на контактную работу обучающихся с преподавателем (по видам учебных занятий) и на самостоятельную работу обучающихся
4. Содержание дисциплины (модуля), структурированное по темам (разделам) с указанием отведенного на них количества академических часов и видов учебных занятий
  - 4.1. Структура и тематический план контактной и самостоятельной работы по дисциплине (модулю)
  - 4.2. Содержание дисциплины (модуля)
5. Перечень учебно-методического обеспечения для самостоятельной работы обучающихся по дисциплине (модулю)
6. Фонд оценочных средств по дисциплине (модулю)
7. Перечень литературы, необходимой для освоения дисциплины (модуля)
8. Перечень ресурсов информационно-телекоммуникационной сети "Интернет", необходимых для освоения дисциплины (модуля)
9. Методические указания для обучающихся по освоению дисциплины (модуля)
10. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине (модулю), включая перечень программного обеспечения и информационных справочных систем (при необходимости)
11. Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине (модулю)
12. Средства адаптации преподавания дисциплины (модуля) к потребностям обучающихся инвалидов и лиц с ограниченными возможностями здоровья
13. Приложение №1. Фонд оценочных средств
14. Приложение №2. Перечень литературы, необходимой для освоения дисциплины (модуля)
15. Приложение №3. Перечень информационных технологий, используемых для освоения дисциплины (модуля), включая перечень программного обеспечения и информационных справочных систем

Программу дисциплины разработал(а)(и) доцент, к.н. (доцент) Липачев Е.К. (Кафедра Интеллектуальные технологии поиска, Институт информационных технологий и интеллектуальных систем), elipachev@gmail.com ; доцент, к.н. Насрутдинов М.Ф. (кафедра компьютерной математики и информатики, отделение педагогического образования), Marat.Nasrutdinov@kpfu.ru

## 1. Перечень планируемых результатов обучения по дисциплине (модулю), соотнесенных с планируемыми результатами освоения ОПОП ВО

Обучающийся, освоивший дисциплину (модуль), должен обладать следующими компетенциями:

Обучающийся, освоивший дисциплину (модуль):

Должен знать:

Знать синтаксис и основные конструкции языка C#, знать принципы технологии .NET, навыки разработки программ на языке C#

Должен уметь:

Уметь: работать в среде разработки MS Visual Studio, создавать и отлаживать проекты.

Владеть приемами программирования в среде MS Visual Studio.

Должен владеть:

Владеть приемами программирования в среде MS Visual Studio.

Должен демонстрировать способность и готовность:

Создавать проекты на C#

## 2. Место дисциплины (модуля) в структуре ОПОП ВО

Данная дисциплина (модуль) включена в раздел "Б1.В.ДВ.05.07 Дисциплины (модули)" основной профессиональной образовательной программы 01.03.01 "Математика (Общий профиль)" и относится к дисциплинам по выбору.

Осваивается на 4 курсе в 8 семестре.

## 3. Объем дисциплины (модуля) в зачетных единицах с указанием количества часов, выделенных на контактную работу обучающихся с преподавателем (по видам учебных занятий) и на самостоятельную работу обучающихся

Общая трудоемкость дисциплины составляет 2 зачетных(ые) единиц(ы) на 72 часа(ов).

Контактная работа - 42 часа(ов), в том числе лекции - 20 часа(ов), практические занятия - 22 часа(ов), лабораторные работы - 0 часа(ов), контроль самостоятельной работы - 0 часа(ов).

Самостоятельная работа - 30 часа(ов).

Контроль (зачёт / экзамен) - 0 часа(ов).

Форма промежуточного контроля дисциплины: зачет в 8 семестре.

## 4. Содержание дисциплины (модуля), структурированное по темам (разделам) с указанием отведенного на них количества академических часов и видов учебных занятий

### 4.1 Структура и тематический план контактной и самостоятельной работы по дисциплине (модулю)

N	Разделы дисциплины / модуля	Семестр	Виды и часы контактной работы, их трудоемкость (в часах)			Самостоятельная работа
			Лекции	Практические занятия	Лабораторные работы	
1.	Тема 1. Обзор платформы Microsoft .Net	8	1	0	0	
2.	Тема 2. Обзор языка программирования C#	8	1	0	0	4
3.	Тема 3. Система типов языка C#	8	2	0	0	2
4.	Тема 4. Выражения и исключения в C#	8	2	2	0	2
5.	Тема 5. Инструкции управления	8	2	4	0	
6.	Тема 6. Массивы и строки	8	2	2	0	4

N	Разделы дисциплины / модуля	Семестр	Виды и часы контактной работы, их трудоемкость (в часах)			Самостоятельная работа
			Лекции	Практические занятия	Лабораторные работы	
7.	Тема 7. Принципы объектно-ориентированного программирования	8	2	2	0	4
8.	Тема 8. Использование ссылочных типов	8	1	2	0	4
9.	Тема 9. Перегрузка операторов	8	1	0	0	2
10.	Тема 10. Наследование в C#	8	1	2	0	2
11.	Тема 11. Агрегирование, пространства имен и область видимости	8	1	0	0	4
12.	Тема 12. Операции, делегаты и события	8	1	2	0	
13.	Тема 13. Свойства и индексы	8	1	2	0	
14.	Тема 14. Интерфейсы, структуры и перечисления	8	1	2	0	
15.	Тема 15. Ввод и вывод в C#	8	1	2	0	2
	Итого		20	22	0	30

## 4.2 Содержание дисциплины (модуля)

### Тема 1. Обзор платформы Microsoft .Net

Основные понятия. .NET Framework - инфраструктура платформы Microsoft.NET. CLS (Common Language Specification) - общая спецификация языков программирования. CLS как это основа межъязыкового взаимодействия в рамках платформы Microsoft.NET. CLR (Common Language Runtime) - основной компонент NET Framework - Среда Времени Выполнения или Виртуальная Машина, обеспечивающая выполнение сборки. Управляемый код - программный код, который при своем выполнении способен использовать службы, предоставляемые CLR. FCL (.NET Framework Class Library) соответствующая CLS- спецификации объектно-ориентированная библиотека классов, интерфейсов и системы типов (типов-значений), которые включаются в состав платформы Microsoft.NET. .NET-приложение - приложение, разработанное для выполнения на платформе Microsoft.NET Реализуется на языках программирования, соответствующих CLS. MSIL (Microsoft Intermediate Language) - промежуточный язык платформы Microsoft.NET.

### Тема 2. Обзор языка программирования C#

Синтаксис языка программирования C#. Типы языка программирования C#. Объявление констант и переменных. Классы и объекты. Составляющие класса. Класс object. Пространства имен. Важные пространства имен. Доступ к пространствам имен. Типы приложений. Создание консольного приложения. Базовое приложение на языке программирования C# с вводом и выводом на консоль.

### Тема 3. Система типов языка C#

Система типов: типы значений (типы ?значения), ссылочные типы (типы-ссылки), параметризованные типы (типы-шаблоны). Простые (элементарные) типы - это типы, имя и основные свойства которых известны компилятору. Sbyte, Byte, Short, Ushort, Int, uint, long, ulong, char, float, double, decimal, bool. Среди множества массов выделяют предопределенные ссылочные типы object и string, которым соответствуют FCL-типы System.Object и System.String.

Литералы. Шестнадцатеричные литералы. Управляющие последовательности символов. Строковые литералы. Преобразование и приведение типов. Автоматическое преобразование типов. Приведение несовместимых типов. Преобразование типов в выражениях. Приведение типов в выражениях.

### Тема 4. Выражения и исключения в C#

Арифметические операторы. Инкремент и декремент. Операторы отношений и логические операторы. Сокращенные логические операторы. Оператор присваивания. Составные операторы присваивания. Поразрядные операторы. Поразрядные операторы И, ИЛИ, исключающее ИЛИ и НЕ. Операторы сдвига. Поразрядные составные операторы присваивания. Оператор ?. Использование пробелов и круглых скобок. Приоритет операторов.

### Тема 5. Инструкции управления

Инструкция if. Вложенные if-инструкции. Конструкция if-else-if. Инструкция switch. Вложенные инструкции switch. Цикл for. Вариации на тему цикла for. Использование нескольких управляющих переменных цикла. Условное выражение. Отсутствие элементов в определении цикла. Бесконечный цикл. Циклы без тела. Объявление управляющей переменной в цикле for. Цикл while. Цикл do-while. Цикл foreach. Использование инструкции break для выхода из цикла. Использование инструкции continue. Инструкция return. Инструкция goto.

### **Тема 6. Массивы и строки**

Массивы. Одномерные массивы. Инициализация массива. Соблюдение "пограничного режима". Многомерные массивы. Двумерные массивы. Массивы трех и более измерений. Инициализация многомерных массивов. Рваные массивы. Присвоение значений ссылочным переменным массивов. Использование свойства Length. Использование свойства Length при работе с рваными массивами. Цикл foreach. Строки. Создание строк. Работа со строками. Массивы строк. Постоянство строк. Использование строк в switch-инструкциях.

### **Тема 7. Принципы объектно-ориентированного программирования**

Общая форма определения класса. Определение класса. Создание объектов. Переменные ссылочного типа и присвоение им значений. Методы. Добавление методов в класс Building. Возвращение из метода. Возврат значения. Использование параметров. Добавление параметризованного метода в класс Building. Как избежать написания недостижимого кода. Конструкторы. Параметризованные конструкторы. Добавление конструктора в класс Building. Использование оператора new. Применение оператора new к переменным типа значений. Сбор "мусора" и использование деструкторов. Деструкторы. Ключевое слово this.

### **Тема 8. Использование ссылочных типов**

Управление доступом к членам класса. Спецификаторы доступа C#. Применение спецификаторов доступа public и private. Передача объектов методам. Как происходит передача аргументов. Использование ref- и out-параметров. Использование модификатора ref. Использование модификатора out. Использование модификаторов ref и out для ссылочных параметров. Использование переменного количества аргументов. Возвращение методами объектов. Возвращение методами массивов. Перегрузка методов. Перегрузка конструкторов. Вызов перегруженного конструктора с помощью ссылки this. Метод Main(). Возвращение значений из метода Main(). Передача аргументов методу Main(). Рекурсия. Использование модификатора типа static. Статические конструкторы.

### **Тема 9. Перегрузка операторов**

Основы перегрузки операторов. Перегрузка бинарных операторов. Перегрузка унарных операторов. Выполнение операций над значениями встроенных C#-типов. Перегрузка операторов отношений. Перегрузка операторов true и false. Перегрузка логических операторов. Простой случай перегрузки логических операторов. Включение операторов, действующих по сокращенной схеме вычислений. Операторы преобразования. Рекомендации и ограничения по созданию перегруженных операторов.

### **Тема 10. Наследование в C#**

Основы наследования. Доступ к членам класса и наследование. Использование защищенного доступа. Конструкторы и наследование. Вызов конструкторов базового класса. Наследование и сокрытие имен. Использование ключевого слова base для доступа к скрытому имени. Создание многоуровневой иерархии. Последовательность вызова конструкторов. Ссылки на базовый класс и объекты производных классов. Виртуальные методы и их переопределение. Зачем переопределять методы. Применение виртуальных методов. Использование абстрактных классов. Использование ключевого слова sealed для предотвращения наследования. Класс object. Приведение к объектному типу и восстановление значения. Использование класса object в качестве обобщенного типа данных.

### **Тема 11. Агрегирование, пространства имен и область видимости**

Пространства имен. Объявление пространства имен. Пространства имен предотвращают конфликты по совпадению имен. Ключевое слово using. Вторая форма использования директивы using. Аддитивность пространств имен. Пространства имен могут быть вложенными. Пространство имен по умолчанию. Препроцессор. #define. #if и #endif. #else и #elif. #undef. #error. #warning. #line. #region и #endregion. Компоновочные файлы и модификатор доступа internal. Модификатор доступа internal.

### **Тема 12. Операции, делегаты и события**

Делегаты. Многоадресная передача. Класс System.Delegate. Назначение делегатов. События. Пример события для многоадресной передачи. Сравнение методов экземпляров классов со статическими методами, используемыми в качестве обработчиков событий. Использование событийных средств доступа. Смешанные средства обработки событий. Рекомендации по обработке событий в среде .NET Framework. Использование встроенного делегата EventHandler.

### **Тема 13. Свойства и индексы**

Индексаторы в языке программирования C#. Создание одномерных индексаторов. Перегрузка индексаторов. Индексаторам не требуется базовый массив. Многомерные индексаторы. Свойства в языке программирования C#. Правила использования свойств в языке программирования C#. Свойства только для чтения. Примеры свойств. Использование индексаторов и свойств.

### **Тема 14. Интерфейсы, структуры и перечисления**

Интерфейсы. Реализация интерфейсов. Использование интерфейсных ссылок. Интерфейсные свойства. Интерфейсные индексы. Наследование интерфейсов. Скрытие имен с помощью наследования интерфейсов. Явная реализация членов интерфейса. Закрытая реализация. Как избежать неопределенности с помощью явной реализации. Выбор между интерфейсом и абстрактным классом. Стандартные интерфейсы среды .NET Framework. Структуры. Зачем нужны структуры. Перечисления. Инициализация перечислений. Задание базового типа перечисления. Использование перечислений.

### **Тема 15. Ввод и вывод в C#**

Организация C#-системы ввода-вывода. Байтовые и символьные потоки. Встроенные потоки. Классы потоков. Класс Stream. Байтовые классы потоков. Символьные классы потоков. Двоичные потоки. Консольный ввод-вывод данных. Считывание данных из консольного входного потока. Запись данных в консольный входный поток. Класс FileStream и файловый ввод-вывод на побайтовой основе. Как открыть и закрыть файл. Считывание байтов из объекта класса FileStream. Запись данных в файл. Использование класса FileStream для копирования файла. Файловый ввод-вывод с ориентацией на символы. Использование класса StreamWriter. Использование класса StreamReader. Перенаправление стандартных потоков. Считывание и запись двоичных данных. Класс BinaryWriter. Класс BinaryReader. Демонстрация использования двоичного ввода-вывода. Файлы с произвольным доступом. Использование класса MemoryStream. Использование классов StringReader и StringWriter. Преобразование числовых строк во внутреннее представление.

### **5. Перечень учебно-методического обеспечения для самостоятельной работы обучающихся по дисциплине (модулю)**

Самостоятельная работа обучающихся выполняется по заданию и при методическом руководстве преподавателя, но без его непосредственного участия. Самостоятельная работа подразделяется на самостоятельную работу на аудиторных занятиях и на внеаудиторную самостоятельную работу. Самостоятельная работа обучающихся включает как полностью самостоятельное освоение отдельных тем (разделов) дисциплины, так и проработку тем (разделов), осваиваемых во время аудиторной работы. Во время самостоятельной работы обучающиеся читают и конспектируют учебную, научную и справочную литературу, выполняют задания, направленные на закрепление знаний и отработку умений и навыков, готовятся к текущему и промежуточному контролю по дисциплине.

Организация самостоятельной работы обучающихся регламентируется нормативными документами, учебно-методической литературой и электронными образовательными ресурсами, включая:

Порядок организации и осуществления образовательной деятельности по образовательным программам высшего образования - программам бакалавриата, программам специалитета, программам магистратуры (утвержден приказом Министерства образования и науки Российской Федерации от 5 апреля 2017 года №301)

Письмо Министерства образования Российской Федерации №14-55-996ин/15 от 27 ноября 2002 г. "Об активизации самостоятельной работы студентов высших учебных заведений"

Устав федерального государственного автономного образовательного учреждения "Казанский (Приволжский) федеральный университет"

Правила внутреннего распорядка федерального государственного автономного образовательного учреждения высшего профессионального образования "Казанский (Приволжский) федеральный университет"

Локальные нормативные акты Казанского (Приволжского) федерального университета

C# Lessons - <http://functionx.com/csharp1/index.htm>

Введение в программирование на C# 2.0. - <http://www.intuit.ru/department/pl/csharp>

Руководство по языку C# - <https://docs.microsoft.com/ru-ru/dotnet/csharp/index>

### **6. Фонд оценочных средств по дисциплине (модулю)**

Фонд оценочных средств по дисциплине (модулю) включает оценочные материалы, направленные на проверку освоения компетенций, в том числе знаний, умений и навыков. Фонд оценочных средств включает оценочные средства текущего контроля и оценочные средства промежуточной аттестации.

В фонде оценочных средств содержится следующая информация:

- соответствие компетенций планируемым результатам обучения по дисциплине (модулю);
- критерии оценивания сформированности компетенций;
- механизм формирования оценки по дисциплине (модулю);
- описание порядка применения и процедуры оценивания для каждого оценочного средства;
- критерии оценивания для каждого оценочного средства;
- содержание оценочных средств, включая требования, предъявляемые к действиям обучающихся, демонстрируемым результатам, задания различных типов.

Фонд оценочных средств по дисциплине находится в Приложении 1 к программе дисциплины (модулю).

### **7. Перечень литературы, необходимой для освоения дисциплины (модуля)**

Освоение дисциплины (модуля) предполагает изучение основной и дополнительной учебной литературы. Литература может быть доступна обучающимся в одном из двух вариантов (либо в обоих из них):

- в электронном виде - через электронные библиотечные системы на основании заключенных КФУ договоров с правообладателями;

- в печатном виде - в Научной библиотеке им. Н.И. Лобачевского. Обучающиеся получают учебную литературу на абонементе по читательским билетам в соответствии с правилами пользования Научной библиотекой.

Электронные издания доступны дистанционно из любой точки при введении обучающимся своего логина и пароля от личного кабинета в системе "Электронный университет". При использовании печатных изданий библиотечный фонд должен быть укомплектован ими из расчета не менее 0,5 экземпляра (для обучающихся по ФГОС 3++ - не менее 0,25 экземпляра) каждого из изданий основной литературы и не менее 0,25 экземпляра дополнительной литературы на каждого обучающегося из числа лиц, одновременно осваивающих данную дисциплину.

Перечень основной и дополнительной учебной литературы, необходимой для освоения дисциплины (модуля), находится в Приложении 2 к рабочей программе дисциплины. Он подлежит обновлению при изменении условий договоров КФУ с правообладателями электронных изданий и при изменении комплектования фондов Научной библиотеки КФУ.

### 8. Перечень ресурсов информационно-телекоммуникационной сети "Интернет", необходимых для освоения дисциплины (модуля)

Library Genesis. Программирование на C# - <http://libgen.io>

Visual Studio Documentation - <https://docs.microsoft.com/ru-ru/visualstudio/>

Visual Studio Downloads -

<https://www.visualstudio.com/ru/downloads/?rr=https%3A%2F%2Fblogs.msdn.microsoft.com%2Fvisualstudio%2F>

### 9. Методические указания для обучающихся по освоению дисциплины (модуля)

Вид работ	Методические рекомендации
лекции	Лекции сопровождаются презентациями. Презентация предоставлена обучаемым в начале курса и предполагается, что при проведении лекции обучаемые уже знакомы с содержанием лекции, что дает возможность в живом общении обсудить наиболее сложные моменты обсуждаемого материала. Предполагается, что во время лекции имеется возможность запуска компьютерных программ и их корректировка.
практические занятия	Практические занятия проводятся в компьютерных классах Института математики и механики. Предполагается, что на компьютерах установлено необходимое программное обеспечение, в частности, среда разработки MS Visual Studio версии не ниже 2008. Кроме того, с компьютеров имеется доступ в сеть Интернет и локальную сеть Института.
самостоятельная работа	При проведении самостоятельной работы рекомендую использовать сетевые ресурсы научной библиотеки КФУ, а также локальные электронные ресурсы Института математики и механики КФУ. Лекции сопровождаются презентацией, в которой достаточно подробно изложен материал данного курса - презентация предоставляется всем студентам, посещающим курс.
зачет	Зачет проводится индивидуально. Предварительно обучаемый в течении семестра загружает на сетевой диск выполненные компьютерные программы по выбранным темам, а также презентации научных докладов, сделанных во время прохождения курса. На зачете обучаемый получает индивидуальное задание, соответствующее программе.

### 10. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине (модулю), включая перечень программного обеспечения и информационных справочных систем (при необходимости)

Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине (модулю), включая перечень программного обеспечения и информационных справочных систем, представлен в Приложении 3 к рабочей программе дисциплины (модуля).

### 11. Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине (модулю)

Материально-техническое обеспечение образовательного процесса по дисциплине (модулю) включает в себя следующие компоненты:

Помещения для самостоятельной работы обучающихся, укомплектованные специализированной мебелью (столы и стулья) и оснащенные компьютерной техникой с возможностью подключения к сети "Интернет" и обеспечением доступа в электронную информационно-образовательную среду КФУ.

Учебные аудитории для контактной работы с преподавателем, укомплектованные специализированной мебелью (столы и стулья).

Компьютер и принтер для распечатки раздаточных материалов.

Мультимедийная аудитория.

Компьютерный класс.

## **12. Средства адаптации преподавания дисциплины к потребностям обучающихся инвалидов и лиц с ограниченными возможностями здоровья**

При необходимости в образовательном процессе применяются следующие методы и технологии, облегчающие восприятие информации обучающимися инвалидами и лицами с ограниченными возможностями здоровья:

- создание текстовой версии любого нетекстового контента для его возможного преобразования в альтернативные формы, удобные для различных пользователей;
- создание контента, который можно представить в различных видах без потери данных или структуры, предусмотреть возможность масштабирования текста и изображений без потери качества, предусмотреть доступность управления контентом с клавиатуры;
- создание возможностей для обучающихся воспринимать одну и ту же информацию из разных источников - например, так, чтобы лица с нарушениями слуха получали информацию визуально, с нарушениями зрения - аудиально;
- применение программных средств, обеспечивающих возможность освоения навыков и умений, формируемых дисциплиной, за счёт альтернативных способов, в том числе виртуальных лабораторий и симуляционных технологий;
- применение дистанционных образовательных технологий для передачи информации, организации различных форм интерактивной контактной работы обучающегося с преподавателем, в том числе вебинаров, которые могут быть использованы для проведения виртуальных лекций с возможностью взаимодействия всех участников дистанционного обучения, проведения семинаров, выступления с докладами и защиты выполненных работ, проведения тренингов, организации коллективной работы;
- применение дистанционных образовательных технологий для организации форм текущего и промежуточного контроля;
- увеличение продолжительности сдачи обучающимся инвалидом или лицом с ограниченными возможностями здоровья форм промежуточной аттестации по отношению к установленной продолжительности их сдачи:
- продолжительности сдачи зачёта или экзамена, проводимого в письменной форме, - не более чем на 90 минут;
- продолжительности подготовки обучающегося к ответу на зачёте или экзамене, проводимом в устной форме, - не более чем на 20 минут;
- продолжительности выступления обучающегося при защите курсовой работы - не более чем на 15 минут.

Программа составлена в соответствии с требованиями ФГОС ВО и учебным планом по направлению 01.03.01 "Математика" и профилю подготовки "Общий профиль".



Приложение 2  
к рабочей программе дисциплины (модуля)  
Б1.В.ДВ.05.07 Программирование на C# и платформа. Net

**Перечень литературы, необходимой для освоения дисциплины (модуля)**

Направление подготовки: 01.03.01 - Математика

Профиль подготовки: Общий профиль

Квалификация выпускника: бакалавр

Форма обучения: очное

Язык обучения: русский

Год начала обучения по образовательной программе: 2017

**Основная литература:**

1. Медведев, М. А. Программирование на СИ#: учебное пособие / Медведев М.А., Медведев А.Н., - 2-е изд., стер. - Москва: Флинта, Издательство Уральского университета, 2017. - 64 с. ISBN 978-5-9765-3169-7. - Текст : электронный. - URL: <https://znanium.com/catalog/product/948428> (дата обращения: 17.03.2020). - Режим доступа: по подписке.
2. Гуриков, С. Р. Введение в программирование на языке Visual C# : учебное пособие / С.Р. Гуриков. - Москва : ФОРУМ : ИНФРА-М, 2018. - 447 с. - (Высшее образование: Бакалавриат). - ISBN 978-5-16-105882-4. - Текст : электронный. - URL: <https://znanium.com/catalog/product/967691> (дата обращения: 17.03.2020). - Режим доступа: по подписке.
3. Хорев, П. Б. Объектно-ориентированное программирование с примерами на C#: учебное пособие / Хорев П.Б. - Москва : Форум, НИЦ ИНФРА-М, 2016. - 200 с. (Высшее образование: Бакалавриат) ISBN 978-5-00091-144-0. - Текст : электронный. - URL: <https://znanium.com/catalog/product/529350> (дата обращения: 17.03.2020). - Режим доступа: по подписке.

**Дополнительная литература:**

1. Корнеев, В. И. Программирование графики на C++. Теория и примеры : учебное пособие / В.И. Корнеев, Л.Г. Гагарина, М.В. Корнеева. - Москва : ИД 'ФОРУМ' : ИНФРА-М, 2018. - 517 с. - (Высшее образование: Бакалавриат). - [www.dx.doi.org/10.12737/23113](http://www.dx.doi.org/10.12737/23113). - ISBN 978-5-16-106928-8. - Текст : электронный. - URL: <https://znanium.com/catalog/product/981150> (дата обращения: 17.03.2020). - Режим доступа: по подписке.
2. Колдаев, В. Д. Основы алгоритмизации и программирования : учебное пособие / под ред. проф. Л. Г. Гагариной. - Москва : ИД 'ФОРУМ' : ИНФРА-М, 2017. - 416 с. : ил. - (Профессиональное образование). - ISBN 978-5-8199-0279-0. - Текст : электронный. - URL: <https://znanium.com/catalog/product/902236> (дата обращения: 17.03.2020). - Режим доступа: по подписке.
3. Голицына, О. Л. Языки программирования : учебное пособие / О.Л. Голицына, Т.Л. Партыка, И.И. Попов. - 3-е изд., перераб. и доп. - Москва : ФОРУМ : ИНФРА-М, 2018. - 399 с. - (Среднее профессиональное образование). - ISBN 978-5-00091-613-1. - Текст : электронный. - URL: <https://znanium.com/catalog/product/973007> (дата обращения: 17.03.2020). - Режим доступа: по подписке.

Приложение 3  
к рабочей программе дисциплины (модуля)  
Б1.В.ДВ.05.07 Программирование на C# и платформа. Net

**Перечень информационных технологий, используемых для освоения дисциплины (модуля), включая перечень программного обеспечения и информационных справочных систем**

Направление подготовки: 01.03.01 - Математика

Профиль подготовки: Общий профиль

Квалификация выпускника: бакалавр

Форма обучения: очное

Язык обучения: русский

Год начала обучения по образовательной программе: 2017

Освоение дисциплины (модуля) предполагает использование следующего программного обеспечения и информационно-справочных систем:

Операционная система Microsoft Windows 7 Профессиональная или Windows XP (Volume License)

Пакет офисного программного обеспечения Microsoft Office 365 или Microsoft Office Professional plus 2010

Браузер Mozilla Firefox

Браузер Google Chrome

Adobe Reader XI или Adobe Acrobat Reader DC

Kaspersky Endpoint Security для Windows