

Государственное образовательное учреждение
высшего профессионального образования
«Приднестровский государственный университет им. Т.Г. Шевченко»

Инженерно-технический институт

Кафедра «Программное обеспечение вычислительной техники
и автоматизированных систем»

УТВЕРЖДАЮ:

Директор института, доцент



Ф.Ю. Бурменко

20 16 г.

РАБОЧАЯ ПРОГРАММА

на 2016/2017 учебный год

УЧЕБНОЙ ДИСЦИПЛИНЫ

Б1.В.ДВ.2.1 «ПРИКЛАДНОЕ ПРОГРАММИРОВАНИЕ»

Направление подготовки:

09.03.04 Программная инженерия

Профиль подготовки

Разработка программно-информационных систем

Для набора

2016 года

Квалификация (степень) выпускника

бакалавр

Форма обучения:

очная

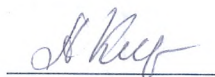
Тирасполь, 2016

Рабочая программа дисциплины «Прикладное программирование» /сост. А.В. Кирсанова – Тирасполь: ГОУ ПГУ, 2016. – 14 с.

Рабочая программа предназначена для преподавания дисциплины по выбору вариативной части цикла Дисциплины (модули) студентам очной формы обучения по направлению подготовки 09.03.04 «ПРОГРАММНАЯ ИНЖЕНЕРИЯ».

Рабочая программа составлена с учетом Федерального Государственного образовательного стандарта высшего образования по направлениям подготовки: 09.03.04 «ПРОГРАММНАЯ ИНЖЕНЕРИЯ», утвержденного приказом Министерства образования и науки Российской Федерации от 12 марта 2015 г. № 229.

Составитель



/Кирсанова А.В., доцент кафедры ПОВТиАС

«25» 08 20 16 г.

© Кирсанова А.В., 2016

© ГОУ ПГУ, 2016

1. Цели и задачи освоения дисциплины

Современный уровень инженерного труда требует хорошего знания вычислительной техники и умения использовать ее в своей практической деятельности. Одним из основных требований к специалистам в этой области является умение программировать и знание языков программирования.

Курс «Прикладное программирование» относится к тем дисциплинам, которые закладывают основу знаний по программированию на языках высокого уровня для всех последующих курсов, связанных с программированием.

Цели дисциплины:

- приобретение теоретических знаний и практических навыков разработки, отладки и тестирования программного обеспечения на языках высокого уровня;

- освоение студентами методики постановки, подготовки и решения инженерно-технических задач на современных компьютерах, создание необходимой фундаментальной основы знаний, необходимой при программировании для вычислительных систем.

Задачи дисциплины: изучение способов построения и представления алгоритмов, основных алгоритмических конструкций, основ универсального языка программирования высокого уровня.

2. Место дисциплины в структуре ООП ВО

Б1.В.ДВ.2.1. Дисциплины (модули). Вариативная часть. Дисциплины по выбору.

Общая трудоемкость дисциплины составляет 5 зачетных единиц, 180 часов.

Для освоения дисциплины «Прикладное программирование» студенты используют знания, умения, навыки, способы деятельности и установки, полученные и сформированные в ходе изучения дисциплины «Информатика» и «Основы программирования».

Изучение дисциплины «Прикладное программирование» является базой для дальнейшего освоения дисциплин «Программирование», «Алгоритмы обработки данных», «Теория языков программирования», «Компьютерная графика», «Базы данных», курсов по выбору профессионального цикла, а также для прохождения практики.

3. Требования к результатам освоения дисциплины

Изучение дисциплины направлено на формирование следующих компетенций: ПК-1, ПК-3. Расшифровка компетенций дана в следующих таблицах.

Таблица – Формулировка компетенции

Код компетенции	Формулировка компетенции
ПК-1	готовность применять основные методы и инструменты разработки программного обеспечения;
ПК-3	владение навыками использования различных технологий разработки программного обеспечения

В результате освоения дисциплины обучающийся должен:

3.1. Знать:

- основные принципы алгоритмизации;
- принципы построения алгоритмов;
- этапы разработки программ и методы автоматизации программирования;
- структуру программ на языке;
- типы данных и базовые конструкции изучаемых языков программирования;
- основные приемы программирования;
- интегрированные среды изучаемых языков программирования;
- основные принципы и методологию разработки прикладного программного обеспечения, типовые способы организации программных данных, а также типовые подходы к построению программных алгоритмов;
- синтаксис и семантику универсального алгоритмического языка программирования высокого уровня;
- жизненный цикл программного продукта.

3.2. Уметь:

- составлять простые блок-схемы алгоритмов и создавать схему алгоритма для любой функциональной задачи;
- осуществлять декомпозицию решения задачи и составлять алгоритмы отдельных его частей в соответствии с современной технологией программирования;
- применять основные операторы языка C#, общие для всех языков программирования;
- провести отладку и тестирование созданного программного продукта
- средствами изученных интегрированных сред;
- работать с ресурсами компьютера программными средствами;
- составлять программы на языке высокого уровня;
- работать в интегрированной среде изучаемых языков программирования.
- согласованно решать задачи разработки эффективных моделей данных и алгоритмов их обработки при создании прикладного программного обеспечения, а также получать программные реализации полученных решений на языке высокого уровня:

3.3. Владеть:

- навыками постановки задачи и разработки спецификации программы;
- навыками составления, отладки, испытания и документирования программ с использованием процедурного языка;
- навыками работы в интегрированных средах программирования и с использованием библиотек.

Рабочая программа учебной дисциплины рассчитана на 86 часов аудиторных занятий, в том числе 34 часа отводится на лекционные занятия, 34 часа – на лабораторные занятия, 18 – на практические занятия.

С целью систематизации и закрепления полученных теоретических знаний и практических умений в рабочей программе учебной дисциплины предусмотрена самостоятельная работа студентов. Внеаудиторная самостоятельная работа студентов – 56 часов. На подготовку к экзамену – 36 часов.

Для проверки знаний студентов в рабочей программе указаны по окончании изучения каких разделов следует проводить рубежный контроль. Учебная дисциплина изучается один семестр и заканчивается промежуточной аттестацией студентов в форме экзамена.

Формированию отмеченных знаний, умений и владений соответствуют разделы дисциплины. Ее изучение предполагает, что студенты знакомы с принципами работы компьютера, десятичной, двоичной, восьмеричной и шестнадцатеричной системами счисления, а также основными понятиями информатики.

4. Структура и содержание дисциплины (модуля)

4.1. Распределение трудоемкости в часах по видам аудиторной и самостоятельной работы студента по семестрам

Семестр	Трудоемкость з.е./часы	Количество часов						Форма итогового контроля
		В том числе				Самост. работа	Экзамен	
		Аудиторных						
		Всего	Лекции	Лаб. раб.	Практич. занятия			
2	5/180	86	34	34	18	58	36	Экзамен
Итого	5/180	86	34	34	18	58	36	

4.2. Распределение видов учебной работы и их трудоемкости по разделам дисциплины

№ п/п	Наименование разделов	Количество часов				
		Всего	Аудиторная работа			Внеаудиторная работа (СР)
			Л	ПЗ	ЛР	
1	Технологии разработки прикладного программного обеспечения.	8	2	-	-	6
2	Основы прикладного программирования с использованием языка программирования высокого уровня С#.	24	8	2	6	8
3	Реализация объектно-ориентированного программирования.	46	10	8	14	14
4	Пользовательский интерфейс прикладных программ.	6	2	-	-	4
5	Организация разработки прикладного программного обеспечения.	60	12	8	14	26
	Итого	144	34	18	34	58
	Подготовка к экзамену	36	-	-	-	-
	Всего	180	34	18	34	58

4.3. Тематический план по видам учебной деятельности

Лекции

№ п/п	Номер раздела дисциплины	Объем часов	Тема лекции	Учебно-наглядные Пособия
1	2	3	4	5
1	1	2	Обзор современных инструментальных средств визуального программирования.	презентация в <i>Power Point</i>
2	2	2	Основные понятия объектно-ориентированного программирования. Пример программы на C#.	презентация в <i>Power Point</i>
3	2	2	Состав языка C#. Типы данных. Форматирование	презентация в <i>Power Point</i>
4	2	2	Операции на C#. Преобразование типов.	презентация в <i>Power Point</i>
5	2	2	Класс <i>Math</i> . Операторы языка C#	презентация в <i>Power Point</i>
6	3	2	Понятие директивы. Виды директив.	презентация в <i>Power Point</i>
7	3	2	Процедуры и функции. Перегрузка	презентация в <i>Power Point</i>
8	3	2	Локальные и глобальные переменные	презентация в <i>Power Point</i>
9	3	2	Виды исключений. Генерация собственных исключений	презентация в <i>Power Point</i>
10	3	2	Одно- и двумерные прямоугольные массивы	презентация в <i>Power Point</i>
11	4	2	Ступенчатые массивы	презентация в <i>Power Point</i>
12	5	2	Символы <i>char</i>	презентация в <i>Power Point</i>
13	5	2	Неизменяемые строки <i>String</i> Изменяемые строки <i>StringBuilder</i>	презентация в <i>Power Point</i>
14	5	2	Метасимволы в регулярных выражениях. Класс <i>Regex</i>	презентация в <i>Power Point</i>
15	5	2	Байтовый, символьный потоки	презентация в <i>Power Point</i>
16	5	2	Двоичный поток. Перенаправление стандартных потоков	презентация в <i>Power Point</i>
17	5	2	Работа с каталогами, файлами.	презентация в <i>Power Point</i>
	Итого	34		

Практические занятия

№ п/п	Номер раздела дисциплины	Объем часов	Тема практического занятия	Учебно-наглядные пособия
1	2	3	4	5
1	2	2	Решение задач на линейные, ветвящиеся и циклические алгоритмы	презентация в <i>Power Point</i>
2	3	2	Функции и процедуры.	презентация в <i>Power Point</i>
3	3	2	Одномерные и многомерные массивы	презентация в <i>Power Point</i>
4	3	2	Строки в C#. Классы <i>Char[], String, String-Builder</i> и их методы.	презентация в <i>Power Point</i>
5	3	2	Регулярные выражения.	презентация в <i>Power Point</i>
6	5	2	Регулярные выражения.	презентация в <i>Power Point</i>
7	5	2	Байтовый, символьный потоки.	презентация в <i>Power Point</i>
8	5	2	Двоичный поток. Файловая система.	презентация в <i>Power Point</i>
9	5	2	Работа с каталогами, файлами.	презентация в <i>Power Point</i>
	Итого	18		

Лабораторные работы

№ п/п	Номер раздела дисциплины	Объем часов	Тема лабораторного занятия	Учебно-наглядные пособия
1	2	3	4	5
1	2	2	Решение линейных задач.	Электр. вариант лаб.раб.
2	2	2	Решение задач на ветвящиеся алгоритмы	Электр. вариант лаб.раб.
3	2	2	Решение задач с использованием <i>switch</i>	Электр. вариант лаб.раб.
4	3	2	Решение циклических задач	Электр. вариант лаб.раб.
5	3	2	Функции и процедуры. Различные способы передачи параметров в функцию.	Электр. вариант лаб.раб.
6	3	2	Функции и процедуры. Различные способы передачи параметров в функцию.	Электр. вариант лаб.раб.
7	3	2	Обработка исключений	Электр. вариант лаб.раб.
8	3	2	Обработка исключений	Электр. вариант лаб.раб.
9	3	2	Одномерные и многомерные массивы в C#. Методы класса <i>Array</i>	Электр. вариант лаб.раб.

1		3	4	5
10	3	2	Одномерные и многомерные массивы в C#. Методы класса <i>Array</i>	Электр. вариант лаб.раб.
11	5	2	Строки в C#. Классы <i>Char[]</i> , <i>String</i>	Электр. вариант лаб.раб.
12	5	2	Класс <i>StringBuilder</i> и его методы	Электр. вариант лаб.раб.
13	5	2	Регулярные выражения	Электр. вариант лаб.раб.
14	5	2	Регулярные выражения	Электр. вариант
15	5	2	Байтовый, символьный и двоичный потоки.	Электр. вариант
16	5	2	Байтовый, символьный и двоичный потоки. Перенаправление стандартных потоков	Электр. вариант лаб.раб.
17	5	2	Работа с каталогами. Работа с дисками Работа с файлами: запись и чтение данных в/из файла.	Электр. вариант лаб.раб.
	Итого	34		

Лабораторные работы включают: создание алгоритма решения задачи, разработку интерфейса приложения, написание программного кода для реализации алгоритма и защиты приложения от ошибок ввода, устранение ошибок написания кода, устранение ошибок времени выполнения, тестирование алгоритма, отладку приложения, создание отчёта по лабораторной работе в соответствии с требованиями, защита выполненной работы.

Самостоятельная работа студента

Раздел дисциплины	№ п/п	Тема и вид СРС	Трудоемкость (в часах)
1	2	3	4
Раздел 1	1	Тема. Организация ввода-вывода данных. Форматирование СРС №1. Работа со справочной и дополнительной литературой. Подготовка презентации по теме «Форматирование».	4
	2	Тема. Платформа <i>Microsoft .Net Framework 2.0</i> . СРС №2. Подготовка презентации по теме «Языки программирования, эволюция».	2
Раздел 2	3	Тема. Решение задач на линейные алгоритмы и алгоритмы ветвления на C#. СРС №3. Составление программ линейной, разветвляющейся и циклической структуры. Выполнение индивидуальных заданий по решению задач.	4
	4	Тема. Циклические алгоритмы на C#. СРС №4. Составление программ линейной, разветвляющейся и циклической структуры. Выполнение индивидуальных заданий по решению задач.	4

Раздел 3	5	Тема. Директивы препроцессора. СРС №5. Составление опорного конспекта по теме.	2
	6	Тема. Функции и процедуры. СРС №6. Составление опорного конспекта по теме «Процедуры и функции». Выполнение индивидуальных заданий по решению задач.	2
	7	Тема. Различные способы передачи параметров в функцию. СРС №7. Выполнение индивидуальных заданий по решению задач.	2
	8	Тема. Генерация собственных исключений. СРС №8. Составление опорного конспекта по темам: - Генерация собственных исключений файлы. - Обработка исключений Выполнение индивидуального задания	2
	9	Тема. Организация работы с одномерными массивами. СРС №9. Составление опорного конспекта по темам: – Организация работы с одномерными массивами	2
	10	Тема. Организация работы с двумерными прямоугольными массивами. СРС №10. Составление опорного конспекта по темам: – Организация работы с двумерными прямоугольными массивами.	2
	11	Тема. Ступенчатые массивы. СРС №11. Составление опорного конспекта по теме «Ступенчатые массивы».	2
Раздел 4	12	Тема. Пользовательский интерфейс прикладных программ. СРС №12. Составление опорного конспекта по теме. Создание презентации.	4
Раздел 5	13	Тема. Класс <i>Char[]</i> и его методы СРС №13. Составление опорного конспекта по теме «Класс <i>Char[]</i> и его методы».	2
	14	Тема. Класс <i>String</i> и его методы СРС №14. Составление опорного конспекта по теме «Класс <i>String</i> и его методы».	2
	15	Тема. Класс <i>StringBuilder</i> и его методы СРС №15. СРС №1. Составление опорного конспекта по теме «Класс <i>StringBuilder</i> и его методы».	4
	16	Тема. Регулярные выражения СРС №16. Составление опорного конспекта по теме «Регулярные выражения» Создание проектов по индивидуальному заданию.	4

17	Тема. Байтовый, символьный, двоичный потоки. СРС №17. Составление опорного конспекта по теме – Байтовый, символьный и двоичный потоки. Создание проектов по индивидуальному заданию.	4
18	Тема. Перенаправление стандартных потоков СРС №18. Составление опорного конспекта по теме – Перенаправление стандартных потоков Создание проектов по индивидуальному заданию.	2
19	Тема. Работа с каталогами СРС №19. Составление опорного конспекта по теме – Работа с каталогами Создание проектов по индивидуальному заданию.	2
20	Тема. Работа с дисками СРС №20. Составление опорного конспекта по теме – Работа с дисками Создание проектов по индивидуальному заданию.	2
21	Тема. Работа с файлами: запись и чтение данных в/из файла. СРС №21. Составление опорного конспекта по теме – Работа с файлами: запись и чтение данных в/из файла. Создание проектов по индивидуальному заданию.	4
Итого		58

5. Примерная тематика курсовых проектов (работ)

Курсовых работ не предусмотрено

6. Образовательные технологии

Семестр	Вид занятия (Л, ПР, ЛР)	Используемые интерактивные образовательные технологии	Количество часов
2	Л	- информационно-развивающие технологии; - компьютерные технологии обучения (проблемная лекция, лекция-дискуссия (лекция-обсуждение), комплексная лекция (лекция-панель, лекция вдвоем), письменная программированная лекция, лекция-визуализация, лекция с заранее запланированными ошибками (метод контрольного изложения), лекция-конференция.	34
	ПР	- задачная (поисково-исследовательская) технология; - технология коллективной мыслительной деятельности; - компьютерные технологии обучения; - метод аналогии, теория решения изобретательских задач; - мозговая атака или мозговой штурм.	18
	ЛР	- компьютерные технологии обучения деятельностные; - технология учебного проектирования;	34
Итого			86

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

Пример билета к модульному контролю № 1.

1. Какие компоненты включает в себя *.NET Framework*?
2. Расшифруйте и сформулируйте назначение следующих аббревиатур *MSIL*, *JIT*, *CTS*.
3. Дайте определения понятиям: инкапсуляция, наследование и полиморфизм. Приведите пример полиморфизма.
4. Каково «взаимоотношение» между классом и объектом?
5. Показать на примере в чем состоит отличие в работе оператора множественного выбора в C++ и в C#.
6. Решить одну из предложенных задач (задачи даны по возрастанию количества баллов)
 - 6.1. Стороны одного прямоугольника равны A и B . Стороны другого равны X и Y . Написать алгоритм и программу проверки прямоугольников на равенство.
 - 6.2. Тело имеет массу M г и объем V см³. Будет ли оно плавать в жидкости, объем которой равен V_1 см³, а масса M_1 г? (Указание: найдите плотности тела и жидкости, а затем сравните их).
 - 6.3. Смекалкин, уходя в школу, вышел из дому на n минут позже младшего брата. Расстояние до школы S метров. Смекалкин идет со скоростью V_1 м/мин, а его брат – V_2 м/мин. Догонит ли Смекалкин брата, прежде чем тот придет в школу?

Пример билета к модульному контролю № 2.

1. Особенности хранения одномерного массива в C# (в памяти ЭВМ). Способы описания (объявления) и инициализации одномерного массива.
2. Описание формальных аргументов (способ передачи параметра в метод).
3. Тип `char[]` и его отличительные особенности. Преобразование типа `String` в тип `char[]` и наоборот. Методы класса `char[]`.
4. Написать программу (одну из трех), которая
 - 4.1) находит в строке количество знаков пунктуации.
 - 4.2) в строке после каждого слова добавляет число, указывающее сколько раз встретилось это слово в строке. Например: Роза2 красивый2 цветок2. Да1 роза2 самый1 красивый2 цветок2!
 - 4.3) находит в строке самое длинное слово и определяет, сколько раз оно встретилось в тексте. Считать, что такое слово одно.

ВОПРОСЫ К ЭКЗАМЕНУ

1. Типы C#. Категории, на которые делятся все типы C#. Перечислить базовые типы.
2. Явное и неявное (автоматическое) преобразование типов. Привести примеры.
3. Проверяемые преобразования. Операции `checked` и `unchecked`. Привести примеры.
4. Виды констант. Привести примеры.
5. Перечисления: назначение, синтаксис, примеры. Привести примеры.
6. Организация ввода данных в программу и вывода данных. Привести примеры.
7. Управление форматом вывода данных. Спецификаторы вывода. Привести примеры.
8. Процедуры и функции. Отличия. Привести примеры.
9. Описание методов (процедур и функций). Синтаксис. Привести примеры.
10. Описание формальных аргументов. Синтаксис. Виды аргументов метода. Привести примеры.
11. Перегрузка методов. Привести примеры.
12. Виды исключений. Стандартные исключения. Привести примеры.
13. Способы объявления и инициализации одномерных массивов. Привести примеры.
14. Методы и свойства класса `Array`. Привести примеры использования.
15. Способы объявления и инициализации многомерных массивов: прямоугольный, и ступенчатые массивы.
16. Символьный тип `char`, методы класса `Char()`.
17. Неизменяемые строки `string`. Способы создания. Методы класса `String()`.
18. Изменяемые строки `StringBuilder`. Способы создания. Методы класса `StringBuilder()`.
19. Метасимволы в регулярных выражениях.
20. Конструкторы для описания регулярного выражения. Поиск в тексте по шаблону (методы).
21. Редактирование текста (методы класса `Regex()`).
22. Организация C#-системы ввода-вывода: дать определение потока, объяснить иерархию потоков ввода-вывода. Перенаправление стандартных потоков.
23. Работа с каталогами: абстрактный класс `FileInfo`, классы `DirectoryInfo` и `Directory`.
24. Работа с дисками. Класс `DriveInfo`.
25. Работа с файлами: абстрактный класс `FileInfo`, классы `FileInfo` и `File`.

8. Учебно-методическое и информационное обеспечение дисциплины (модуля)

8.1. Основная литература

1. Гвоздева В. Введение в специальность программиста: учебник. - М.: ИД ФОРУМ: ИНФРА-М, 2010. - 208 с.
2. Пол Дейтел, Харви Дейтел. Как программировать на Visual C# 2012, 5-е издание. - СПб.: Питер, 2013.
3. Джон Шарп. Microsoft Visual C#. Подробное руководство. 8-е изд (+ Examples). - СПб.: Питер, 2017.
4. Мартин Дрейер C# для школьников. Учебное пособие. - М.: БИНОМ. Лаборатория знаний, 2009.

2.2. Дополнительная литература

1. Либерти Джесс. Программирование на C#. Пер. с англ. - 2-е изд, СПб.: Издательство «Символ-плюс», 2012. - 684 с.
2. Либерти Джесс. Создание Net приложений. Программирование на C#. 2010
3. Павловская Т.А. C#. Программирование на языке высокого уровня. Учебник для вузов. - СПб.: Питер, 2012. - 432 с.
4. Рихтер Джеффри. Программирование на платформе Microsoft .NET Framework 2.0 на языке C#. Мастер класс./Пер. с англ. — 2-е изд., испр. — М.: Издательско-торговый дом «Русская Редакция», 2011. - 656 с.

8.3. Программное обеспечение и Интернет-ресурсы

Интернет-ресурсы

- 1) msdn.microsoft.com/ru
- 2) alleng.ru
- 3) intuit.ru

Программное обеспечение:

ОС Windows, Visual Studio.NET.

Программное обеспечение: ОС Windows, пакет MS Office.

8.4. Методические указания и материалы по видам занятий

- 1) Кирсанова А.В. Сборник задач по программированию. Часть 1. ПГУ им. Т.Г.Шевченко, 2009, 2,1 п.л.

9. Материально-техническое обеспечение дисциплины (модуля):

Учебный кабинет, лаборатория ИТО ИТИ.

10. Методические рекомендации по организации изучения дисциплины:

Обучающийся должен овладеть теоретическими знаниями по дисциплине, а также иметь навыки применения полученных знаний на практике.

По окончании курса обучающийся должен знать определения и термины, составляющие основу понятийного аппарата дисциплины.

Освоение курса требует самостоятельной работы обучающегося. В программе предусмотрено и отведено время, необходимое для работы обучающегося над темой.

Самостоятельная работа включает:

- изучение и конспектирование рекомендованной литературы;

- анализ и проработку учебного материала по рекомендованной литературе и конспектам лекций;
- работа со справочной и дополнительной литературой;
- подготовка презентаций;
- решение задач;
- подготовку к экзамену.

Рабочая учебная программа по дисциплине «Прикладное программирование» составлена в соответствии с требованиями Федерального Государственного образовательного стандарта ВО по направлению 09.03.04 «ПРОГРАММНАЯ ИНЖЕНЕРИЯ» и учебного плана по профилю подготовки «Разработка программно-информационных систем».

ТЕХНОЛОГИЧЕСКАЯ КАРТА ДИСЦИПЛИНЫ

Курс 1

Семестр 2

Группа ИТ16ДР62ПИ1

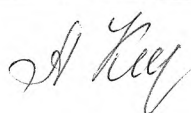
Преподаватель – лектор Кирсанова А.В.

Преподаватели, ведущие практические занятия – Белоконов О.С.

Кафедра программного обеспечения вычислительной техники и автоматизированных систем

Наименование дисциплины/курса	Уровень образования (бакалавриат, специалитет, магистратура)	Статус дисциплины в учебном плане (А, Б, В)	Количество зачетных единиц	
Прикладное программирование	бакалавриат	А	5	
СМЕЖНЫЕ ДИСЦИПЛИНЫ ПО УЧЕБНОМУ ПЛАНУ:				
Информатика, Основы программирования				
БАЗОВЫЙ МОДУЛЬ (проверка знаний и умений по дисциплине)				
Тема, задание или мероприятие текущего контроля	Виды текущей аттестации	Аудиторная или внеаудиторная	Минимальное количество баллов	Максимальное количество баллов
1-ый календарный модуль	Тест, ПЗ	аудиторная	13	25
Лабораторная работа №1	ЛР1	аудиторная	2	5
Лабораторная работа №2	ЛР2	аудиторная	3	5
Лабораторная работа №3	ЛР3	аудиторная	2	5
РУБЕЖНЫЙ КОНТРОЛЬ	РК		20	40
2-ый календарный модуль	Тест, ПЗ	аудиторная	18	35
Лабораторная работа №4	ЛР4	аудиторная	2	5
Лабораторная работа №5	ЛР5	аудиторная	2	5
Лабораторная работа №6	ЛР6	аудиторная	2	5
Лабораторная работа №7	ЛР7	аудиторная	3	5
Лабораторная работа №8	ЛР8	аудиторная	3	5
РУБЕЖНАЯ АТТЕСТАЦИЯ	РА		30	60
		Итого	50	100

Составитель, доцент



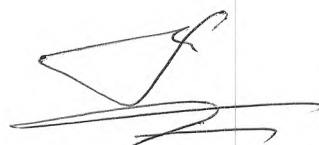
А.В. Кирсанова

Рабочая учебная программа рассмотрена методической комиссией инженерно-технического института протокол № 4 от «23» 09 2016 г. и признана соответствующей требованиям Федерального Государственного образовательного стандарта и учебного плана по направлению 09.03.04 «ПРОГРАММНАЯ ИНЖЕНЕРИЯ».

Председатель МК ИТИ

Е.И. Андрианова

Зав. кафедрой ПОВТ и АС, доцент



С.Г. Федорченко

