

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



подписано электронно-цифровой подписью

Программа дисциплины

Основы нанотехнологий / программирование на C# и платформа Net Б2.ДВ.3

Направление подготовки: 010100.62 - Математика

Профиль подготовки: Общий профиль

Квалификация выпускника: бакалавр

Форма обучения: очное

Язык обучения: русский

Автор(ы):

Липачев Е.К.

Рецензент(ы):

Елизаров А.М.

СОГЛАСОВАНО:

Заведующий(ая) кафедрой: Авхадиев Ф. Г.

Протокол заседания кафедры No ____ от " ____ " _____ 201__ г

Учебно-методическая комиссия Института математики и механики им. Н.И. Лобачевского :

Протокол заседания УМК No ____ от " ____ " _____ 201__ г

Регистрационный No 817232314

Казань
2014

Содержание

1. Цели освоения дисциплины
2. Место дисциплины в структуре основной образовательной программы
3. Компетенции обучающегося, формируемые в результате освоения дисциплины /модуля
4. Структура и содержание дисциплины/ модуля
5. Образовательные технологии, включая интерактивные формы обучения
6. Оценочные средства для текущего контроля успеваемости, промежуточной аттестации по итогам освоения дисциплины и учебно-методическое обеспечение самостоятельной работы студентов
7. Литература
8. Интернет-ресурсы
9. Материально-техническое обеспечение дисциплины/модуля согласно утвержденному учебному плану

Программу дисциплины разработал(а)(и) доцент, к.н. (доцент) Липачев Е.К. Кафедра теории функций и приближений отделение математики , Evgeny.Lipachev@kpfu.ru

1. Цели освоения дисциплины

Целями освоения дисциплины (модуля) "Программирование на С# и платформа .NET" являются развитие у студентов развитие у студентов профессиональных компетенций в компьютерных науках.

2. Место дисциплины в структуре основной образовательной программы высшего профессионального образования

Данная учебная дисциплина включена в раздел " Б2.ДВ.3 Общепрофессиональный" основной образовательной программы 010100.62 Математика и относится к дисциплинам по выбору. Осваивается на 4 курсе, 8 семестр.

Цикл Б2.ДВ.3. Дисциплина "Программирование на С# и платформа .NET" входит в цикл профессиональных дисциплин по выбору.

Для прохождения курса необходимы знания компьютерных наук в объеме стандартного университетского курса. Освоение дисциплины "Программирование на С# и платформа .NET" позволит обучающимся познакомиться с одной из самых передовых технологий программирования, получить необходимые знания по использованию компьютерных средств при выполнении курсовых и дипломных работ. Изучается на 4 курсе (8 семестр)

3. Компетенции обучающегося, формируемые в результате освоения дисциплины /модуля

В результате освоения дисциплины формируются следующие компетенции:

Шифр компетенции	Расшифровка приобретаемой компетенции
ОК-10 (общекультурные компетенции)	мение находить, анализировать и контекстно обрабатывать научно-техническую информацию
ОК-11 (общекультурные компетенции)	фундаментальной подготовкой по основам профессиональных знаний и готовностью к использованию их в профессиональной деятельности
ОК-12 (общекультурные компетенции)	навыками работы с компьютером
ОК-13 (общекультурные компетенции)	базовыми знаниями в областях информатики и современных информационных технологий, навыки использования программных средств и навыки работы в компьютерных сетях, умение создавать базы данных и использовать ресурсы Интернет
ОК-6 (общекультурные компетенции)	способностью применять знания на практике
ОК-7 (общекультурные компетенции)	исследовательскими навыками
ОК-8 (общекультурные компетенции)	способностью приобретать новые знания, используя современные образовательные и информационные технологии

Шифр компетенции	Расшифровка приобретаемой компетенции
ОК-9 (общекультурные компетенции)	способностью понимать сущность и значение информации в развитии современного общества, соблюдением основных требований информационной безопасности, в том числе защиты государственных интересов и приоритетов
ПК-11 (профессиональные компетенции)	самостоятельным построением алгоритма и его анализ
ПК-12 (профессиональные компетенции)	пониманием того, что фундаментальное знание является основой компьютерных наук
ПК-14 (профессиональные компетенции)	контекстной обработкой информации
ПК-17 (профессиональные компетенции)	умением извлекать полезную научно-техническую информацию из электронных библиотек, реферативных журналов, сети Интернет
ПК-19 (профессиональные компетенции)	владением методом алгоритмического моделирования при анализе постановок математических задач
ПК-20 (профессиональные компетенции)	владением методами математического и алгоритмического моделирования при решении прикладных задач
ПК-21 (профессиональные компетенции)	владением методами математического и алгоритмического моделирования при анализе теоретических проблем и задач
ПК-24 (профессиональные компетенции)	ладением методами математического и алгоритмического моделирования при анализе управленческих задач в научно-технической сфере
ПК-27 (профессиональные компетенции)	умением точно представить математические знания в устной форме
ПК-7 (профессиональные компетенции)	умением грамотно пользоваться языком предметной области

В результате освоения дисциплины студент:

1. должен знать:

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

2. должен уметь:

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

3. должен владеть:

приемами программирования в среде MS Visual Studio.

4. должен продемонстрировать способность и готовность:

разрабатывать приложения на платформе .NET на языке C# в среде MS Visual Studio 2010, 2012.

4. Структура и содержание дисциплины/ модуля

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

Форма промежуточного контроля дисциплины зачет в 8 семестре.

Суммарно по дисциплине можно получить 100 баллов, из них текущая работа оценивается в 50 баллов, итоговая форма контроля - в 50 баллов. Минимальное количество для допуска к зачету 28 баллов.

86 баллов и более - "отлично" (отл.);

71-85 баллов - "хорошо" (хор.);

55-70 баллов - "удовлетворительно" (удов.);

54 балла и менее - "неудовлетворительно" (неуд.).

4.1 Структура и содержание аудиторной работы по дисциплине/ модулю Тематический план дисциплины/модуля

N	Раздел Дисциплины/ Модуля	Семестр	Неделя семестра	Виды и часы аудиторной работы, их трудоемкость (в часах)			Текущие формы контроля
				Лекции	Практические занятия	Лабораторные работы	
1.	Тема 1. Обзор платформы Microsoft .Net	8	1	1	0	0	домашнее задание
2.	Тема 2. Обзор C#	8	2	1	0	0	домашнее задание
3.	Тема 3. Использование размерных типов	8	3	1	1	0	устный опрос
4.	Тема 4. Выражения и исключения	8	4	1	1	0	домашнее задание
5.	Тема 5. Методы и параметры	8	5	1	1	0	контрольная точка устный опрос
6.	Тема 6. Массивы	8	6	2	2	0	контрольная работа домашнее задание
7.	Тема 7. Принципы объектно-ориентированного программирования	8	7-8	1	2	0	домашнее задание
8.	Тема 8. Использование ссылочных типов	8	9	1	1	0	домашнее задание
9.	Тема 9. Создание и разрушение объектов	8	10	1	1	0	устный опрос
10.	Тема 10. Наследование в C#	8	11-12	1	1	0	контрольная работа домашнее задание

N	Раздел Дисциплины/ Модуля	Семестр	Неделя семестра	Виды и часы аудиторной работы, их трудоемкость (в часах)			Текущие формы контроля
				Лекции	Практические занятия	Лабораторные работы	
11.	Тема 11. Агрегирование, пространства имен и область видимости	8	13	1	1	0	домашнее задание
12.	Тема 12. Операции, делегаты и события	8	14	1	1	0	домашнее задание
13.	Тема 13. Свойства и индексы	8	15	1	1	0	домашнее задание
14.	Тема 14. Атрибуты	8	16	1	1	0	устный опрос
15.	Тема 15. Ввод и вывод в C#	8	17-18	1	2	0	домашнее задание
	Тема . Итоговая форма контроля	8		0	0	0	зачет
	Итого			16	16	0	

4.2 Содержание дисциплины

Тема 1. Обзор платформы Microsoft .Net

лекционное занятие (1 часа(ов)):

.NET Framework программная платформа, выпущенная компанией Microsoft в 2002 году. Основой платформы является общезыковая среда исполнения Common Language Runtime (CLR). Функциональные возможности CLR доступны в любых языках программирования, использующих эту среду. Программа для .NET Framework переводится компилятором в единый для .NET промежуточный байт-код Common Intermediate Language (CIL). В терминах .NET получается сборка, англ. assembly. Затем код либо исполняется виртуальной машиной Common Language Runtime (CLR). Использование виртуальной машины предпочтительно, так как избавляет разработчиков от необходимости заботиться об особенностях аппаратной части. В случае использования виртуальной машины CLR, встроенный в неё JIT-компилятор (just in time) преобразует промежуточный байт-код в машинные коды нужного процессора. Современная технология динамической компиляции позволяет достигнуть высокого уровня быстродействия. Виртуальная машина CLR также сама заботится о базовой безопасности, управлении памятью и системе исключений. В CLI описаны типы данных .NET, формат метаданных о структуре программы, система исполнения байт-кода. Объектные классы .NET, доступные для всех поддерживаемых языков программирования, содержатся в библиотеке Framework Class Library (FCL). В FCL входят классы Windows Forms, ADO.NET, ASP.NET, Language Integrated Query, Windows Presentation Foundation, Windows Communication Foundation и другие. Ядро FCL называется Base Class Library (BCL).

Тема 2. Обзор C#

лекционное занятие (1 часа(ов)):

C# разработан в 1998-2001 годах группой инженеров под руководством Андерса Хейлсберга в компании Microsoft. Стандартизирован как ECMA-334 и ISO/IEC 23270. C# относится к семье языков с C-подобным синтаксисом, из них его синтаксис наиболее близок к C++ и Java. Язык имеет статическую типизацию, поддерживает полиморфизм, перегрузку операторов (в том числе операторов явного и неявного приведения типа), делегаты, атрибуты, события, свойства, обобщённые типы и методы, итераторы, анонимные функции с поддержкой замыканий, LINQ, исключения, комментарии в формате XML.

Тема 3. Использование размерных типов

лекционное занятие (1 часа(ов)):

C# подразделяет типы на два вида: встроенные типы, которые определены в языке, и определяемые пользователем типы, которые выбирает программист. C# также подразделяет типы на две другие категории: размерные и ссылочные. Основное различие между ними - это способ, которым их значения сохраняются в памяти. Размерные типы сохраняют свое фактическое значение в стеке. Ссылочные типы хранят в стеке лишь адрес объекта, а сам объект сохраняется в куче.

практическое занятие (1 часа(ов)):

Составление программ в Visual Studio с использованием типов int, double.

Тема 4. Выражения и исключения

лекционное занятие (1 часа(ов)):

Обзор выражений языка

практическое занятие (1 часа(ов)):

Составление программ в Visual Studio

Тема 5. Методы и параметры

лекционное занятие (1 часа(ов)):

Основные сведения о методах и параметрах

практическое занятие (1 часа(ов)):

Составление программ в Visual Studio

Тема 6. Массивы

лекционное занятие (2 часа(ов)):

Особенности создания массивов. Динамические массивы. Многомерные массивы. Коллекции

практическое занятие (2 часа(ов)):

Составление программ в Visual Studio - работа с матрицами

Тема 7. Принципы объектно-ориентированного программирования

лекционное занятие (1 часа(ов)):

Особенности ООП в языке C# Отметить, что в C# все - классы!

практическое занятие (2 часа(ов)):

Составление программ в Visual Studio, содержащих несколько классов.

Тема 8. Использование ссылочных типов

лекционное занятие (1 часа(ов)):

Переменные ссылочных типов, называемые объектами, сохраняют ссылки на фактические данные. Для объявления ссылочных типов используются ключевые слова: class interface delegate

практическое занятие (1 часа(ов)):

Составление программ в Visual Studio

Тема 9. Создание и разрушение объектов

лекционное занятие (1 часа(ов)):

Поскольку классы являются ссылочными типами, в переменной объекта класса хранится ссылка на адрес объекта в управляемой куче. Если первому объекту назначен второй объект того же типа, обе переменные ссылаются на объект, расположенный по данному адресу. Каждый раз, когда создается класс или структура, вызывается конструктор. Класс или структура может иметь несколько конструкторов, принимающих различные аргументы. Конструкторы позволяют программисту задавать значения по умолчанию, ограничивать число установок и писать код, который является гибким и удобным для чтения. Деструкторы используются для уничтожения экземпляров классов.

практическое занятие (1 часа(ов)):

Составление программ в Visual Studio

Тема 10. Наследование в C#

лекционное занятие (1 часа(ов)):

Чтобы построить (в терминах данных или поведения) класс на основе другого класса, применяется наследование. Оно принадлежит к правилам заменяемости, а именно к тому, которое гласит, что производный класс может быть заменен базовым. Чтобы унаследовать один класс от другого, используется синтаксис: `class <производный_класс> <базовый_класс>` Концептуально, производный класс является специализацией базового класса. При определении класса для наследования от другого класса, производный класс явно получает все члены базового класса, за исключением его конструкторов и деструкторов. Производный класс может таким образом повторно использовать код в базовом классе без необходимости в его повторной реализации. В производном классе можно добавить больше членов. Таким образом, производный класс расширяет функциональность базового класса. Когда базовый класс объявляет метод как виртуальный, производный класс может переопределить метод с помощью своей собственной реализации. Если базовый класс объявляет член как абстрактный, то этот метод должен быть переопределен в любом неабстрактном классе, который прямо наследует от этого класса. Если производный класс сам является абстрактным, то он наследует абстрактные члены, не реализуя их. Абстрактные и виртуальные члены являются основой для полиморфизма, который является второй основной характеристикой объектно-ориентированного программирования. Можно объявить класс как абстрактный, если необходимо предотвратить прямое создание экземпляров с помощью ключевого слова `new`. При таком подходе класс можно использовать, только если новый класс является производным от него. Абстрактный класс может содержать один или несколько сигнатур методов, которые сами объявлены в качестве абстрактных. Эти сигнатуры задают параметры и возвращают значение, но не имеют реализации (тела метода). Абстрактному классу необязательно содержать абстрактные члены; однако, если класс все же содержит абстрактный член, то сам класс должен быть объявлен в качестве абстрактного. Производные классы, которые сами не являются абстрактными, должны предоставить реализацию для любых абстрактных методов из абстрактного базового класса.

практическое занятие (1 часа(ов)):

Составление программ в Visual Studio

Тема 11. Агрегирование, пространства имен и область видимости

лекционное занятие (1 часа(ов)):

Область видимости членов класса (то есть область кода, из которой к ним можно обращаться по некавалифицированному имени без указания имени класса или объекта) не зависит от их области доступа, и всегда совпадает с кодом методов класса.

практическое занятие (1 часа(ов)):

Составление программ в Visual Studio

Тема 12. Операции, делегаты и события

лекционное занятие (1 часа(ов)):

Делегат это аналог указателя на функцию в других языках программирования. Делегаты выполняют ту же самую функцию, но более удобно и безопасно, то есть основное назначение это - обратный вызов (отправление сообщения объектам, которые их породили). Делегат определяется так `public delegate void Play(object music, int volume)`; делегат в момент компиляции превращается в класс, наследуемый от `system.Multicastdelegate`

практическое занятие (1 часа(ов)):

Составление программ в Visual Studio

Тема 13. Свойства и индексаторы

лекционное занятие (1 часа(ов)):

Свойство - это член, предоставляющий гибкий механизм для чтения, записи или вычисления значения частного (`private`) поля. Свойства можно использовать, как если бы они являлись открытыми членами данных, хотя в действительности они являются специальными методами, называемыми методами доступа. Индексаторы позволяют индексировать экземпляры класса или структуры так же, как массивы. Индексаторы напоминают свойства, но их методы доступа принимают параметры.

практическое занятие (1 часа(ов)):

Составление программ в Visual Studio

Тема 14. Атрибуты

лекционное занятие (1 часа(ов)):

Атрибуты находятся в исполняемом файле и могут оттуда при необходимости извлекаться. Все атрибуты являются классами (потомками класса System.Attribute). Атрибуты делятся на предопределенные (встроенные) и пользовательские (которые пишет программист). Встроенные атрибуты могут использоваться при сериализации (сохранении в поток) данных класса. Атрибуты в C# заключаются в квадратные скобки.

практическое занятие (1 часа(ов)):

Составление программ в Visual Studio

Тема 15. Ввод и вывод в C#

лекционное занятие (1 часа(ов)):

Схема работы с файлами. Учет кодировки при чтении текстовых файлов.

практическое занятие (2 часа(ов)):

Составление программ в Visual Studio

4.3 Структура и содержание самостоятельной работы дисциплины (модуля)

N	Раздел Дисциплины	Семестр	Неделя семестра	Виды самостоятельной работы студентов	Трудоемкость (в часах)	Формы контроля самостоятельной работы
1.	Тема 1. Обзор платформы Microsoft .Net	8	1	подготовка домашнего задания	2	домашнее задание
2.	Тема 2. Обзор C#	8	2	подготовка домашнего задания	2	домашнее задание
3.	Тема 3. Использование размерных типов	8	3	подготовка к устному опросу	2	устный опрос
4.	Тема 4. Выражения и исключения	8	4	подготовка домашнего задания	2	домашнее задание
5.	Тема 5. Методы и параметры	8	5	подготовка к контрольной точке	2	контрольная точка
				подготовка к устному опросу	2	устный опрос
6.	Тема 6. Массивы	8	6	подготовка домашнего задания	1	домашнее задание
				подготовка к контрольной работе	1	контрольная работа
7.	Тема 7. Принципы объектно-ориентированного программирования	8	7-8	подготовка домашнего задания	4	домашнее задание
8.	Тема 8. Использование ссылочных типов	8	9	подготовка домашнего задания	2	домашнее задание
9.	Тема 9. Создание и разрушение объектов	8	10	подготовка к устному опросу	2	устный опрос

N	Раздел Дисциплины	Семестр	Неделя семестра	Виды самостоятельной работы студентов	Трудоемкость (в часах)	Формы контроля самостоятельной работы
10.	Тема 10. Наследование в C#	8	11-12	подготовка домашнего задания	2	домашнее задание
				подготовка к контрольной работе	2	контрольная работа
11.	Тема 11. Агрегирование, пространства имен и область видимости	8	13	подготовка домашнего задания	2	домашнее задание
12.	Тема 12. Операции, делегаты и события	8	14	подготовка домашнего задания	4	домашнее задание
13.	Тема 13. Свойства и индексы	8	15	подготовка домашнего задания	2	домашнее задание
14.	Тема 14. Атрибуты	8	16	подготовка к устному опросу	2	устный опрос
15.	Тема 15. Ввод и вывод в C#	8	17-18	подготовка домашнего задания	4	домашнее задание
	Итого				40	

5. Образовательные технологии, включая интерактивные формы обучения

Активные и интерактивные формы.

6. Оценочные средства для текущего контроля успеваемости, промежуточной аттестации по итогам освоения дисциплины и учебно-методическое обеспечение самостоятельной работы студентов

Тема 1. Обзор платформы Microsoft .Net

домашнее задание , примерные вопросы:

Изучение материала

Тема 2. Обзор C#

домашнее задание , примерные вопросы:

Изучение материала

Тема 3. Использование размерных типов

устный опрос , примерные вопросы:

Опрос по билетам

Тема 4. Выражения и исключения

домашнее задание , примерные вопросы:

Изучение материала

Тема 5. Методы и параметры

контрольная точка , примерные вопросы:

Предоставить практический пример

устный опрос , примерные вопросы:

Опрос по билетам

Тема 6. Массивы

домашнее задание , примерные вопросы:

Изучение материала

контрольная работа , примерные вопросы:

Решение задач на матрицы

Тема 7. Принципы объектно-ориентированного программирования

домашнее задание , примерные вопросы:

Изучение материала

Тема 8. Использование ссылочных типов

домашнее задание , примерные вопросы:

Изучение материала

Тема 9. Создание и разрушение объектов

устный опрос , примерные вопросы:

Опрос по билетам

Тема 10. Наследование в C#

домашнее задание , примерные вопросы:

Изучение материала

контрольная работа , примерные вопросы:

Создание программы с полиморфным поведением методов

Тема 11. Агрегирование, пространства имен и область видимости

домашнее задание , примерные вопросы:

Изучение материала

Тема 12. Операции, делегаты и события

домашнее задание , примерные вопросы:

Изучение материала

Тема 13. Свойства и индексаторы

домашнее задание , примерные вопросы:

Изучение материала

Тема 14. Атрибуты

устный опрос , примерные вопросы:

Опрос по билетам

Тема 15. Ввод и вывод в C#

домашнее задание , примерные вопросы:

Изучение материала. Учет кодировки при чтении и записи файлов.

Тема . Итоговая форма контроля

Примерные вопросы к зачету:

В течение семестра студенты делают доклады на семинарах по темам, перечисленным в приведенной программе.

Вопросы к зачету

1. Основные типы
2. Базовые выражения языка
3. Операторы цикла
4. Методы, статические методы, передача параметров
5. Создание массивов, инициализация массивов
6. Операции с массивами
7. Передача массивов как параметров

8. Динамические массивы
9. Создание классов и объектов
10. Определение ссылочных типов
11. Как создаются объекты, конструкторы, деструкторы
12. Наследование
13. Виртуальные методы
14. Полиморфное поведение объектов
15. Пространства имен, области видимости объектов
16. Делегаты.
17. Правила создания делегатов
18. Определение свойств, свойства только для чтения
19. Пример применения атрибутов
20. Операции с файлами
21. Учет кодировки при работе с текстовыми файлами

7.1. Основная литература:

1. Программирование на языке C#: учебно-методическое пособие / И. Л. Александрова, Д.Н. Тумаков; Казан. федер. ун-т, Ин-т вычисл. мат. и информ. технологий. Казань: [б. и.], 2011. 103 с.
2. Технология программирования. Базовые конструкции C/C++: учебно-справочное пособие / Е. К. Липачёв; Казан. федер. ун-т. Казань: [Казанский университет], 2012. 139, [1] с
3. Немцова Т. И. Программирование на языке высокого уровня. Программир. на языке C++: Уч. пос. /Под ред. Л.Г.Гагариной - М.: ИД ФОРУМ: ИНФРА-М, 2012. - 512 с. // <http://znanium.com/bookread.php?book=244875>
4. Липачёв Е.К. Технология программирования. Базовые конструкции C/C++. Учебно-справочное пособие // Казань: Казан.ун-т, 2012. " 142 с. :http://kpfu.ru/publication?p_id=47437

7.2. Дополнительная литература:

1. Программирование на C++, C++. NET/C# и .NET компоненты / В. И. Медведев. 2-е изд., [испр.]. Казань: Мастер Лайн, 2007. 295 с.
2. Литвиненко Н. А. Технология программирования на C++. Win32 API-приложения. ? СПб.: БХВ-Петербург, 2010. ? 280 с.: <http://znanium.com/bookread.php?book=351463>

7.3. Интернет-ресурсы:

Введение в программирование на C# 2.0 - <http://www.intuit.ru/studies/courses/109/109/info>
Основы программирования на C# - <http://www.intuit.ru/studies/courses/2247/18/info>
Платформа .NET Framework - <http://msdn.microsoft.com/ru-ru/vstudio/aa496123>
Сайт Microsoft Материалы для создания, обслуживания и изучения приложений - <http://msdn.microsoft.com/ru-ru/vstudio/>
Учебные материалы FunctionX - www.functionx.com

8. Материально-техническое обеспечение дисциплины(модуля)

Освоение дисциплины "Основы нанотехнологий / программирование на C# и платформа Net" предполагает использование следующего материально-технического обеспечения:

Мультимедийная аудитория, вместимостью более 60 человек. Мультимедийная аудитория состоит из интегрированных инженерных систем с единой системой управления, оснащенная современными средствами воспроизведения и визуализации любой видео и аудио информации, получения и передачи электронных документов. Типовая комплектация мультимедийной аудитории состоит из: мультимедийного проектора, автоматизированного проекционного экрана, акустической системы, а также интерактивной трибуны преподавателя, включающей тач-скрин монитор с диагональю не менее 22 дюймов, персональный компьютер (с техническими характеристиками не ниже Intel Core i3-2100, DDR3 4096Mb, 500Gb), конференц-микрофон, беспроводной микрофон, блок управления оборудованием, интерфейсы подключения: USB, audio, HDMI. Интерактивная трибуна преподавателя является ключевым элементом управления, объединяющим все устройства в единую систему, и служит полноценным рабочим местом преподавателя. Преподаватель имеет возможность легко управлять всей системой, не отходя от трибуны, что позволяет проводить лекции, практические занятия, презентации, вебинары, конференции и другие виды аудиторной нагрузки обучающихся в удобной и доступной для них форме с применением современных интерактивных средств обучения, в том числе с использованием в процессе обучения всех корпоративных ресурсов. Мультимедийная аудитория также оснащена широкополосным доступом в сеть интернет. Компьютерное оборудование имеет соответствующее лицензионное программное обеспечение.

Компьютерный класс, представляющий собой рабочее место преподавателя и не менее 15 рабочих мест студентов, включающих компьютерный стол, стул, персональный компьютер, лицензионное программное обеспечение. Каждый компьютер имеет широкополосный доступ в сеть Интернет. Все компьютеры подключены к корпоративной компьютерной сети КФУ и находятся в едином домене.

Для проведения лекций необходимы: проектор, экран, ноутбук. Практические занятия проводятся с использованием персональных компьютеров, на которых установлены: среда разработки MS Visual Studio 2010. Лекции выставляются в Интернет - на сайт факультета.

Программа составлена в соответствии с требованиями ФГОС ВПО и учебным планом по направлению 010100.62 "Математика" и профилю подготовки Общий профиль .

Автор(ы):

Липачев Е.К. _____

"__" _____ 201__ г.

Рецензент(ы):

Елизаров А.М. _____

"__" _____ 201__ г.