МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное автономное образовательное учреждение
высшего образования
"Казанский (Приволжский) федеральный университет"
Институт фундаментальной медицины и биологии

УТВЕРЖДАЮ
Проректор по образовательной деятельности КФУ
Проф. Д.А. Таюрский
_____ «___»_____20__г.

подписано электронно-цифровой подписью

Программа дисциплины

Основы программирования Б1.Б.17

Специальность: 30.05.03 - Медицинская кибернетика
Специализация: не предусмотрено
Квалификация выпускника: врач-кибернетик
Форма обучения: очное
Язык обучения: русский

Автор(ы):
Абрамский М.М. , Насрутдинов М.Ф.

Рецензент(ы):
Таланов Максим Олегович

СОГЛАСОВАНО:

Заведующий(ая) кафедрой: Хасьянов А. Ф.
Протокол заседания кафедры No ___ от "____" _____ 201__г
Учебно-методическая комиссия Института фундаментальной медицины и биологии:
Протокол заседания УМК No ____ от "____" _____ 201__г

Регистрационный No 8494338319

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 Абрамский М.М. Кафедра программной инженерии Высшая школа информационных технологий и интеллектуальных систем , Michael.Abramsky@kpfu.ru ; заместитель директора по образовательной деятельности Насрутдинов М.Ф. Высшая школа информационных технологий и интеллектуальных систем КФУ , Marat.Nasrutdinov@kpfu.ru

1. Цели освоения дисциплины

Дисциплина 'Основы программирования' посвящена изучению основных концепций программирования.

Целью освоения дисциплины является понимание студентами процесса создания программного обеспечения и умение ставить задачи разработчикам

в дальнейшей профессиональной деятельности.

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

Данная учебная дисциплина включена в раздел "Б1.Б.17 Дисциплины (модули)" основной образовательной программы 30.05.03 Медицинская кибернетика и относится к базовой (общепрофессиональной) части. Осваивается на 4 курсе, 7 семестр.

Дисциплина относится к базовой части.

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

В результате освоения дисциплины формируются следующие компетенции:

Шифр компетенции	Расшифровка приобретаемой компетенции
ОК-1 (общекультурные компетенции)	способностью к абстрактному мышлению, анализу, синтезу
ОПК-1 (профессиональные компетенции)	готовностью решать стандартные задачи профессиональной деятельности с использованием информационных, библиографических ресурсов,медико-биологической терминологии,информационно-коммуникационных технологий и учетом основных требований информационной безопасности
ОПК-11 (профессиональные компетенции)	готовностью решать стандартные задачи профессиональной деятельности с использованием информационных, библиографических ресурсов,медико-биологической терминологии,информационно-коммуникационных технологий и учетом основных требований информационной безопасности
ОПК-5 (профессиональные компетенции)	способностью и готовностью анализировать результаты собственной деятельности для предотвращения профессиональных ошибок
ОПК-7 (профессиональные компетенции)	готовностью к использованию основныхфизико-химических, математических и иныхестественнонаучных понятий и методов при решениипрофессиональных задач
ПК-7 (профессиональные компетенции)	готовностью к обучению на индивидуальном и популяционном уровнях основным гигиеническим мероприятиям оздоровительного характера, навыкам самоконтроля основных физиологических показателей,способствующим сохранению и укреплению здоровья,профилактике заболеваний

Шифр компетенции	Расшифровка приобретаемой компетенции
ПК-8 (профессиональные компетенции)	готовностью к просветительской деятельности поустранению факторов риска и формированию навыков здорового образа жизни

В результате освоения дисциплины студент:

1. должен знать:

-основные парадигмы программирования (процедурная, объектно-ориентированная, функциональная),

- основные инструменты, реализующие в себе концепцию этих парадигм
- основные структуры данных и структуры управления,
- базовые алгоритмы обработки данных
- базовые принципы проектирования информационных систем

2. должен уметь:

- алгоритмизировать задачи общего и медицинского характера;
- программировать на процедурном языке и на объектно-ориентированном языке программирования на примере задач общего и медицинского характера;
- проектировать информационные системы на основе объектно-ориентированного подхода

3. должен владеть:

- методами алгоритмизации задач общего и медицинского характера;
- технологиями построения информационных систем на основе объектно-ориентированного подхода;
- способами измерения эффективности алгоритмов и умением писать такие алгоритмы

4. должен демонстрировать способность и готовность:

Применять полученные знания, умения и навыки в профессиональной деятельности

4. Структура и содержание дисциплины/ модуля

Общая трудоемкость дисциплины составляет 5 зачетных(ые) единиц(ы) 180 часа(ов).

Форма промежуточного контроля дисциплины: экзамен в 7 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. Введение в						

алгоритмизацию и программирование. Управляющие операторы языка

7		4	0	12	Лабораторные

работы

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	
2.	Тема 2. Базовые алгоритмы обработки данных.	7		4	0	12	Лабораторные работы
3.	Тема 3. Массивы, действия с ними. Символы и строки	7		4	0	14	Лабораторные работы
4.	Тема 4. Процедурное программирование. Объектно-ориентированный подход (ООП).	7		6	0	19	Лабораторные работы
5.	Тема 5. Введение в структуры данных и алгоритмы	7		6	0	19	Лабораторные работы
.	Тема . Итоговая форма контроля	7		0	0	0	Экзамен
	Итого			24	0	76	

4.2 Содержание дисциплины

Тема 1. Введение в алгоритмизацию и программирование. Управляющие операторы языка

лекционное занятие (4 часа(ов)):

Лекция: Понятие информации. Форма представления информации. Структура компьютера по фон Нейману. Базовые технологии преобразования информации. Парадигмы программирования. Компилируемые и интерпретируемые языки (примеры, различия). Основные положения информатики, медицинской информатики, кибернетики. Математический аппарат теории информатики. Структура компьютера по фон Нейману. Понятие алгоритма. Языки низкого и высокого уровня. Этапы компилирования и исполнения программ. Структура программы. Типы данных и их свойства. Обзор условных и циклических операторов в языке. Обзор задач, решаемых через управляющие конструкции. Средства ввода/вывода.

лабораторная работа (12 часа(ов)):

Лабораторная работа: Прикладное программное обеспечение. Командная строка. Операции в командной строке. Репозиторий git. Принцип работы репозитория. Структурные и неструктурные блок-схемы. Структурирование блок-схем. Структура класса, метод main. Компиляция и исполнение программ. Примитивные типы данных (int, double, float, boolean, char), различия между примитивными и ссылочными типами данных. Приведение типов. Арифметические и логические операции. Задачи на ввод-вывод данных, условный оператор if, оператор switch, сокращенный if, циклы с пред и пост условием while и do while, циклы for и for each, операторы break и continue.

Тема 2. Базовые алгоритмы обработки данных.

лекционное занятие (4 часа(ов)):

Лекция Подходы к построению алгоритмов. Проектирование алгоритмов TOP-DOWN и BOTTOM-UP. Спообоы задания алгоритмов. Автоматы. Машины Тьюринга.

ЭЛЕКТРОННЫЙ УНИВЕРСИТЕТ
ИНФОРМАЦИОННО-АНАЛИТИЧЕСКАЯ СИСТЕМА КФУ

лабораторная работа (12 часа(ов)):

Лабораторная работа: Принадлежность точки фигуре, заданной системой неравенств, схема Горнера, вычисление конечных сумм, вычисление бесконечных сумм с некоторой точностью, вычисление последовательностей, заданных рекуррентной формулой, задачи на проверку условий 'для любого' и 'существует'.

Тема 3. Массивы, действия с ними. Символы и строки

лекционное занятие (4 часа(ов)):

Лекция: Необходимость массивов. Хранение в памяти. Одномерные массивы, алгоритмы обработки массивов, многомерные массивы, ступенчатые массивы. Строки. Тип char, номер символа в ASCII. Юникод. Класс String. Объявление, использование (ввод-вывод, конкатенация, методы charAt и length). Ввод-вывод данных. История развития средств ввода/вывода. Регулярные выражения. Жадные, ленивые регулярные выражения.

лабораторная работа (14 часа(ов)):

Лабораторная работа: Алгоритмы поиска максимума/минимума, алгоритмы сортировки (выбор, обмен, слиянием), многомерные массивы - умножение, сложение матриц, транспонирование матриц. Реализация строковых алгоритмов (поиск подстроки, сравнение). Синтаксис регулярных выражений. Классы Pattern, Matcher, методы compile, match, find, group.

Тема 4. Процедурное программирование. Объектно-ориентированный подход (ООП).

лекционное занятие (6 часа(ов)):

Лекция: Функции и их парадигма. Отличие метода от функции. Принципы ООП - абстракция, 3 кита ООП (инкапсуляция, наследование, полиморфизм), классы и объекты. Методы. Объявление и вызов. Параметры. Перегрузка. Классы. Объявление. Члены класса (атрибуты, методы), Наследование и полиморфизм. Восходящее преобразование и позднее связывание. Объекты. Создание объекта, оператор new. Пакеты в Java. Необходимость. Принципы построения. Интефейсы и абстрактные классы. Абстрактные методы и абстрактные классы. Интерфейсы, их отличие от абстрактных классов. Интерфейсы и множественное наследование. Интерфейсы как средство адаптации. Клонирование и его реализация в Java.

лабораторная работа (19 часа(ов)):

Лабораторная: Оптимизация задач с помощью методов. Проектирование объектов из различных предметных областей. Реализация объектов из различных предметных областей. Оператор this, конструктор класса, модификаторы (public, private, protected, по умолчанию, static, abstract). Создание пакета. Иерархия. Поля в интерфейсах и их инициализация. Вложенные интерфейсы. Интерфейсы и фабрики. Интерфейсы-маркеры. Решение задач на проектирование интерфейсов. Основные сведения о перечисляемых типах (enums).

Тема 5. Введение в структуры данных и алгоритмы

лекционное занятие (6 часа(ов)):

Лекция: Параметризованные типы (Generics). Линейные и древовидные структуры данных. Типы данных линейной структуры с последовательным доступом к данным: стек, очереди, дек. Связанные линейные списки. Алгоритмы обработки структур данных. Сортировка. Алгоритмы сортировки: выбором, обменом, вставками, быстрая, карманная, поразрядовая, пирамидальная. Слияние сортированных последовательностей. Поиск: Последовательный поиск, Бинарный поиск. Деревья. Способы отображения деревьев. Двоичные (бинарные) деревья. Деревья двоичного поиска. Операции с двоичными деревьями. Сбалансированные деревья. Основные определения. Графы. Основные понятия и определения. Способы задания графов. Алгоритмы на графах. Поиск в глубину. Поиск в ширину.

лабораторная работа (19 часа(ов)):

Лабораторное занятие: Решение задач на коллекции объектов. Линейные структуры данных, древовидные структуры данных. Реализация абстрактных типов данных и структур данных. Реализация абстрактных типах в структурах данных. Коллекции объектов. Параметризованные и типизированные контейнеры. Основные понятия. Списки, итераторы, множества, отображения. Collection и Iterator. Foreach и итераторы. Оптимизационные алгоритмы. Кратчайшие пути. Алгоритм перебора с возвратами. Алгоритмы сортировки коллекций данных.

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. Введение в алгоритмизацию и программирование. Управляющие операторы языка	7		Подготовка к лабораторной работе. Написание программ.	7	Лабораторные работы
2.	Тема 2. Базовые алгоритмы обработки данных.	7		Подготовка к лабораторной работе. Написание программ.	7	Лабораторные работы
3.	Тема 3. Массивы, действия с ними. Символы и строки	7		Подготовка к лабораторной работе. Написание программ.	8	Лабораторные работы
4.	Тема 4. Процедурное программирование. Объектно-ориентированный подход (ООП).	7		Подготовка к лабораторной работе. Написание программ.	24	Лабораторные работы
5.	Тема 5. Введение в структуры данных и алгоритмы	7		Подготовка к лабораторной работе. Написание программ.	25	Лабораторные работы
	Итого				71	

5. Образовательные технологии, включая интерактивные формы обучения

Работа в командах. Обсуждение алгоритмов на семинарских занятиях.

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

Тема 1. Введение в алгоритмизацию и программирование. Управляющие операторы языка

Лабораторные работы , примерные вопросы:

1. Реализовать с помощью длинной арифметики деление одного числа на другое. На экран должны выводиться целая часть и остаток от деления.2. Реализовать с помощью длинной арифметики умножение одного числа на другое.3. Реализовать с помощью длинной арифметики умножение числа на цифру.4. Подсчитать в двумерном массиве максимум по суммам элементов на главных диагоналях.5. Подсчитать в двумерном массиве максимум из минимумов по строкам

Тема 2. Базовые алгоритмы обработки данных.

Лабораторные работы , примерные вопросы:

6. Проверить, что в двумерном массиве каждая сумма элементов на побочных диагоналях является четной7. Проверить, что в трехмерном массиве в каждом его двумерном массиве существует такая строка, что в ней все элементы делятся на три.8. Написать программу, которая сравнивает две строки и выводит, какая находится раньше лексикографически.Не использовать compareTo. 9. Вводится строка, представляющая собой слова на английском языке, записанные через пробел. Подсчитать,сколько слов начинается на заглавную букву.10. Подсчитать кусочно-заданную функцию с помощью условного оператора.11. Подсчитать сумму числового ряда.12. Вводится квадратная матрица размера n. Привести ее к треугольному виду и вывести на экран. Не забудьте,что надо обрабатывать случай поиска ненулевого элемента в столбце (поиск, существование, квантор).13. Реализовать с помощью двумерных массивов умножение матриц.14. Вводится n чисел. Проверить, что существует ровно три числа, в котором цифры идут по убыванию.15. Вводится n чисел. Проверить, что существует такое число, в котором цифры идут по возрастанию.16. Вводится число n. Построить число m, которое будет содержать только нечетные цифры числа n (в сохраненном порядке). Например, для n = 123456, m = 135.17. Дано натуральное число N. Приписать по единице в начало и в конец записи числа N и вывести на экран(аналогично, нужно число!

Тема 3. Массивы, действия с ними. Символы и строки

Лабораторные работы , примерные вопросы:

1. Рекурсивно решить задачу подсчета факториала.2. Рекурсивно решить задачу поиска максимума в массиве.3. Спроектировать класс "Вектор на плоскости"4. Спроектировать класс "Комплексное число"5. Спроектировать класс "Рациональная дробь"6. Спроектировать класс "Комплексное число из рациональных дробей".7. Спроектировать класс "Комплекснозначный вектор"8. Спроектировать класс "Система линейных уравнений"9. Спроектировать класс "Трехмерная матрица",10. Спроектировать класс "n-мерный вектор".

Тема 4. Процедурное программирование. Объектно-ориентированный подход (ООП).

Лабораторные работы , примерные вопросы:

. Рекурсивно решить задачу сортировки массива.2. Рекурсивно решить задачу поиска максимума минимумов строк двумерного массива.3. Спроектировать класс "n-мерный комплекснозначный вектор".4. Спроектировать класс "Вектор из рациональных дробей"5. Спроектировать класс "Вектор из комплексных чисел"6. Спроектировать класс "Вектор из комплексных чисел из рациональных дробей"7. Спроектировать класс "Матрица из рациональных дробей"8. Спроектировать класс "Матрица из комплексных чисел"9. Спроектировать класс "Матрица из комплексных чисел из рациональных дробей"10. Спроектировать класс "n-мерный вектор из рациональных дробей".

Тема 5. Введение в структуры данных и алгоритмы

Лабораторные работы , примерные вопросы:

1. Реализовать интерфейс MyStack с методами push, pop, isEmpty. Далее реализовать от него классMyLinkedStack, в котором для хранения элементов используется линейный односвязный список, элементыкоторого являются объектами класса Elem2. Для введенной строки проверить в ней правильность расстановки скобок. При этом обязательно вывести типаошибки, если она произошла: "не все открывающие закрыты", "встретилась лишняя закрывающая", "скобки несоответствуют друг другу" + выводя позицию, на которой произошла ошибка.3. С помощью MyLinkedStack для введенной строки, представляющей собой постфиксную запись, написатьалгоритм, вычисляющий представленное в ней арифметическое выражения. Считать вводимые данные всегдаправильными, лексемы разделены пробелом запятой.4. Напишите алгоритм перевода обычной записи арифметического выражения в его префиксную/посфикснуюзаписи. В исходном арифметическом выражении могут быть целые отрицательные и положительные числа,названия переменных, знаки операций (+, -, *, /) и скобки.5. Реализуйте MyLinkedQueue от интерфейса java.util.Queue, использовав в качестве реализации линейныйодносвязный список. Реализовать методы add(e), offer(e), remove(), poll(), element(), peek(), остальные оставитьзаглушками6. Реализуйте MyLinkedQueue от интерфейса java.util.Queue, использовав в качестве реализации 2 стекаMyLinkedStack (которые вы реализовали в задании 016). Реализовать уметоды add(e), offer(e), remove(), poll(),element(), peek(), остальные оставить заглушками.7. Построить первые N натуральных чисел, которые состоят только из произведения двоек, троек или пятерок.8. Реализовать MyLinkedDeque от интерфейса java.util.Deque, использовав в качестве реализации односвязныйлинейный список.Реализуйте только те методы, которые описаны в таблицах до Method Summary:https://docs.oracle.com/javase/7/docs/api/java/util/Deque.html9. Реализовать Set в классе MyDummyTreeSet, храня внутри реализацию с помощью с помощью дерева поиска(балансировать не нужно). Без итератора.10. Найти произведение всех его вершин с помощью поиска в глубину (рекурсивно)11. Найти максимум среди всех его вершин с помощью поиска в глубину (стековый КЛП).12. Реализовать в классе Graph статические методы, позволяющие сконвертировать одно из четырех указанных представлений графа в другое (т.е. всего 12 методов)13. Найти с помощью алгоритма перебора с возвратами все строки длины n из строчных английских букв, чтобы в строке было не более трех гласных.14. Найти с помощью алгоритма перебора с возвратами все целые положительные числа с m цифрами, у которых сумма цифр меньше числа k.15. Найти раскраску графа (вводится список ребер, создается объект класса Graph) с помощью алгоритма перебора с возвратами

Итоговая форма контроля

экзамен (в 7 семестре)

Примерные вопросы к итоговой форме контроля

Вопросы к экзамену:

1. Структура компьютера по фон Нейману.

2. Понятие алгоритма.

3. Этапы компилирования и исполнения программ. Компилирование и выполнение с помощью командной строки.

4. Примитивные типы данных, различия между примитивными и ссылочными типами данных. Приведение типов.

5. Арифметические и логические операции. Преобразование типов в при арифметических операциях. Сужениепри преобразовании типов

6. Условные и циклические операторы языка. Условные операторы if, switch case, (P ? v1 : v2).

7. Циклические операторы while, do while, циклы for и for each, операторы break и continue.

8. Массивы, действия с ними. Одномерные массивы объявление ссылки и выделение памяти. Обращение кэлементу массива. Как пройти по всем элементам массива.Многомерные массивы - объявление, способ пройти по всем элементам. Ступенчатые массивы.

9. Символы. Тип char, номер символа в ASCII. Способ узнать номер.

10. Класс String. Объявление, использование (ввод-вывод, конкатенация, характерные методы). Особенностихранения строковых констант в Java.

11. Регулярные выражения. Синтаксис регулярных выражений. Классы Pattern, Matcher, методы compile, match,find, group. Жадные, ленивые регулярные выражения.

12. Методы. Объявление и вызов. Правила вызова метода. Возвращаемые значения, return. Параметры, передачапо ссылке, по значению. Перегрузка.

13. Классы и объекты. Классы. Структура класса, члены класса. Объявление объекта класса. Объявление ссылкии выделение памяти. Конструктор класса, оператор this.

14. Инкапсуляция. Модификаторы доступа - private, protected, public, по умолчанию. Модификатор static.

15. Наследование и полиморфизм. Правила наследования методов и атрибутов. Оператор super.Переопределение.

16. Интерфейсы. Объявление, структура. Что могут содержать интерфейсы?

17. Наследование интерфейсов. Правила реализации интерфейса классом. Восходящее преобразование в случаеинтерфейса.

18. Исключения. Примеры известных. Блок try - catch, правила работы. Использование finally. Корректнаяпоследовательность обработки исключений

19. Абстрактные типы данных и структуры. Линейные структуры.

20. Абстрактные типы данных и структуры. Древовидные структуры.

7.1. Основная литература:

1. Хабибуллин И. Ш. Самоучитель Java / Ильдар Хабибуллин. - 3-е изд., перераб. и доп. - СПб.: БХВ-Петербург,2008. - 758 с.: ил. - Режим доступа: http://znanium.com/bookread2.php?book=350488

2. Язык программирования Java и среда NetBeans: Курс лекций / Монахов В.В. - СПб: БХВ-Петербург, 2011. - 703 с. - Режим доступа: http://znanium.com/catalog/product/355260

3. Машнин Т. С. Современные Java-технологии на практике. - СПб.: БХВ-Петербург, 2010. - 560 с. - Режим доступа: http://znanium.com/bookread2.php?book=351236

7.2. Дополнительная литература:

1. Информатика: Курс лекций. Учебное пособие / Е.Л. Федотова, А.А. Федотов. - М.: ИД ФОРУМ: ИНФРА-М,2011. - 480 с.: ил. - (Высшее образование). - Режим доступа: http://znanium.com/bookread.php?book=204273

Задачи по программированию: учебное пособие / С.М. Окулов, Т.В. Ашихмина, Н.А. Бушмелева, М.А. Корчёмкин. - 3-е изд. - Москва: Лаборатория знаний, 2017. - 826 с. - Текст: электронный // Электронно-библиотечная система 'Лань': [сайт]. - URL: https://e.lanbook.com/book/94162

7.3. Интернет-ресурсы:

Online-площадка для изучения языков программирования - https://www.codecademy.com/

Загрузка Java - https://java.com/ru/download/

Образовательный справочник технических вопросов по программированию и информатике - http://www.quizful.net/test

Руководства по изучению Java и связанных технологий и фреймворков - http://tutorials.jenkov.com/

ЭБС Знаниум - http://znanium.com

8. Материально-техническое обеспечение дисциплины(модуля)

Освоение дисциплины "Основы программирования" 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 оборудованием 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.

-

Программа составлена в соответствии с требованиями ФГОС ВПО и учебным планом по специальности: 30.05.03 "Медицинская кибернетика" и специализации не предусмотрено .

Автор(ы):
Насрутдинов М.Ф. _____
Абрамский М.М. _____
"__" _____ 201 __ г.

Рецензент(ы):
Таланов Максим Олегович _____
"__" _____ 201 __ г.