

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное автономное образовательное учреждение
высшего образования
"Казанский (Приволжский) федеральный университет"
Институт вычислительной математики и информационных технологий



УТВЕРЖДАЮ

Проректор
по образовательной деятельности КФУ
Проф. Таюрский Д.А.

_____ 20__ г.

Программа дисциплины
Язык С# и технологии .Net Б1.В.ДВ.22

Направление подготовки: 01.03.02 - Прикладная математика и информатика

Профиль подготовки: Системное программирование

Квалификация выпускника: бакалавр

Форма обучения: очное

Язык обучения: русский

Автор(ы):

Мухтарова Т.М.

Рецензент(ы):

-

СОГЛАСОВАНО:

Заведующий(ая) кафедрой: Миссаров М. Д.

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

Учебно-методическая комиссия Института вычислительной математики и информационных технологий:

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

Регистрационный No

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) старший преподаватель, б/с Мухтарова Т.М. кафедра анализа данных и исследования операций отделение фундаментальной информатики и информационных технологий , Tatyana.Moukhtarova@kpfu.ru

1. Цели освоения дисциплины

Целью изучения данной дисциплины является формирование у будущих специалистов практических навыков по основам объектно-ориентированного программирования. Объектно-ориентированное программирование - это современная технология, на основе которой разрабатывается большинство информационных систем. В качестве основы обучения выбран язык программирования С# - современный и динамично развивающийся язык.

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

Данная учебная дисциплина включена в раздел "Б1.В.ДВ.22 Дисциплины (модули)" основной образовательной программы 01.03.02 Прикладная математика и информатика и относится к дисциплинам по выбору. Осваивается на 3 курсе, 5 семестр.

Для изучения данного курса студенты должны пройти курс 'Информатика', 'Лабораторный практикум по информационно-компьютерным технологиям', 'Языки и методы программирования'. Материалы данного курса будут активно использоваться при написании курсовых и дипломных работ.

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

В результате освоения дисциплины формируются следующие компетенции:

Шифр компетенции	Расшифровка приобретаемой компетенции
ОПК-2 (профессиональные компетенции)	способность приобретать новые научные и профессиональные знания, используя современные образовательные и информационные технологии
ОПК-1 (профессиональные компетенции)	способность использовать базовые знания естественных наук, математики и информатики, основные факты, концепции, принципы теорий, связанных с прикладной математикой и информатикой
ПК-1 (профессиональные компетенции)	способность собирать, обрабатывать и интерпретировать данные современных научных исследований, необходимые для формирования выводов по соответствующим научным исследованиям
ПК-2 (профессиональные компетенции)	способность понимать, совершенствовать и применять современный математический аппарат

В результате освоения дисциплины студент:

1. должен знать:

- синтаксис языка программирования С#
- принципы объектно-ориентированного программирования
- основные классы из библиотеки классов языка программирования С# для создания объектно-ориентированных приложений.

2. должен уметь:

- создавать программы на языке программирования С# , разрабатывая собственные классы, а также, используя классы из библиотек этого языка.

3. должен владеть:

- навыками разработки консольных приложений в стиле объектно-ориентированного программирования на С#;
- навыками разработки консольных приложений с использованием внешних источников данных - текстовых файлов.

4. должен демонстрировать способность и готовность:

- применять полученные знания и навыки в процессе дальнейшего обучения при написании курсовых и дипломной работ, а также в своей дальнейшей профессиональной деятельности.

4. Структура и содержание дисциплины/ модуля

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

Форма промежуточного контроля дисциплины: зачет в 5 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	Практи- ческие занятия	Лабора- торные работы	
1.	Тема 1. Язык С # . Особенности работы .NET Framework. Типы данных. Операции и операторы языка. Особенности ввода-вывода.	4		0	0	4	Компьютерная программа
2.	Тема 2. Принципы объектно-ориентированного программирования и их реализация в С#. Понятие класса. Инкапсуляция. Конструкторы и деструкторы классов, их необходимость.	4		0	0	2	Компьютерная программа
3.	Тема 3. Свойства и методы класса. Перегрузка операций.	4		0	0	6	Компьютерная программа

N	Раздел Дисциплины/ Модуля	Семестр	Неделя семестра	Виды и часы аудиторной работы, их трудоемкость (в часах)			Текущие формы контроля
				Лекции	Практи- ческие занятия	Лабора- торные работы	
4.	Тема 4. Наследование. Виртуальные функции и абстрактные классы. Интерфейсы.	4		0	0	6	Контрольная работа
5.	Тема 5. Делегаты и события.	4		0	0	4	Компьютерная программа
6.	Тема 6. Обобщения. Основные понятия. Уточнения, используемые в обобщениях. Обобщенные интерфейсы, методы, делегаты.	4		0	0	4	Компьютерная программа
7.	Тема 7. Работа с коллекциями из библиотеки классов С#, их использование для решения всевозможных задач.	4		0	0	4	Контрольная работа
8.	Тема 8. Использование внешних источников данных. Работа с текстовыми и двоичными файлами.	4		0	0	4	Компьютерная программа
9.	Тема 9. Обработка исключений.	4		0	0	2	Компьютерная программа
.	Тема . Итоговая форма контроля	5		0	0	0	Зачет
	Итого			0	0	36	

4.2 Содержание дисциплины

Тема 1. Язык С # . Особенности работы .NET Framework. Типы данных. Операции и операторы языка. Особенности ввода-вывода.

лабораторная работа (4 часа(ов)):

Об особенностях платформы .NET - новый формат выполнения программных модулей с использованием промежуточного языка программирования и метаданных; создание специальной VM CLR; общая библиотека классов FCL. Типы данных - структурные и ссылочные. Различие в использовании структурных и ссылочных типов данных. Массивы как ссылочный тип данных. Многомерные массивы. Функции. Способы передачи параметров по ссылке и по значению. Возвращаемые значения в параметрах. Отличия в передаче параметров структурных и ссылочных типов данных.

Тема 2. Принципы объектно-ориентированного программирования и их реализация в С#. Понятие класса. Инкапсуляция. Конструкторы и деструкторы классов, их необходимость.

лабораторная работа (2 часа(ов)):

Разработка собственных классов, содержащих конструкторы, методы ввода-вывода. Примеры использования классов в программах.

Тема 3. Свойства и методы класса. Перегрузка операций.

лабораторная работа (6 часа(ов)):

Переопределение операторов. Свойства, индексаторы, итераторы. Пример необходимости применения статических членов класса.

Тема 4. Наследование. Виртуальные функции и абстрактные классы. Интерфейсы.

лабораторная работа (6 часа(ов)):

Иерархия классов. Наследование, его виды, примеры. Виртуальные функции как механизм позднего связывания. Особенности использования виртуальных функций в классах-наследниках. Интерфейсы и абстрактные классы.

Тема 5. Делегаты и события.

лабораторная работа (4 часа(ов)):

Определение функционального типа и его экземпляров. Функции высших порядков. Делегаты и анонимные методы - лямбда-выражения. Делегаты как свойства. Операции над делегатами. Пример комбинирования делегатов. События. Обработчики событий. Создание классов с событиями.

Тема 6. Обобщения. Основные понятия. Уточнения, используемые в обобщениях.

Обобщенные интерфейсы, методы, делегаты.

лабораторная работа (4 часа(ов)):

Обобщения как параметризованный тип данных. Класс object и его использование. Преимущества применения обобщенных типов данных. Разработка классов-обобщений. Сконструированный тип - созданный при подстановке аргумента типа вместо обобщенного. Уточнения, используемые в обобщениях. Обобщенные интерфейсы. Обобщенные методы. Обобщенные делегаты.

Тема 7. Работа с коллекциями из библиотеки классов С#, их использование для решения всевозможных задач.

лабораторная работа (4 часа(ов)):

Классы из пространства имен Collections. Коллекции-контейнеры. Классы из пространства имен Collections.Generic - различия в использовании. Использование нумераторов. Сравнение элементов коллекций. Коллекции-шаблоны. Примеры задач на использование коллекций.

Тема 8. Использование внешних источников данных. Работа с текстовыми и двоичными файлами.

лабораторная работа (4 часа(ов)):

Файловый ввод-вывод. Байтовый поток. Символьный поток. Двоичный поток. Примеры решения задач работы с файлами.

Тема 9. Обработка исключений.

лабораторная работа (2 часа(ов)):

Обработка исключений. Использование стандартных исключений. Класс Exception. Создание собственных исключений.

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. Язык С # . Особенности работы .NET Framework. Типы данных. Операции и операторы языка. Особенности ввода-вывода.	4		Разработка программ с применением полученных знаний по данной теме.	4	Компьютерная программа

N	Раздел дисциплины	Семестр	Неделя семестра	Виды самостоятельной работы студентов	Трудоемкость (в часах)	Формы контроля самостоятельной работы
2.	Тема 2. Принципы объектно-ориентированного программирования и их реализация в С#. Понятие класса. Инкапсуляция. Конструкторы и деструкторы классов, их необходимость.	4		Разработка программ с применением полученных знаний по данной теме.	2	Компьютерная программа
3.	Тема 3. Свойства и методы класса. Перегрузка операций.	4		Разработка программ с применением полученных знаний по данной теме.	4	Компьютерная программа
4.	Тема 4. Наследование. Виртуальные функции и абстрактные классы. Интерфейсы.	4		подготовка к контрольной работе	4	Контрольная работа
5.	Тема 5. Делегаты и события.	4		Разработка программ с применением полученных знаний по данной теме.	2	Компьютерная программа
6.	Тема 6. Обобщения. Основные понятия. Уточнения, используемые в обобщениях. Обобщенные интерфейсы, методы, делегаты.	4		Разработка программ с применением полученных знаний по данной теме.	6	Компьютерная программа

№	Раздел дисциплины	Семестр	Неделя семестра	Виды самостоятельной работы студентов	Трудоемкость (в часах)	Формы контроля самостоятельной работы
7.	Тема 7. Работа с коллекциями из библиотеки классов C#, их использование для решения всевозможных задач.	4		подготовка к контрольной работе	4	Контрольная работа
8.	Тема 8. Использование внешних источников данных. Работа с текстовыми и двоичными файлами.	4		Разработка программ с применением полученных знаний по данной теме.	8	Компьютерная программа
9.	Тема 9. Обработка исключений.	4		Разработка программ с применением полученных знаний по данной теме. Разработка программ с применением полученных знаний по данной теме.	2	Компьютерная программа
	Итого				36	

5. Образовательные технологии, включая интерактивные формы обучения

Аудиторная работа по данной дисциплине проводится в форме лабораторных занятий. Занятия посвящены выработке навыков составления программ в объектно-ориентированном стиле на языке программирования C#. Занятия проходят в компьютерном классе, оснащенный интерактивной доской, которая используется для наглядности процесса создания и разработки программ. В процессе разработки программ студенты вовлекаются в совместное обсуждение различных подходов по реализации предлагаемых для решения задач, а также самостоятельно выполняют различные задания, консультируясь, при необходимости с преподавателем.

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

Тема 1. Язык C # . Особенности работы .NET Framework. Типы данных. Операции и операторы языка. Особенности ввода-вывода.

Компьютерная программа , примерные вопросы:

Отрабатываются навыки решения различных задач в стиле структурного программирования на языке C#. Пример типового задания. Написать функцию для перемножения двух матриц. Написать функцию упорядочивания элементов массива методом пирамидальной сортировки.

Тема 2. Принципы объектно-ориентированного программирования и их реализация в C#. Понятие класса. Инкапсуляция. Конструкторы и деструкторы классов, их необходимость.

Компьютерная программа , примерные вопросы:

Отрабатываются навыки разработки и применения классов в программах на языке C#. Пример типового задания. Разработать класс "Дробь". Определить конструкторы, методы ввода-вывода и реализации арифметических операций.

Тема 3. Свойства и методы класса. Перегрузка операций.

Компьютерная программа , примерные вопросы:

Отрабатываются навыки разработки и применения классов в программах на языке C#. Пример типового задания. В классе "Дробь" определить свойства и переопределить арифметические операции.

Тема 4. Наследование. Виртуальные функции и абстрактные классы. Интерфейсы.

Контрольная работа , примерные вопросы:

Типовое задание на контрольную работу. Разработать класс "Отрезок на прямой". Определить конструкторы, свойства, методы ввода-вывода и операции пересечения и объединения отрезков. Типовой пример задания по теме. Создать абстрактный класс Function. Класс должен включать различные абстрактные методы (нахождение максимума, минимума, корней функции на заданном интервале, вычисление производной функции в заданной точке). В классах наследниках реализовать абстрактные методы для различных видов функций.

Тема 5. Делегаты и события.

Компьютерная программа , примерные вопросы:

Отрабатываются навыки работы с функциональными типами данных. Пример типового задания. Найти корни уравнения различными методами: методом деления отрезка пополам, методом Ньютона, методом хорд. В метод передать функцию вычисления значения в виде параметра с использованием делегата. Создать интерфейс пользователя с возможностью задания вида функции и выбора метода решения уравнения.

Тема 6. Обобщения. Основные понятия. Уточнения, используемые в обобщениях. Обобщенные интерфейсы, методы, делегаты.

Компьютерная программа , примерные вопросы:

Отрабатываются навыки разработки и применения классов с использованием обобщений. Типовой пример. Разработать класс "Стек" с возможностью хранения данных различных типов. Определить Методы добавления-извлечения элементов из стека, печати содержимого стека. Решить задачи перевода числа в двоичную систему и проверки правильности скобочного выражения с использованием созданного класса.

Тема 7. Работа с коллекциями из библиотеки классов C#, их использование для решения всевозможных задач.

Контрольная работа , примерные вопросы:

Отрабатываются навыки применения библиотечных классов-коллекций. Пример типового задания. Преобразовать арифметическое выражение, заданное в виде строки в реверсированную форму. Осуществить вычисление арифметического выражения. Использовать классы "Стек" и "Очередь".

Тема 8. Использование внешних источников данных. Работа с текстовыми и двоичными файлами.

Компьютерная программа , примерные вопросы:

Решаются задачи с использованием внешних источников данных в виде текстовых и бинарных файлов. Пример типового задания. В текстовом файле, хранящем информацию о книгах в библиотеке, осуществить поиск необходимого издания. Осуществить сортировку двоичного файла методом внешней сортировки.

Тема 9. Обработка исключений.

Компьютерная программа , примерные вопросы:

Обзор стандартных исключений и их обработка в программах. Пример типового задания. В задачах по обработке файлов обработать исключения конца файла и "файл не найден".

Итоговая форма контроля

зачет (в 5 семестре)

Примерные вопросы к итоговой форме контроля

Зачет происходит в форме выполнения студентами практического задания по пройденным темам. При выполнении задания будут оцениваться следующие навыки:

- знание терминологии и принципов разработки программ на С#;
- умение создавать собственные классы и наполнять их функциональными свойствами;
- умение работать с источниками данных (текстовыми и двоичными файлами).

Примеры типового задания.

1. Разработать класс "Множество целых чисел". Реализовать конструкторы, свойства и индексатор. Переопределить операции ввода-вывода и операции объединения, пересечения и сравнения двух множеств.
2. Разработать абстрактный класс "Функция", определив в нем переменные для хранения коэффициентов, методы ввода-вывода и вычисления значения функции в заданной точке. Создать классы-наследники "Линейная функция", "Квадратичная функция", "кубическая функция", дополнив их необходимыми данными и переопределив абстрактные методы.
3. Разработать класс "Упорядоченный список". Реализовать конструкторы, свойства и индексатор. Переопределить операции ввода-вывода и операции добавления, поиска и удаления элемента, а также слияние двух упорядоченных списков.
4. Разработать класс "Дерево". Реализовать конструкторы, свойства . Переопределить операции ввода-вывода и операции добавления, поиска и удаления элемента, а также слияние двух упорядоченных деревьев.
5. Разработать класс "Дерево". Реализовать конструкторы, свойства . Переопределить операции ввода-вывода и операции добавления, поиска и удаления элемента, а также методы, подсчитывающий количество уровней в дереве и количество элементов на заданном уровне.
6. Разработать класс "Дерево". Реализовать конструкторы, свойства . Переопределить операции ввода-вывода и операции добавления, поиска и удаления элемента, а также метод печати элементов дерева по уровням.
7. Разработать класс "Дерево". Реализовать конструкторы, свойства . Переопределить операции ввода-вывода и операции добавления, поиска и удаления элемента, а также метод, который строит копию исходного дерева.
8. Разработать класс "Дерево". Реализовать конструкторы, свойства . Переопределить операции ввода-вывода и операции добавления, поиска и удаления элемента, а также метод, печатающий все значения из листьев дерева.
9. Разработать класс "Двунаправленный список". Реализовать конструкторы, свойства и индексатор. Переопределить операции ввода-вывода и операции добавления, поиска и удаления элемента, а также метод, определяющий, входит ли список L1 в список L2.
10. Разработать класс "Двунаправленный список". Реализовать конструкторы, свойства и индексатор. Переопределить операции ввода-вывода и операции добавления, поиска и удаления элемента, а также метод, определяющий количество вхождений максимального элемента в заданный список.

11. Разработать класс "Двунаправленный список". Реализовать конструкторы, свойства и индексатор. Переопределить операции ввода-вывода и операции добавления, поиска и удаления элемента, а также метод, определяющий, является ли заданный список симметричным.
12. Разработать класс "Двунаправленный список". Реализовать конструкторы, свойства и индексатор. Переопределить операции ввода-вывода и операции добавления, поиска и удаления элемента, а также метод, определяющий является ли заданный список упорядоченным.
13. Разработать класс "Двунаправленный список". Реализовать конструкторы, свойства и индексатор. Переопределить операции ввода-вывода и операции добавления, поиска и удаления элемента, а также метод, переворачивающий заданный список.
14. Разработать класс "Двунаправленный список". Реализовать конструкторы, свойства и индексатор. Переопределить операции ввода-вывода и операции добавления, поиска и удаления элемента, а также метод, соединяющий два заданных списка.
15. Разработать класс "Многочлен". Реализовать конструкторы, свойства и индексатор. Переопределить операции ввода-вывода и операции сложения, вычитания, умножения на число, перемножения двух многочленов.
16. Разработать класс "Многочлен". Реализовать конструкторы, свойства и индексатор. Переопределить операции ввода-вывода и методы вычисления значения многочлена в заданной точке и вычисления нового многочлена, являющегося производной функцией по отношению к заданному.
17. Разработать класс "Двунаправленный список". Реализовать конструкторы, свойства и индексатор. Переопределить операции ввода-вывода и операции добавления, поиска и удаления элемента, а также метод, удаляющий элементы из списка, у которых равные соседи.
18. Разработать класс "Двунаправленный список". Реализовать конструкторы, свойства и индексатор. Переопределить операции ввода-вывода и операции добавления, поиска и удаления элемента, а также метод, печатающий элементы, находящиеся между наименьшим и наибольшим элементами этого списка.
19. Разработать класс "Двунаправленный список". Реализовать конструкторы, свойства и индексатор. Переопределить операции ввода-вывода и операции добавления, поиска и удаления элемента, а также метод, удаляющий элементы, находящиеся между наименьшим и наибольшим элементами этого списка.
20. Разработать класс "Многочлен". Реализовать конструкторы, свойства и индексатор. Переопределить операции ввода-вывода и методы вычисления значения многочлена в заданной точке и вычисления нового многочлена, являющегося первообразной функцией по отношению к заданному.
21. Разработать класс "Многочлен". Реализовать конструкторы, свойства и индексатор. Переопределить операции ввода-вывода и метод приведения подобных членов многочлена.

7.1. Основная литература:

1. Введение в программирование на языке Visual C#: Учебное пособие / С.Р.Гуриков. - М.:Форум: НИЦ ИНФРА-М, 2013. - 448 с. - (Высшее образование: Бакалавриат). ISBN 978-5-91134-738-3 Режим доступа: <http://znanium.com/bookread2.php?book=404441>
2. Информатика: Курс лекций / Е.Л. Федотова, А.А. Федотов. - М.: ИД ФОРУМ: ИНФРА-М, 2015. - 480 с. - (Высшее образование). ISBN 978-5-8199-0448-0 - Режим доступа: <http://znanium.com/catalog/product/500194>
3. Корнеев В.И. Программирование графики на C++. Теория и примеры : учеб. пособие / В.И. Корнеев, Л.Г.Гагарина, М.В. Корнеева. - М. : ИД 'ФОРУМ' : ИНФРА-М, 2017. - 517 с. Режим доступа: <http://znanium.com/bookread2.php?book=562914>

7.2. Дополнительная литература:

1. Гагарина Л.Г. Программирование на языке высокого уровня. Программирование на языке C++: учеб. пособие /Т.И. Немцова, С.Ю. Голова, А.И. Терентьев ; под ред. Л.Г. Гагариной. - М. : ИД 'ФОРУМ' : ИНФРА-М, 2018. - 512 с. - 480 с Режим доступа: <http://znanium.com/bookread2.php?book=918098>
2. Культин Н.Б. Microsoft Visual C# 2010 в задачах и примерах/ Никита Культин. - СПб: БХВ-Петербург, 2009. - 310 с. - ISBN 978-5-9775-0410-2. Режим доступа: <http://znanium.com/bookread2.php?book=350678>
3. Акулич, И.Л. Математическое программирование в примерах и задачах [Электронный ресурс] : учебное пособие / И.Л. Акулич. - Электрон. дан. - Санкт-Петербург : Лань, 2011. - 352 с. - Режим доступа: <https://e.lanbook.com/book/2027>
4. Абайдуллин Б. Р., Тагиров Р. Р. Лабораторные работы по курсу 'Классы и структуры данных': учеб.-метод. пособие / Б.Р. Абайдуллин, Р.Р. Тагиров. - Казань: Казан. ун-т, 2017. - 46 с. Режим доступа: https://repository.kpfu.ru/?p_id=157767

7.3. Интернет-ресурсы:

Интернет-портал образовательных ресурсов КФУ - <http://www.kfu-elearning.ru/>

Интернет-портал образовательных ресурсов по ИТ - <http://www.intuit.ru>

Интернет-портал по программным средствам Microsoft - <https://docs.microsoft.com/ru-ru/dotnet/csharp/language-reference/index>

Интернет-портал со статьями по алгоритмике и программированию - <http://algolist.manual.ru>

Электронная библиотека по техническим наукам - <http://techlibrary.ru>

8. Материально-техническое обеспечение дисциплины(модуля)

Освоение дисциплины "Язык C# и технологии .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.

Учебно-методическая литература для данной дисциплины имеется в наличии в электронно-библиотечной системе "ZNANIUM.COM", доступ к которой предоставлен студентам. ЭБС "ZNANIUM.COM" содержит произведения крупнейших российских учёных, руководителей государственных органов, преподавателей ведущих вузов страны, высококвалифицированных специалистов в различных сферах бизнеса. Фонд библиотеки сформирован с учетом всех изменений образовательных стандартов и включает учебники, учебные пособия, УМК, монографии, авторефераты, диссертации, энциклопедии, словари и справочники, законодательно-нормативные документы, специальные периодические издания и издания, выпускаемые издательствами вузов. В настоящее время ЭБС ZNANIUM.COM соответствует всем требованиям федеральных государственных образовательных стандартов высшего профессионального образования (ФГОС ВПО) нового поколения.

Учебно-методическая литература для данной дисциплины имеется в наличии в электронно-библиотечной системе Издательства "Лань", доступ к которой предоставлен студентам. ЭБС Издательства "Лань" включает в себя электронные версии книг издательства "Лань" и других ведущих издательств учебной литературы, а также электронные версии периодических изданий по естественным, техническим и гуманитарным наукам. ЭБС Издательства "Лань" обеспечивает доступ к научной, учебной литературе и научным периодическим изданиям по максимальному количеству профильных направлений с соблюдением всех авторских и смежных прав.

Лабораторные занятия проводятся в компьютерных классах лаборатории малой вычислительной техники института ВМ и ИТ, оснащенных мультимедийным оборудованием.

Программа составлена в соответствии с требованиями ФГОС ВПО и учебным планом по направлению 01.03.02 "Прикладная математика и информатика" и профилю подготовки Системное программирование .

Автор(ы):

Мухтарова Т.М. _____

"__" _____ 201__ г.

Рецензент(ы):

"__" _____ 201__ г.