

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ
 ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ
 ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
 «ВОРОНЕЖСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»
 (ФГБОУ ВО «ВГТУ», ВГТУ)

«УТВЕРЖДАЮ»
 Председатель Ученого совета
 Факультета информационных
 технологий и компьютерной
 безопасности
 Пасмурнов С.М. 
 (подпись)
 2017 г.

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

Объектно-ориентированное программирование

(наименование дисциплины по УП)

Закреплена за кафедрой: Компьютерных интеллектуальных технологий проектирования

Направление подготовки (специальности):

09.03.01 Информатика и вычислительная техника

(код, наименование)

Профили: Вычислительные машины, комплексы, системы и сети, Системы автоматизированного проектирования, Системы автоматизированного проектирования в машиностроении

(название профиля по УП)

Часов по УП: 180; Часов по РПД: 180;

Часов по УП (без учета часов на экзамены): 180; Часов по РПД: 180;

Часов на самостоятельную работу по УП: 126 (70 %);

Часов на самостоятельную работу по РПД: 126 (70 %);

Общая трудоемкость в ЗЕТ: 5;

Виды контроля в семестрах: Экзамены – 0; Зачеты – 0; Зачеты с оценкой – 3; Курсовые проекты -0; Курсовые работы - 0.

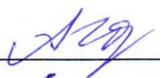
Форма обучения: очная;

Срок обучения: нормативный.

Распределение часов дисциплины по семестрам

Вид занятий	№ семестров, число учебных недель в семестрах																	
	1/18		2/18		3/18		4/18		5/18		6/18		7/18		8/12		Итого	
	УП	РПД	УП	РПД	УП	РПД	УП	РПД	УП	РПД	УП	РПД	УП	РПД	УП	РПД	УП	РПД
Лекции							18	18									18	18
Лабораторные							36	36									36	36
Практические																		
Ауд. занятия							54	54									54	54
Сам. работа							126	126									126	126
Итого							180	180									180	180

Сведения о ФГОС, в соответствии с которым разработана рабочая программа дисциплины (модуля) – 09.03.01 «Информатика и вычислительная техника», утвержден приказом Министерства образования и науки Российской Федерации от 12 января 2016г. № 5.

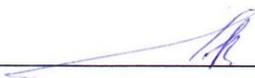
Программу составил:  к.т.н., Юров А.Н.
(подпись, ученая степень, ФИО)

Рецензент (ы):  зав. каф. ИИТ ВГУ, к.ф.-н.и, доц. Тюкачев Н.И.

Рабочая программа дисциплины составлена на основании учебного плана подготовки специалистов по направлению 09.03.01 Информатика и вычислительная техника, специализация все профили

Рабочая программа обсуждена на заседании кафедры компьютерных интеллектуальных технологий проектирования

Зав. кафедрой КИТП  М.И. Чижов

Согласовано:
Зав. кафедрой САПРИС  Я.Е. Львович

Согласовано:
Зав. кафедрой АВС  С.Л. Подвальный

1. ЦЕЛИ ОСВОЕНИЯ ДИСЦИПЛИНЫ

1.1	<p>Цель изучения дисциплины является приобретение базовых знаний и навыков программирования, проектирование и разработка приложений с применением объектно-ориентированного подхода, приобретение навыков, позволяющих будущим специалистам вести успешную разработку специализированного программного обеспечения в тех областях и сферах деятельности, в которых они будут трудиться.</p> <p>Изучение дисциплины должно способствовать формированию у студентов основ научного мышления, в том числе: владение основными методами, способами и инструментами создания программного обеспечения, использования для решения практических задач</p>
1.2	<p>Для достижения цели ставятся задачи:</p>
1.2.1	ознакомление с тенденцией развития программного обеспечения;
1.2.2	обучение разработке алгоритмов на основе объектно-ориентированного подхода;
1.2.3	освоение языка программирования C++, а также стандартов кодирования, спецификаций и последующих решений на их основе;
1.2.4	работа с современными программными инструментальными средствами разработки
1.2.5	изучение особенностей разработки программного обеспечения под имеющиеся операционные системы, в том числе и мобильные платформы.

2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ОПОП ВО

Цикл (раздел) ООП: Б1	код дисциплины в УП: Б1.Б.8.2
<p>2.1 Требования к предварительной подготовке обучающегося</p>	
<p>Для успешного освоения дисциплины студент должен иметь базовую подготовку по следующим дисциплинам:</p>	
	Математика
	Информатика
	Технология программирования
<p>2.2 Дисциплины и практики, для которых освоение данной дисциплины (модуля) необходимо как предшествующее</p>	
	Методы оптимизации

3. КОМПЕТЕНЦИИ ОБУЧАЮЩЕГОСЯ, ФОРМИРУЕМЫЕ В РЕЗУЛЬТАТЕ ОСВОЕНИЯ ДИСЦИПЛИНЫ

ОПК-2	Способностью осваивать методики использования программных средств для решения
-------	---

	<p>практических задач</p> <p>Знать: основы по алгоритмическим конструкциям, команды языка программирования и средства среды разработки, позволяющие работать с информацией</p> <p>Уметь: применять полученные знания при разработке приложений</p> <p>Владеть: инструментальными средствами разработки по обработке информации и применять их на практике</p>
ОПК-1	<p>Способностью устанавливать программное и аппаратное обеспечение для информационных и автоматизированных систем</p> <p>Знать: технологии и методы разработки ПО</p> <p>Уметь: разрабатывать приложения оптимальным способом</p> <p>Владеть: средствами отладки и тестирования приложений</p>

В результате освоения дисциплины обучающийся должен

3.1	Знать:
3.1.1	основные принципы конструирования алгоритмов (ОПК-2)
3.1.2	основные объекты представления данных (классы) и методы их обработки, а также способы реализации (ОПК-2)
3.1.3	методы и технологии программирования (ОПК-2)
3.1.4	принципы и определения объектно-ориентированного подхода (ОПК-2)
3.1.5	порядок настройки и работы в интегрированных средах разработки программного обеспечения (ОПК-1)
3.2	Уметь:
3.2.1	разрабатывать алгоритмы (ОПК-2)
3.2.2	реализовывать алгоритмы на языке высокого уровня (ОПК-2)
3.2.3	описывать основные структуры данных на основе объектно-ориентированного подхода (ОПК-2)
3.3	Владеть:
3.3.1	методами описания основных классов и алгоритмов на их основе (ОПК-2)
3.3.2	навыками работы в различных средах программирования (ОПК-1)

4. СТРУКТУРА И СОДЕРЖАНИЕ ДИСЦИПЛИНЫ

№ П./п	Наименование раздела дисциплины	Семестр	Неделя семестра	Вид учебной нагрузки и их трудоемкость в часах				
				Лекции	Практические занятия	Лабораторные работы	СРС	Всего часов
1	Интегрированные среды и средства разработки на языке программирования высокого уровня	3	1-2	2		4	20	26
2	Базовые конструкции и ключевые слова языка программирования высокого уровня	3	3-4	2		4	18	24

3	Объектно-ориентированный подход к разработке программных средств	3	5-8	2		8	18	28
4	Разработка приложений с графическими интерфейсными возможностями;	3	9-14	6		8	18	32
5	Создание подсистем на основе подключаемых библиотек;	3	15	2		4	18	24
6	Построение кроссплатформенных приложений;	3	16-17	2		4	18	24
7	Создание инсталляционных пакетов программ.	3	18	2		4	16	22
Итого				18		36	126	180

4.1 Лекции

Неделя семестра	Тема и содержание лекции	Объем часов	В том числе, в интерактивной форме (ИФ)
3 семестр		18	
Интегрированные среды и средства разработки на языке программирования высокого уровня		2	
1	Введение Анализ языков программирования и обзор сред разработки программного обеспечения. Возможности и производительность современных компиляторов, сборщиков проектов.	1	
2	Среды разработки Интерфейс и конфигурация интегрированной среды разработки (IDE) под решение вычислительных задач. Разработка консольных проектов и приложений с графическим интерфейсом	1	
Базовые конструкции и ключевые слова языка программирования высокого уровня		2	
3	Синтаксис и операции Типы данных языка и расширения типов, предоставляемые средой. Арифметические операции. Правила именования методов, переменных и констант	1	
4	Основные алгоритмические конструкции Использование структур, условных конструкций, циклов, функций, а также директив предпроцессора	1	
Объектно-ориентированный подход к разработке программных средств		2	
5	Введение в классы и объекты Описание и свойства объектов. Конструкторы, деструкторы классов, объявления статических объектов, переменных и методов в классе	0.5	
6-7	Наследование в ООП Наследование, виды наследования. Абстрактные классы. Множественное наследование	1	
8	Управление данными, директивы предпроцессора Структуры, исключения, набор директив для сборки проектов	0.5	
Разработка приложений с графическими интерфейсными		6	

ВОЗМОЖНОСТЯМИ			
9	Введение в разработку пользовательского интерфейса Понятие интерфейса пользователя, метафоры и поиск подобных решений, интерактивные средства разработки	1	
10	Средства разработки интерфейса IDE сред Основы разработки программных интерфейсов, средства разработки, иерархия и структура интерфейсных классов	1	
11	Обработка событий в приложениях Средства взаимодействия между приложением и действиями, исходящими от пользователя	1	
12	Виды приложений и особенности их создания Однооконные и многооконные приложения. Обработка данных, запись результатов (SDI и MDI)	1	
13	Организация вывода графической информации Работа с графикой на плоскости, технологии реализации вывода в 3D, подключение к проектам библиотек, поддерживающих OpenGL	1	
14	Представление, ввод и вывод данных на носители информации Работа с файлами, поддержка баз данных, создание архивов и извлечение информации из них, регулярные выражения	1	
Создание подсистем на основе подключаемых библиотек		2	
15	Динамически-подключаемые источники данных Работа с системными библиотеками операционных систем	2	
Разработка кроссплатформенных приложений		2	
16	Введение и особенности создания межплатформенных проектов Построение кроссплатформенных приложений средствами IDE	1	
17	Технологии и инструменты сред разработки по созданию кроссплатформенных приложений Использование базовых классов среды. Перенос данных.	1	
Создание инсталляционных пакетов программ		2	
18	Создание установочных пакетов Средства и решения для автоматического развёртывания проектов на ЭВМ пользователя	2	
Итого часов		18	

4.2 Лабораторные работы

Неделя семестра	Наименование лабораторной работы	Объем часов	В том числе в интерактивной форме (ИФ)	Виды контроля
3 семестр		36		
Интегрированные среды и средства разработки на языке программирования высокого уровня		4		отчет и защита

1	Вводное занятие. Инструктаж по технике безопасности. Работа с ПК. Студенты выполняют работу в соответствии с индивидуальным графиком: -настройка профиля для последующей работы в IDE среде; -подготовка проектов с помощью готовых шаблонов	4		
Базовые конструкции и ключевые слова языка программирования высокого уровня		4		
3	Студенты выполняют одну из работ в соответствии с индивидуальным графиком: -изучение стандартных команд и инструкций алгоритмического языка программирования; -использование средств ввода-вывода на экран ЭВМ.	4		отчет и защита
Объектно-ориентированный подход к разработке программных средств		8		
5	Студенты выполняют одну из работ в соответствии с индивидуальным графиком: -создание расчетного приложения с описанием объектов в классе и инициализацией параметров; -создание проектов с динамически-изменяемыми структурами данных с освобождением памяти ЭВМ в процессе работы приложения; -использование статических переменных для обмена информацией между классами в разрабатываемом проекте	2		отчет и защита
6	Студенты выполняют одну из работ в соответствии с индивидуальным графиком: -создание приложения с применением множественного наследования; -работа с виртуальными функциями в проекте	2		отчет и защита
7	Студенты выполняют одну из работ в соответствии с индивидуальным графиком: -работа со структурами в классах при разработке приложений; -использование исключений в проблемных фрагментах кода приложения	4		отчет и защита
Разработка приложений с графическими интерфейсными возможностями		8		
9	Студенты выполняют одну из работ в соответствии с индивидуальным графиком: -разработка графического интерфейса расчетного приложения средствами программного функционала API; -разработка графического интерфейса расчетного приложения средствами дизайнера среды IDE	2		отчет и защита
10	Студенты выполняют одну из работ в соответствии с индивидуальным графиком: -создание многооконного приложения; -создание приложения на основе шаблона SDI	2		отчет и защита

11	Студенты выполняют одну из работ в соответствии с индивидуальным графиком: -обработка событий в приложении; -обработка исключений в приложении	2		отчет и защита
12-13	Студенты выполняют одну из работ в соответствии с индивидуальным графиком: -создание инструментальных панелей графического приложения; -использование ресурсов, работа с диалогами, организация всплывающих подсказок, интерактивных справочных данных и т.д.	2		отчет и защита
Создание подсистем на основе подключаемых библиотек		4		
15	Студенты выполняют одну из работ в соответствии с индивидуальным графиком: -использование динамически-подключаемых библиотек; -использование статических ресурсов, а также иных средств, подготовленных сторонними разработчиками для заданной интегрированной среды разработки ПО.	4		отчет и защита
Разработка кроссплатформенных приложений		4		
16	Студенты выполняют одну из работ в соответствии с индивидуальным графиком: -разработка приложений для операционных систем семейства Windows; -работа с директивами предпроцессора, конфигурирование проекта под запуск приложения в заданной ОС.	2		отчет и защита
17	Студенты выполняют одну из работ в соответствии с индивидуальным графиком: -разработка приложений, работающих в операционных системах под мобильные платформы; -работа с директивами предпроцессора, конфигурирование проекта под запуск приложения в заданной ОС.	2		отчет и защита
Создание инсталляционных пакетов программ		4		
18	Студенты выполняют одну из работ в соответствии с индивидуальным графиком: -создание установочных решений, позволяющих подготовить автоматизацию развертывания приложения средствами интегрированной среды разработки; -создание установочных решений, позволяющих подготовить автоматизацию развертывания приложения специализированными средствами с использованием скриптовых языков.	4		отчет и защита
Итого часов		36		

4.3 Самостоятельная работа студента (СРС)

Неделя семестра	Содержание СРС	Виды контроля	Объем часов
3 семестр		Зачет, КР	126

2	Подготовка к выполнению лаб. работы	допуск к выполнению	3,5
3	Подготовка к выполнению лаб. работы	допуск к выполнению	3,5
	Работа с конспектом лекций, с учебником		3,5
4	Работа с конспектом лекций, с учебником		3,5
	Подготовка к выполнению лаб. работы	допуск к выполнению	3,5
5	Подготовка к выполнению лаб. работы	допуск к выполнению	3,5
	Самостоятельное изучение материала		
	Работа с конспектом лекций, с учебником		3,5
6	Самостоятельное изучение материала	конспект	3,5
	Подготовка конспекта по теме для самостоятельного изучения	проверка конспекта	3,5
7	Подготовка к защите лаб. работ	отчет, защита	3,5
	Самостоятельное изучение материала	конспект	3,5
	Подготовка конспекта по теме для самостоятельного изучения	проверка конспекта	
8	Самостоятельное изучение материала	проверка конспекта	3,5
	Подготовка конспекта по теме для самостоятельного изучения	проверка конспекта	3,5
9	Подготовка к выполнению лаб. работы	допуск к выполнению	3,5
	Самостоятельное изучение материала	проверка конспекта	3,5
10	Самостоятельное изучение материала	проверка конспекта	3,5
	Работа с конспектом лекций, с учебником		3,5
11	Подготовка к коллоквиуму	коллоквиум	5
	Подготовка к контрольной работе	контр. раб.	4,5
12	Самостоятельное изучение материала	проверка конспекта	3,5
	Подготовка конспекта по теме для самостоятельного изучения	проверка конспекта	3,5
13	Подготовка к защите лаб. работ	отчет, защита	4,5
	Самостоятельное изучение материала	проверка конспекта	3,5
14	Самостоятельное изучение материала	проверка конспекта	3,5
	Подготовка конспекта по теме для самостоятельного изучения	проверка конспекта	3,5
15	Подготовка к выполнению лаб. работы	допуск к выполнению	3,5
	Самостоятельное изучение материала	проверка конспекта	
	Подготовка конспекта по теме для самостоятельного изучения	проверка конспекта	3,5
16	Самостоятельное изучение материала	проверка конспекта	3,5
	Работа с конспектом лекций, с учебником		3,5
17	Подготовка к коллоквиуму	коллоквиум	5,5
	Подготовка к защите лаб. работ	отчет, защита	3,5
18	Подготовка к контрольной работе	контр. раб.	3,5
	Подготовка к защите КР		8,5

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ ПО ДИСЦИПЛИНЕ
«Объектно-ориентированное программирование»

**Индексированные результаты обучения и показатели оценивания
Результатов**

Индекс	Результат	Индекс	Показатель
ОПК-1 Способностью устанавливать программное и аппаратное обеспечение для информационных и автоматизированных систем			
Р1. ОПК-1	Знать основы алгоритмических языков программирования	П1. Р1. ОПК-1	Знать основы по алгоритмическим конструкциям, команды языка программирования и средства среды разработки, позволяющие работать с информацией
		П2. Р1. ОПК-1	Знать основы по разработке приложений CLI
ОПК-2 Способностью осваивать методики использования программных средств для решения практических задач			
Р1. ОПК-2	Уметь разрабатывать приложения с графическим интерфейсом	П1. Р1. ОПК-2	Уметь применить типовые алгоритмические конструкции в разработке приложений
		П2. Р1. ОПК-2	Уметь проводить отладку и тестирование приложений

5. ОБРАЗОВАТЕЛЬНЫЕ ТЕХНОЛОГИИ

	В рамках изучения дисциплины предусмотрены следующие образовательные технологии:
5.1	Информационные лекции;
5.2	Практические занятия: а) работа в команде (ИФ) - совместное обсуждение вопросов лекций, домашних заданий, решение и пояснение алгоритмических задач; б) проведение контрольных работ;
5.3	лабораторные работы: – выполнение лабораторных работ в соответствии с индивидуальным графиком; – защита выполненных работ;
5.4	самостоятельная работа студентов: – изучение теоретического материала;

	<ul style="list-style-type: none"> – подготовка к лекциям, лабораторным работам; – работа с учебно-методической литературой; – оформление конспектов лекций, подготовка отчетов; – подготовка к текущему контролю успеваемости, зачету, курсовой работе;
5.5	консультации по всем вопросам учебной программы.

6. ОЦЕНОЧНЫЕ СРЕДСТВА ДЛЯ ТЕКУЩЕГО КОНТРОЛЯ УСПЕВАЕМОСТИ, ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ ПО ИТОГАМ ОСВОЕНИЯ ДИСЦИПЛИНЫ И УЧЕБНО-МЕТОДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ САМОСТОЯТЕЛЬНОЙ РАБОТЫ СТУДЕНТОВ

6.1	Контрольные вопросы и задания
6.1.1	<p>Используемые формы текущего контроля:</p> <ul style="list-style-type: none"> – тестирование; – отчет и защита выполненных лабораторных работ; – курсовое проектирование.
6.1.2	<p>Рабочая программа дисциплины обеспечена фондом оценочных средств для проведения входного, текущего контроля и промежуточной аттестации. Фонд включает примерные варианты контрольных заданий, вопросы к тестам, вопросы к экзамену. Фонд оценочных средств представлен в учебно–методическом комплексе дисциплины.</p> <p>Контрольные вопросы:</p> <ol style="list-style-type: none"> 1. Понятие класса в C++. Применение конструкторов и деструкторов класса 2. Перегруженные и встраиваемые (inline) функции в C++ 3. Области видимости переменных и методов в классе, использование статических (static) объектов 4. Наследование в C++. Примеры простого и множественного наследования 5. Виртуальные функции. Примеры 6. Работа с шаблонными методами и классами в C++ 7. Понятие пользовательского интерфейса, критерии и свойства при разработке приложений 8. Типы пользовательских интерфейсов (CLI и GUI), основные преимущества и области применения 9. Использование специализированных технологий при создании приложений. Пример программы с описанием всех файлов проекта 10. Среда разработки (IDE) (работа с листингом, отладчиком, настройка и сборка проекта) 11. Использование визуальных средств разработки, в которых реализован GUI интерфейс 12. Механизм обработки сообщений в разрабатываемом проекте. Примеры 13. Описание базовых классов для создания окон и диалогов в приложении 14. Разработка SDI приложения 15. Работа с файлами 16. Работа с датой 17. Работа с временем 18. Механизм наследования классов. Показать на примере с произвольным классом 19. Классы для вывода информационных сообщений. Примеры использования 20. Классы по работе со строками. Примеры использования 21. Классы для ввода текста. Примеры использования

	<p>22. Классы для отображения графической и текстовой информации. Примеры использования</p> <p>23. Классы для информирования пользователя о процессе работы приложения. Примеры использования</p> <p>24. Классы по работе с палитрой цветов. Примеры использования</p> <p>25. Разработка собственных диалоговых окон. Модальный и немодальный режим</p> <p>26. Классы для создания кнопок. Примеры использования</p> <p>27. Контейнерные классы</p> <p>28. Структура проектного файла простейшего графического приложения</p> <p>29. Использование классов, предоставляющих подсказки пользователям в приложениях</p> <p>30. Отладочные средства IDE среды при разработке приложений</p>
6.2	<p>Темы письменных работ</p> <ol style="list-style-type: none"> 1. Разработка электронных справочников и средств тестирования 2. Просмотр векторных графических форматов САД систем 3. Создание поисковых утилит и простых файловых менеджеров 4. Расчетные приложения под мобильные платформы 5. Разработка программных средств по управлению БД 6. Базовые подсистемы САПР
3 семестр	
6.2.1	Интегрированные среды разработки
6.2.2	Базовые конструкции языка программирования
6.2.3	Объектно-ориентированный подход к разработке приложений
6.2.4	Разработка приложений с графическим интерфейсом
6.2.5	Создание кроссплатформенных приложений
6.3	Другие виды контроля
6.3.1	Курсовое проектирование. Темы курсовых работ представлены учебно-методическом комплексе дисциплины.

7. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ (МОДУЛЯ)

7.1 Рекомендуемая литература				
№ п/п	Авторы, составители	Заглавие	Годы издания. Вид издания	Обеспечен ность
7.1.1. Основная литература				
7.1.1.1	Бадд Т.	Объектно-ориентированное программирование в действии	2010 печат.	0,2
7.1.1.2	Страуструп Б.	Язык программирования C++	2011 печат.	0,5
7.1.1.3	Прата С.	Язык программирования C++	2007 печат.	0,6
7.1.1.4	Мюссер Д.Р.	C++ и STL: справочное руководство	2010 печат.	0,3
7.1.2. Дополнительная литература				

7.1.2.1	Кордюкова Е.Н.	Информатика: учебное пособие	2007 печат.	0,6
7.1.3 Методические разработки				
7.1.3.1	Юров А.Н.	Методические указания к лабораторным работам № 1-3 по дисциплине “Программирование” 14-2013	2013 магн.	1
7.1.3.2	Юров А.Н.	Методические указания к лабораторным работам № 4-5 по дисциплине “Программирование” 15-2013	2013 магн.	1
7.1.4 Программное обеспечение и интернет ресурсы				
7.1.4.1	Методические указания к выполнению лабораторных работ представлены на сайте: http://calsvstu.ru/index.php/project/uchebnaya-literatura			
7.1.4.2	Компьютерные лабораторные работы: <ul style="list-style-type: none"> – Работа с интегрированными средами разработки ПО (CISCO IT Essentials Virtual Desktop) – Создание приложений с графическим интерфейсом – Обработка событий в приложении – Использование динамических библиотек при разработке приложений – Подготовка инсталляционных пакетов 			
7.1.4.3	Мультимедийные видеофрагменты: <ul style="list-style-type: none"> – Работа с виджетами и компонентами – Отладочные средства среды IDE – Работа со справочной документацией – Компоновка консольных приложений – Создание приложений с графическим интерфейсом – Создание многодокументного приложения – Вывод графических объектов и примитивов – Работа с табличными данными – Подключение к проекту созданию БД – Использование готовых алгоритмов – Использование средств мультимедиа в разрабатываемых приложениях – Кроссплатформенность приложений 			
7.1.4.4	Мультимедийные лекционные демонстрации: <ul style="list-style-type: none"> – Ссылки на ресурсы сети Internet – Конфигурирование среды разработки 			

8. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ (МОДУЛЯ)

8.1	Специализированная лекционная аудитория , оснащенная оборудованием для лекционных демонстраций и проекционной аппаратурой
8.2	Учебные лаборатории: <ul style="list-style-type: none"> – “Лаборатория интеллектуальных систем проектирования” – “Лаборатория компьютерного моделирования и дизайна” – “Лаборатория телекоммуникационных и сетевых технологий”

	<ul style="list-style-type: none"> – “Интернет-лаборатория” – ”Учебный центр ВГТУ, академия Софтлайн, сетевой академии CISCO”
8.3	Дисплейный класс , оснащенный компьютерными программами для проведения лабораторного практикума
8.4	Кабинеты , оборудованные проекторами и интерактивными досками
8.5	Натурные лекционные демонстрации: <ul style="list-style-type: none"> – Разработка приложений под Windows – Создание установочных пакетов – Отладка, сборка и тестирование проектов

ПРИЛОЖЕНИЕ 4

Фонд оценочных средств

Примерные темы курсовых работ:

- Разработка проекта с кроссплатформенным графическим интерфейсом
- Работа с 2D и 3D графикой
- Разработка и визуализация алгоритмов

Вопросы на зачет:

1. Понятие класса в C++. Применение конструкторов и деструкторов класса.
2. Перегруженные и встраиваемые (inline) функции в C++.
3. Области видимости переменных и методов в классе, использование статических (static) объектов.
4. Наследование в C++. Примеры простого и множественного наследования.
5. Виртуальные функции. Примеры.
6. Работа с шаблонными методами и классами в C++.
7. Понятие пользовательского интерфейса, критерии и свойства при разработке приложений.
8. Типы пользовательских интерфейсов (CLI и GUI), основные преимущества и области применения.
9. Использование QT при создании приложений. Пример программы с описанием всех файлов проекта.
10. Среда QT Creator (работа с листингом, отладчиком, настройка и сборка проекта).

11.Использование QT Designer в проектах QT, в которых реализован GUI интерфейс.

12.Механизм сигналов и слотов. Примеры.

13.Описание базовых классов QMainWindow и QDialog.

14.Разработка SDI приложения с помощью QT.

15.Работа с файлами на QT (QFile).

16.Работа с датой (QDate).

17.Работа с временем (QTime).

18. Механизм наследования классов в QT. Показать на примере с произвольным классом.

19. Класс QLabel. Примеры использования.

20. Класс QString. Примеры использования.

21. Класс QLineEdit. Примеры использования.

22. Класс QMessageBox. Примеры использования.

23. Класс QProgressDialog. Примеры использования.

24. Класс QColorDialog. Примеры использования.

25. Разработка собственных диалоговых окон. Модальный и немодальный режим.

26. Класс QPushButton. Примеры использования.

27. Класс QVector. Примеры использования.

28. Контейнерные классы, отличные от QVector.

29. Структура файла .pro простейшего графического приложения на QT.

30. Использование классов, предоставляющих подсказки пользователям в приложениях QT.

Оценочная шкала.

Студент допускается к экзамену при условии сдачи всех лабораторных работ и индивидуальных заданий. На экзамене выдается 2 вопроса из списка «Вопросы к экзамену» и практическое задание. При ответе студента:

- на 1 вопрос выставляется отметка «удовлетворительно»;

- на 2 вопроса выставляется отметка «хорошо»;

- на 2 вопроса и выполнении практического задания выставляется отметка «отлично».