МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное учреждение высшего образования

«ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ СИСТЕМ УПРАВЛЕНИЯ И РАДИОЭЛЕКТРОНИКИ» (ТУСУР)

УТВЕРЖДАЮ
Проректор по учебной работе
Н

Документ подписан электронной подписью
Сертификат: 1c6cfa0a-52a6-4f49-aef0-5584d3fd4820
Владелец: Троян Павел Ефимович
Действителен: с 19.01.2016 по 16.09.2019

г.

РАБОЧАЯ ПРОГРАММА УЧЕБНОЙ ДИСЦИПЛИНЫ

Информационные технологии

Уровень основной образовательной программы **бакалавриат**
Направление(я) подготовки (специальность) **27.03.05 Инноватика**
Профиль(и) **Управление инновациями в электронной технике**
(полное наименование профиля направления подготовки (специальности) из ПООП)
Форма обучения очная
Факультет Инновационных технологий (ИТ)
Кафедра Управления инновациями
Курс 1 Семестр 2

Учебный план набора 2013 года и последующих лет.

Распределение рабочего времени:

№	Виды учебной работы	Семестр 1	Семестр 2	Семестр 3	Семестр 4	Семестр 5	Семестр 6	Семестр 7	Семестр 8	Всего	Единицы
1	Лекции		18							18	часов
2	Лабораторные работы		36							36	часов
3	Практические занятия									0	часов
4	Курсовой проект/работа (КРС) (аудиторная)										часов
5	Всего аудиторных занятий		54							54	часов
6	Из них в интерактивной форме		11							11	часов
7	Самостоятельная работа студентов		54							54	часов
8	Всего (без экзамена)		108							108	часов
9	Самост. работа на подготовку, сдачу экзамена		36							36	часов
10	Общая трудоемкость		144							144	часов
	(в зачетных единицах)		4							4	ЗЕТ

Зачет ____ семестр Диф. зачет _____ семестр

Экзамен 2 семестр

Томск 2016

Согласована на портале № 9559

Рассмотрена и одобрена на заседании кафедры
протокол № 13 от « 21 » 12 2016 г.

Лист согласований

Рабочая программа составлена с учетом требований Федерального Государственного образовательного стандарта высшего профессионального образования (ФГОС ВО) третьего поколения по направлению подготовки (специальности) **27.03.05 Инноватика**, утвержденного **11 августа 2016г**., рассмотрена и утверждена на заседании кафедры « *21* » *декабря* 20*16* г., протокол № *13*

Разработчик	ст. преподаватель, каф ПМИ (должность, кафедра)	_____ (подпись)	А.В. Гураков (Ф.И.О.)
Зав. кафедрой	профессор, каф. ПМИ (должность, кафедра)	_____ (подпись)	В.В. Кручинин (Ф.И.О.)

Рабочая программа согласована с факультетом, профилирующей и выпускающей кафедрами направления подготовки (специальности).

Декан ФИТ	_____ (подпись)	Г.Н. Нариманова (Ф.И.О.)
Зав. профилирующей кафедрой УИ	_____ (подпись)	Г.Н. Нариманова (Ф.И.О.)
Зав. выпускающей кафедрой УИ	_____ (подпись)	Г.Н. Нариманова (Ф.И.О.)

Эксперты:

_____ (место работы, занимаемая должность)	_____ (подпись)	(Ф.И.О.)
_____ (место работы, занимаемая должность)	_____ (подпись)	(Ф.И.О.)

Согласована на портале № 9559

1 Цели и задачи дисциплины, её место в учебном процессе

Целью изучения дисциплины «Информационные технологии» является получение теоретических, практических знаний, основных навыков работы на персональном компьютере (ПК), а также освоение программных средств реализации информационных процессов, базового программного обеспечения (ПО), изучение языков высокого уровня. К теоретическим знаниям относятся рассмотрение понятий, методов, используемых при дальнейшем знакомстве работы с ПК. Практические знания – использование теоретических знаний при работе с ПК. Базовое ПО – пакеты программ, используемые при закреплении полученных теоретических знаний и практических навыков работы.

2 Место дисциплины в структуре ООП

Дисциплина является базовой и относится к математическому и естественнонаучному циклу. Для успешного освоения данной дисциплины необходимо и достаточно знаний и умений, приобретенных студентами при изучении дисциплин «Информатика», «Математика. Знания и умения, полученные студентами при успешном освоении курса, будут востребованы при дальнейшем изучении следующих дисциплин учебного плана: «Алгоритмические языки и программирование», «Веб-программирование».

3 Требования к результатам освоения дисциплины

Процесс изучения дисциплины направлен на формирование следующих компетенций, характеризующихся:

- способностью использовать инструментальные средства (ОПК-2);

В результате изучения дисциплины «Информационные технологии» студент должен:

***знать*:**

- принципы алгоритмизации и программирования;
- процесс разработки программного обеспечения;
- принципы построения алгоритмов;
- основные управляющие структуры и способы описания алгоритмов с использованием различных нотаций;
- понятие типа данных, форматы представления данных при решении задач с помощью компьютера;
- синтаксис и семантику алгоритмического языка программирования высокого уровня;
- основные алгоритмы типовых численных методов решения математических задач;
- основные алгоритмы сортировки и поиска данных.

***уметь*:**

Согласована на портале № 9559

- работать в современных средах разработки, составлять блок-схемы алгоритмов, создавать программы на структурном языке программирования высокого уровня;
- решать задачи, используя различные методы разработки алгоритмов и выбирая наиболее подходящие алгоритмы и средства их реализации в зависимости от постановки задачи;
- разрабатывать программы на языке программирования высокого уровня с использованием основных управляющих конструкций и стандартных типов данных;
- использовать языки и системы программирования для решения профессиональных задач.

***владеть*:**

- навыками разработки и анализа алгоритмов решения типовых задач (сортировки и поиска данных и пр.), исследования их свойств;
- методами и инструментальными средствами разработки программ на языке программирования высокого уровня, их тестирования и отладки;
- навыками использования инструментальных программных средств в процессе разработки и сопровождения программных продуктов;
- навыками самостоятельного решения задач с помощью компьютеров, изучения новых средств разработки программ.

4 Объем часов и виды учебной деятельности

Общая трудоемкость дисциплины составляет 4 зачетных единиц.

Вид учебной деятельности		Всего часов	Семестры	
			1	2
Аудиторные занятия (всего)		**54**		**54**
В том числе:				
Лекции		18		18
Лабораторные работы (ЛР)		36		36
Самостоятельная работа (всего)		**54**		**54**
В том числе:				
Подготовка к лабораторным работам		34		34
Подготовка к контрольным работам		20		20
Подготовка к экзамену		36		36
Вид промежуточной аттестации				экз.
Общая трудоемкость	час.	144		144
	ЗЕТ	4		4

5 Содержание дисциплины

5.1 Разделы дисциплин и виды занятий

№	Наименование раздела дисциплины	Лекции	ПЗ	ЛР	СРС	Всего	ОК, ПК

Согласована на портале № 9559

1	Алгоритмы и алгоритмизация	4			10	14	ОПК-2
2	Понятие языка высокого уровня	2			10	12	ОПК-2
3	Программирование	12		36	34	82	ОПК-2

5.2 Содержание разделов дисциплины

Содержание курса соответствует уровню развития современной техники и науки.

№	Наименование разделов	Содержание разделов	Труд. (час.)	ОК, ПК
1	**Алгоритмы и алгоритмизация**	Понятие алгоритма и алгоритмической системы. Две формы представления алгоритмов: визуальная и текстовая. Свойства алгоритмов. Визуализация алгоритмов и блок-схемы. Линейные, разветвленные и цикличные алгоритмы. Вложенные и параллельные алгоритмы. Логические элементы и базовые управляющие структуры визуального структурного программирования. Построение алгоритма из базовых структур. Пошаговая детализация как метод проектирования алгоритмов.	4	ОПК-2
2	**Понятие языка высокого уровня**	Программа как изображение алгоритма в терминах команд, управляющих работой компьютера. Структура программы. Трансляция и компоновка. Исходный и объектный модули, исполняемая программа. Компиляция и интерпретация.	2	ОПК-2
3	**Программирование**	Данные как объект обработки. Типы данных, способы и механизмы управления данными. Элементы и структуры данных, алфавит, идентификаторы, выражения, операции, операторы, реализация логических структур на примере языка FreePascal. Операции с массивами. Ввод и вывод массивов. Алгоритмы поиска и упорядочения массива. Строки. Записи. Программы и подпрограммы. Подпрограммы, их назначение и классификация. Оформление подпрограмм, обращение к ним, передача параметров. Стандартные и пользовательские модули. Файловые типы данных. Файловые переменные. Основные операции с файлами. Примеры работы с файлами. Текстовые файлы.	12	ОПК-2

5.3 Разделы дисциплины и междисциплинарные связи с обеспечивающими и обеспечиваемыми дисциплинами

№	Наименование обеспечивающих дисциплин	Номера разделов		
		1	2	3
Предшествующие дисциплины				
1	Информатика	+	+	+
2	Математика	+	+	+
Последующие дисциплины				
1	Алгоритмические языки и программирование	+	+	+
2	Веб-программирование	+	+	+

5.4 Соответствие компетенций, формируемых при изучении дисциплины, и видов занятий

Перечень компетенций	Виды занятий				Формы контроля
	Лек.	Лаб.	Пр.	СРС	
ОПК-2	+	+	+	+	Отчет по ЛР, контрольная

Согласована на портале № 9559

					работа, опрос на лекции.

6 Методы и формы организации обучения

Организация учебного процесса по курсу «Информационные технологии» реализована в форме лекций и лабораторных занятий. В основе проектирования учебного курса положены: принцип научности, систематичности и последовательности, системности, принцип междисциплинарных связей, связь теории и практики с жизнью, принцип наглядности, доступности, принцип создания положительного отношения к учению и мотивации.

Технологии интерактивного обучения при разных формах занятий в часах

Формы / Методы	Краткое описание	Лекции	ЛР	СРС	Всего
Работа в сотрудничестве	Используется вариант «работа в команде», в котором особое внимание уделяется групповым целям и успеху всей группы.	1	4	2	7
Игры (конкурсы)	Форма деятельности в условных ситуациях, направленная на воссоздание и усвоение общественного опыта.	1	3	2	6
Исследовательский	Предполагает построение процесса обучения наподобие процесса научного исследования, в упрощенной форме: выявление неизвестных (неясных) фактов; уточнение и формулировка проблемы; выдвижение гипотез и т.д.	2			2
Итого:		4	7	4	15

7 Лабораторный практикум

№	№ раздела дисциплины	Наименование лабораторных работ	Трудоемкость (час.)	ОК, ПК
1	1-3	Запись линейных и циклических алгоритмов на языке высокого уровня.	6	ОПК-2
2	1-3	Работа со строками.	6	ОПК-2
3	1-3	Создание программ с использованием процедур и функций. Модули.	6	ОПК-2
4	1-3	Создание программ с использованием структурных типов данных (строки, массивы, записи).	6	ОПК-2
5	1-3	Сортировка одномерных массивов.	6	ОПК-2
6	1-3	Обработка одномерных и двумерных массивов.	6	ОПК-2

8 Практические занятия

Не предусмотрены учебным планом.

9 Самостоятельная работа

№	№ раздела	Тематика самостоятельной работы	Трудоемкость	ОК, ПК	Контроль
1	1	Подготовка к контрольной работе «**Алгоритмы и алгоритмизация**».	18	ОПК-2	Контрольная работа.

Согласована на портале № 9559

2	2	Подготовка к контрольной работе «**Понятие языка высокого уровня**».	18	ОПК-2	Контрольная работа.
3	3	Подготовка к контрольным работам «**Программирование**». Подготовка к лабораторным работам.	18	ОПК-2	Контрольная работа. Отчет по лабораторным работам
4	1-3	Подготовка и сдача экзамена (зачета)	36	ОПК-2	Оценка на экзамене

10. Примерная тематика курсовых проектов (работ)

Не предусмотрены учебным планом.

11. Рейтинговая система оценки успеваемости студентов

Текущий контроль успеваемости осуществляется в течение семестра в виде контрольных и лабораторных работ. Промежуточная аттестация проводится в форме экзамена в течение сессии, в сроки, установленные ВУЗом.

Оценка знаний и умений осуществляется в виде рейтинга. Рейтинг является накопительным, т.е. баллы, выставляемые студенту по каждому элементу контроля и экзамену, суммируются и образуют итоговый рейтинг по дисциплине.

Максимальное количество баллов, которые студент может набрать в течение семестра, равен 100. Из них 30 баллов на экзамене. Для допуска к экзамену, обучающимся необходимо набрать более 30% от максимального количества баллов и выполнить все лабораторные работы.

Распределение баллов для дисциплины в семестре

Элементы учебной деятельности	Максимальный балл на 1-ую КТ с начала семестра	Максимальный балл за период между 1КТ и 2КТ	Максимальный балл за период между 2КТ и на конец семестра	Всего за семестр
Посещение занятий	2	2	2	**6**
Контрольные работы на практических занятиях	4	4	4	**12**
Выполнение и защита результатов лабораторных работ	8	17	18	**43**
Компонент своевременности	3	3	3	**9**
Итого максимум за период:	**17**	**26**	**27**	**70**
Сдача экзамена (максимум)				**30**
Нарастающим итогом	**17**	**43**	**70**	**100**

Оценка по контрольным точкам и дисциплине выставляется на основании рейтинга. Соответствие количества баллов, выраженного в процентах, и оценки представлено в таблице.

Соответствие оценки и рейтинга

Оценка	**Числовой эквивалент**	**Рейтинг (%)**
Отлично	5	91 – 100
Хорошо	4	75 – 90
Удовлетворительно	3	61 – 74
Неудовлетворительно	2	≤ 60

Согласована на портале № 9559

12 Учебно-методические материалы по дисциплине

12.1 Основная литература

1. Гураков А. В., Мещерякова О. И., Мещеряков П. С. Информатика II: учебное пособие. [Электронный ресурс]: – Томск: ФДО, 2015. – 112 с. – URL: https://edu.tusur.ru/training/publications/5546

12.2 Дополнительная литература

1. Фаронов В.В. Турбо Паскаль 7.0: Практика программирования: Учебное пособие - 7-е изд., перераб. - М.: Нолидж, 2001. - 416 с.: ил. (3 экз. в библ.)
2. Вирт Н. Алгоритмы и структуры данных./Пер.с англ. – 2-е изд. – СПб.: Невский Диалект, 2001. -352 с. (1 экз. в библ.)
3. Епанешников, Алексей Михайлович. Программирование в среде Turbo Pascal 7.0: учебное пособие. - М.: Диалог-МИФИ, 1993. - 282, [6] с.: ил. (17 экз. в библ.)
4. Федоров, Алексей. Особенности программирования на Borland Pascal: учебное пособие/ - Киев: Диалектика, 1994. - 144 с. (5 экз. в библ.)

12.3 Учебно-методическое и программное обеспечение

- Гураков А.В., Методические указания к лабораторным работам и самостоятельной работе по дисциплинам «Информатика» и «Информационные технологии» [Электронный ресурс]: – Томск: ФДО, 2015. – 18с. – URL: https://edu.tusur.ru/training/publications/5563
- Интегрированная среда разработки программ Free Pascal.
- Набор презентаций для информационной поддержки курса «Информационные технологии», в формате OpenOffice.org Impress.

13. Материально-техническое обеспечение дисциплины

Для проведения занятий по дисциплине необходим компьютерный класс. Число компьютеров должно соответствовать количеству человек в группе (1 компьютер на 1 студента).

Согласована на портале № 9559

Приложение к рабочей программе

Федеральное государственное бюджетное образовательное учреждение высшего образования

«ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ СИСТЕМ УПРАВЛЕНИЯ И РАДИОЭЛЕКТРОНИКИ» (ТУСУР)

УТВЕРЖДАЮ
Проректор по учебной работе
_____ П. Е. Троян
«___» _____ 2016 г.

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ *ПО УЧЕБНОЙ ДИСЦИПЛИНЕ*

Информационные технологии

Уровень основной образовательной программы **бакалавриат**
Направление(я) подготовки (специальность) **27.03.05 Инноватика**
Профиль(и) **Управление инновациями в электронной технике**
Форма обучения – очная
Факультет Инновационных технологий (ИТ)
Кафедра Управления инновациями
Курс 1 Семестр 2

Учебный план набора 2013 года и последующих лет.

Зачет ___семестр Диф. зачет___ семестр
Экзамен **2** семестр

Томск 2016

Согласована на портале № 9559

1 Введение

Фонд оценочных средств (ФОС) является приложением к рабочей программе дисциплины (практики) и представляет собой совокупность контрольно-измерительных материалов (типовые задачи (задания), контрольные работы, тесты и др.) и методов их использования, предназначенных для измерения уровня достижения студентом установленных результатов обучения.

ФОС по дисциплине (практике) используется при проведении текущего контроля успеваемости и промежуточной аттестации студентов.

Перечень закрепленных за дисциплиной (практикой) компетенций приведен в таблице 1.

Таблица 1 Перечень закрепленных за дисциплиной компетенций

Код	Формулировка компетенции	Этапы формирования компетенции
ОПК-2	способностью использовать инструментальные средства	Знать Уметь. Владеть.

2 Реализация компетенций

2.1 Компетенция ОПК-2

ОПК-2: владеть современными информационными технологиями, быть готовым применять современные средства автоматизированного проектирования и машинной графики при проектировании систем и их отдельных модулей, а также для подготовки конструкторско-технологической документации, соблюдать основные требования информационной безопасности.

Для формирования компетенции необходимо осуществить ряд этапов. Этапы формирования компетенции, применяемые для этого виды занятий и используемые средства оценивания представлены в таблице 2.

Таблица 2 Этапы формирования компетенции и используемые средства оценивания

Состав	Знать	Уметь	Владеть
Содержание этапов	принципы алгоритмизации и программирования; процесс разработки программного обеспечения; принципы построения алгоритмов; основные управляющие структуры и способы описания алгоритмов с использованием различных нотаций;	работать в современных средах разработки, составлять блок-схемы алгоритмов, создавать программы на структурном языке программирования высокого уровня; решать задачи, используя различные методы разработки алгоритмов и выбирая наиболее подходящие	навыками разработки и анализа алгоритмов решения типовых задач (сортировки и поиска данных и пр.), исследования их свойств; методами и инструментальными средствами разработки программ на языке программирования высокого уровня, их тестирования и

Согласована на портале № 9559

	понятие типа данных, форматы представления данных при решении задач с помощью компьютера; синтаксис и семантику алгоритмического языка программирования высокого уровня; основные алгоритмы типовых численных методов решения математических задач; основные алгоритмы сортировки и поиска данных.	алгоритмы и средства их реализации в зависимости от постановки задачи; разрабатывать программы на языке программирования высокого уровня с использованием основных управляющих конструкций и стандартных типов данных; использовать языки и системы программирования для решения профессиональных задач.	отладки; навыками использования инструментальных программных средств в процессе разработки и сопровождения программных продуктов; навыками самостоятельного решения задач с помощью компьютеров, изучения новых средств разработки программ.
Виды занятий	• Лекции.	• Лабораторные работы. • Самостоятельная работа.	• Лабораторные работы. • Самостоятельная работа.
Используемые средства оценивания	• Контрольная работа. • Экзамен.	• Оформление отчетности и защита лабораторных работ. • Конспект самостоятельной работы. • Экзамен	• Защита лабораторных работ. • Контрольная работа. • Экзамен.

Формулировка показателей и критериев оценивания данной компетенции приведена в таблице 3

Таблица 3 Показатели и критерии оценивания компетенции на этапах

Показатели и критерии	Знать	Уметь	Владеть
Отлично (высокий уровень)	принципы алгоритмизации и программирования; процесс разработки программного обеспечения; принципы построения алгоритмов; основные управляющие структуры и способы описания алгоритмов с использованием различных нотаций; понятие типа данных,	работать в современных средах разработки, составлять блок-схемы алгоритмов, создавать программы на структурном языке программирования высокого уровня; решать задачи, используя различные методы разработки алгоритмов и выбирая наиболее подходящие алгоритмы и средства их реализации в	навыками разработки и анализа алгоритмов решения типовых задач (сортировки и поиска данных и пр.), исследования их свойств; методами и инструментальными средствами разработки программ на языке программирования высокого уровня, их тестирования и отладки; навыками использования

Согласована на портале № 9559

	форматы представления данных при решении задач с помощью компьютера; синтаксис и семантику алгоритмического языка программирования высокого уровня; основные алгоритмы типовых численных методов решения математических задач; основные алгоритмы сортировки и поиска данных.	зависимости от постановки задачи; разрабатывать программы на языке программирования высокого уровня с использованием основных управляющих конструкций и стандартных типов данных; использовать языки и системы программирования для решения профессиональных задач.	инструментальных программных средств в процессе разработки и сопровождения программных продуктов; навыками самостоятельного решения задач с помощью компьютеров, изучения новых средств разработки программ.
Хорошо (базовый уровень)	принципы алгоритмизации и программирования; процесс разработки программного обеспечения; принципы построения алгоритмов; основные управляющие структуры и способы описания алгоритмов с использованием различных нотаций; понятие типа данных, форматы представления данных при решении задач с помощью компьютера; синтаксис и семантику алгоритмического языка программирования высокого уровня.	работать в современных средах разработки, составлять блок-схемы алгоритмов, создавать программы на структурном языке программирования высокого уровня; решать задачи, используя различные методы разработки алгоритмов; разрабатывать программы на языке программирования высокого уровня с использованием основных управляющих конструкций и стандартных типов данных.	навыками разработки и анализа алгоритмов решения типовых задач (сортировки и поиска данных и пр.), исследования их свойств; методами и инструментальными средствами разработки программ на языке программирования высокого уровня, их тестирования и отладки.
Удовлетворительно (пороговый уровень)	принципы алгоритмизации и программирования; принципы построения алгоритмов; основные управляющие структуры; понятие типа данных,; синтаксис и семантику алгоритмического языка программирования высокого уровня.	составлять блок-схемы алгоритмов, создавать программы на структурном языке программирования высокого уровня; решать типовые задачи; разрабатывать программы на языке программирования высокого уровня с использованием основных управляющих конструкций и стандартных типов данных.	навыками разработки и анализа алгоритмов решения типовых задач и записи программ на языке программирования высокого уровня.

Согласована на портале № 9559

3 Типовые контрольные задания

Для реализации вышеперечисленных задач обучения используются следующие материалы.

Темы самостоятельной работы:

Понятие алгоритма и алгоритмической системы. Две формы представления алгоритмов: визуальная и текстовая. Свойства алгоритмов. Визуализация алгоритмов и блок-схемы. Линейные, разветвленные и цикличные алгоритмы. Вложенные и параллельные алгоритмы. Логические элементы и базовые управляющие структуры визуального структурного программирования. Построение алгоритма из базовых структур. Пошаговая детализация как метод проектирования алгоритмов.

Программа как изображение алгоритма в терминах команд, управляющих работой компьютера. Структура программы. Трансляция и компоновка. Исходный и объектный модули, исполняемая программа. Компиляция и интерпретация.

Данные как объект обработки. Типы данных, способы и механизмы управления данными. Элементы и структуры данных, алфавит, идентификаторы, выражения, операции, операторы, реализация логических структур на примере языка FreePascal.

Операции с массивами. Ввод и вывод массивов. Алгоритмы поиска и упорядочения массива. Строки. Записи.

Программы и подпрограммы. Подпрограммы, их назначение и классификация. Оформление подпрограмм, обращение к ним, передача параметров. Стандартные и пользовательские модули.

Файловые типы данных. Файловые переменные. Основные операции с файлами. Примеры работы с файлами. Текстовые файлы.

Контрольные работы

Контрольные работы студенты выполняют в электронном виде на сайте mooc.tusur.ru с помощью встроенного интерпретатора. Темы и содержание контрольных работ:

Структура программы. Целочисленные и вещественные типы.

Задание 1.

```
{
Программа находит площадь прямоугольника со сторонами a и b. Укажите тип для
переменной S и запишите оператор присваивания, в котором переменной S будет
присвоен результат выражения a*b.
}
Program Area_of_a_rectangle;
Var
        a : real;
        b : real;
{Опишите переменную S}
```

Согласована на портале № 9559

```
Begin
        Write ('Длина стороны a: (м): ');
        Readln(a);
        Write('Длина стороны b: (м): ');
        Readln(b);
        {Вставьте здесь оператор присваивания}


        Write('Площадь прямоугольника S=',S, ' кв. м.');
end.
```

Задание 2.

```
{
Программа находит площадь треугольника со сторонами a, b и c. Опишите все
переменные и запишите два оператора присваивания, в которых переменной S будет
рассчитанное значение площади, а p - значение половины периметра
S=SQRT(p(p−a)(p−b)(p−c))
p=(a+b+c)/2
}
Program Area_of_a_triangle;
Var
{Опишите переменные}
        a :         ;       {}
        b :         ;       {Стороны треугольника}
        c :         ;       {}
        p :         ;       {Полупериметр треугольника}
        S :         ;       {Площадь}
Begin
        Write ('Длина стороны a: (м): ');
        Readln(a);
        Write('Длина стороны b: (м): ');
        Readln(b);
        Write('Длина стороны c: (м): ');
        Readln(c);
        {Вставьте здесь оператор присваивания}


                Write('Площадь прямоугольника S=',S, ' кв. м.');

        end.
```

Задание 3.

```
{
Написать программу пересчета расстояния из верст в километры (1 верста — это
1066,8 м). Величина расстояния в верстах задается с клавиатуры.
}
Program Versta;
Var
{Опишите переменные}

Begin
                Write ('Введите расстояние в верстах: ');
        {В скобках укажите имя Вашей переменной}
                Readln(     );
```

Согласована на портале № 9559

{*Напишите код программы*}

end.

Ввод и вывод данных.

Задание 1.

{

Выведите на экран текст стихотворения соблюдая формат с использованием только одной процедуры Write или WriteLn

Вы ушли,
как говорится,
в мир в иной.
Пустота…
Летите,
в звезды врезываясь.

}

```
Program kr_10_01;
Uses CRT;
Begin
    ClrScr;
    {Здесь запишите процедуру Write}



End.
```

Задание 2.

{

Составить программу по обмену значениями двух целочисленных переменных без использования третьей.

Результат вывести на экран в виде

До : x=250 y=135
После : x=135 y=250

}

```
Program kr_11_02;
Uses CRT;
Var
    x, y : word;
Begin
    ClrScr;
    x:=250;
    y:=135;
    {Здесь напишите код программы}



End.
```

Задание 3.

{

Дано трехзначное число. Найти число, которое получится при перестановке третьей и второй цифр (десятков и единиц).

Согласована на портале № 9559

Результат вывести на экран в виде

До : 250

После : 205

**число может быть любое*

}

```
Program kr_11_03;
Uses Crt;
Var
    x : integer;
Begin
    Randomize; ClrScr;
    x:=Random(900)+100;    //генерируем случайные целые числа от 100 до 999
    {Напишите код программы ниже}



End.
```

Задание 4.

{

Вывести на экран картинку из специальных символов

['']['_'-]['']

}

```
Program kr_11_04;
Uses crt;
begin
    clrscr;
    //Вставить процедуру вывода здесь


    readkey

end.
```

Организация ветвлений

Задание 1.

{

Написать программу, которая проверяет, делится ли на три введенное с клавиатуры целое число. Ниже приведен рекомендуемый вид экрана программы во время ее работы.

#1

Введите целое число и нажмите <Enter>

-> 451

Число 451 нацело на три не делится.

#2

Введите целое число и нажмите <Enter>

-> 453

Число 453 делится на три нацело.

}

```
Program kr_11_01;
Var
```

Согласована на портале № 9559

```
Begin



End.
```

Задание 2.

{
Написать программу вычисления стоимости покупки с учетом скидки. Скидка в 10% процентов предоставляется, если сумма покупки больше 1000 рублей. Ниже представлен рекомендуемый вид экрана.
#1
Вычисление стоимости покупки с учетом скидки.
Введите сумму покупки и нажмите <Enter>
-> 1200
Вам предоставляется скидка в 10%.
Сумма покупки с учетом скидки: 1080.0 руб.
}

```
Program kr_11_02;
Uses Crt;
Var


Begin
    ClrScr;


End.
```

Задание 3.

{
Дано четырехзначное натуральное число n (n<=9999). Выяснить, является ли оно палиндромом ("перевертышем"). Палиндромом называется число, десятичная запись которого читается одинаково слева направо и справа налево).
Ниже представлен рекомендуемый вид экрана.
#1
Введите четырехзначное натуральное число -> 1234
Число 1234 не является палиндромом
#2
Введите четырехзначное натуральное число -> 1221
Число 1221 является палиндромом
}

```
Program kr_11_03;
Uses Crt;
Var


Begin
```

Согласована на портале № 9559

```
ClrScr;


End.
```

Задание 4.

{

В подъезде жилого дома имеется n квартир, пронумерованных подряд, начиная с номера a. Определить, является ли сумма номеров всех квартир четным числом. Формулу суммы членов арифметической прогрессии использовать нельзя.

Ниже представлен рекомендуемый вид экрана.

#1

Введите номер первой квартиры -> 16

Введите количество квартир -> 15

Сумма номеров всех квартир число нечетное.

#2

Введите номер первой квартиры -> 21

Введите количество квартир -> 20

Сумма номеров всех квартир число четное.

}

```
Program kr_11_03;
Uses Crt;
Var


Begin
    ClrScr;



End.
```

Задание 5.

{

Мастям игральных карт условно присвоены следующие порядковые номера: "пики" - 1, "трефы" - 2 , "бубны" - 3, "червы" - 4. Достоинству карт присвоены следующие порядковые номера: "валету" - 11, "даме" - 12, "королю" - 13, "тузу" - 14. Порядковые номера остальных карт соответствуют их названиям("шестерка", "девятка" и т. п.). По заданному номеру масти m (1<=m<=4) и номеру достоинства карты k (2<=k<=14) определить полное название (масть и достоинство) соответствующей карты в виде: "Дама пик", "Шестерка бубен", "Двойка треф", "Туз червей" и т.п. Оператор If использовать запрещается.

Ниже представлен рекомендуемый вид экрана.

#1

Введите номер масти-> 1

Введите номер достоинства карты -> 14

Туз пик

#2

Введите номер масти-> 4

Введите номер достоинства карты -> 10

Согласована на портале № 9559

Десятка червей
}

```
Program kr_11_03;
Uses Crt;
Var


Begin
    ClrScr;



End.
```

Операторы цикла

Задание 1.

{
Составить программу возведения натурального числа в квадрат,
учитывая следующую закономерность:
1^2=1
2^2=1+3
3^2=1+3+5
4^2=1+3+5+7
...
n^2=1+3+5+7+9+...+(2n-1)
Запрещается использовать операцию умножения и функции возведения в степень, экспоненты и логарифма.
Ниже приведен рекомендуемый вид экрана программы во время ее работы.

#1
Введите натуральное число и нажмите клавишу Enter
-> 4
Квадрат числа 4 равен 16.
}

```
Program prog_kr_12_01;
Uses CRT;

Begin
  ClrScr;

End.
```

Задание 2.

{
Дана последовательность целых чисел a1, a2, ..., a18, в начале которой записаны несколько равных между собой элементов. Определить количество таких элементов последовательности. Условный оператор не использовать.
Ниже приведен рекомендуемый вид экрана программы во время ее работы.

Согласована на портале № 9559

#1
Введите последовательность из 18 целых чисел через пробел:
2 2 2 2 2 2 2 2 2 4 5 6 7 8 3 8 5
В последовательности 10 начальных одинаковых элементов(а)
}

```
Program prog_kr_12_02;
Uses CRT;



Begin
  ClrScr;

End.
```

Задание 3.

{
В некоторой стране используют денежные купюры достоинством ы 1, 2, 4, 16, 32, 64. Дано натуральное число n. Как наименьшим количеством таких денежных купюр можно выплатить сумму n (указать количество каждой из использованных для выплаты купюр)? Предполагается, что денег хватит.
}

```
Program prog_kr__12_02;
Uses CRT;

Begin
   Clrscr;


End.
```

Задание 4.

{
Найти 100 первых простых чисел.
}

```
Program prog_kr_12_04;
Uses crt;



Begin
  CLrScr;


end.
```

Подпрограммы

Задание 1.

Согласована на портале № 9559

{
Опишите функцию CircleS(R:real), находящую площадь круга радиуса R. С помощью этой функции найти площадь кольца, для которого задан внешний и внутренний радиусы. Результат вывести в формате с фиксированной точкой, ограничив дробную часть двумя знаками. Рекомендуемый интерфейс дан ниже.
1#
Введите значение внешнего радиуса кольца (м): ***5.4***
Введите значение внутреннего радиуса кольца (м): ***1.2***
Площадь кольца равна 87.08 кв. м.

Задание 2.

{
Имеется n бактерий красного цвета. Через 1 такт времени красная бактерия меняется на зелёную, затем через 1 такт времени делится на красную и зелёную. Сколько будет всех бактерий через k тактов времени?

Описать и использовать функцию Fib(N), вычисляющую N-й элемент последовательности чисел Фибоначчи.
Рекомендуемый интерфейс:
1#
Введите число красных бактерий: **10**
Введите количество тактов: **9**
Общее число бактерий через 9 тактов равно 550

!Внимание! Запрещается использовать циклы, рекурсию и операторы выбора. Оператор присваивания допустим только один.
}

Задание 3.

{
Описать процедуру Minmax(X, Y), записывающую в переменную X минимальное из значений X и Y, а в переменную Y — максимальное из этих значений.
}

```
Program Prog_kr_13_03;
Uses CRT;



Var
        a, b: real;
Begin
        ClrScr;
        a:=3; b:=5.5;
        MinMax(a,b);
        Writeln(a:1:1, ' меньше ', b:1:1);

        a:=7.8; b:=5.5;
        MinMax(a,b);
        Writeln(a:1:1, ' меньше ', b:1:1);
End.
```

Задание 4.

Согласована на портале № 9559

{
Описать процедуру Minmax(X, Y), записывающую в переменную X минимальное из значений X и Y, а в переменную Y - максимальное из этих значений.
Оба параметра вещественного типа.
Используя четыре вызова этой процедуры, найти минимальное и максимальное из данных чисел A, B, C, D.
В основном блоке программы оператор присваивания не используется.

Рекомендуемый интерфейс:
1#
Введите 4 вещественных числа через пробел: ***2.3 4.5 1.1 8***
1.10 - минимальное из четырех чисел
8.00 - максимальное из четырех чисел
}

Лабораторные работы

В течение семестра студенты должны выполнить шесть лабораторных работ. Студенту необходимо выполнить задание, специальным образом оформить отчет и выложить его на проверку.

1. Запись линейных и циклических алгоритмов на языке высокого уровня.
2. Работа со строками.
3. Создание программ с использованием процедур и функций. Модули.
4. Создание программ с использованием структурных типов данных (строки, массивы, записи).
5. Сортировка одномерных массивов.
6. Обработка одномерных и двумерных массивов.

Экзаменационные вопросы.

Экзамен проводится в электронном виде. Примерный перечень вопросов и заданий.
1 Какие из ниже приведённых символов **не** входят в алфавит языка Free Pascal?

&!?\”23

Примечание: *в качестве ответа ввести строку символов без пробелов в порядке возрастания.*

2 Какое число будет выведено на экран в результате работы следующей программы?

```
Program PrIF_01;
Var
        a, b, c:real;
Begin
        a:=12;
        b:=13;
        c:=15;
        If a>b Then b:=a;
        If c>b Then Writeln(c)
                Else Writeln(b)
End.
```

Согласована на портале № 9559

3 Чему будет равно значение переменной s после выполнения следующей последовательности операторов?

```
s:=0;
for i:=0 to 12 do s:=s+2*i;
```

(В ответ введите число).

4 Чему будет равно значение переменной s после выполнения следующей последовательности операторов?

```
s:=0; i:=0;
while s<10 do
    begin
        s:=s+5*i;
        i:=i+1;
    end;
```

(В ответ введите число)

5 Чему будет равно значение переменной s после выполнения следующей последовательности операторов?

```
s:=0; i:=5;
repeat
        s:=s+4*i;
        i:=i+1;
until s>10;
```

(В ответ введите число)

6 Какое число будет выведено на экран в результате выполнения следующей программы?

```
Var
        a, b, c:integer;
procedure sub;
var
        a, b, c:integer;
begin
        a:=4; b:=7;
end;
begin
        a:=4; b:=4; c:=5;
        sub;
        writeln(a+b);
end.
```

7 Какое число будет выведено на экран в результате выполнения следующей программы?

```
Var
        a, b, c:integer;
procedure sub(var x:integer; var y:integer);
var
        a, b, c:integer;
begin
        a:=8; c:=1;
        x:=8; y:=2;
end;
begin
        a:=2; b:=2; c:=3;
        sub(a, b);
        writeln(a+b+c);
    end.
```

Согласована на портале № 9559

8 Какое обращение к процедуре **S** содержит ошибку?

```
Procedure S (a:byte; b:real; var c:char);
Begin
        c:=Chr(a);
        b:=b+a;
end;
Var
        x:byte;
        y:real;
        z:char;
Begin
...
```

9 Какой функцией (процедурой) следует воспользоваться, чтобы удалить из строки несколько символов?

10 Какое число будет выведено на экран в результате выполнения следующего фрагмента программы (предполагается, что все используемые переменные – целые числа)?

```
a[1]:=0;
for i:=2 to 10 do a[i]:=a[i-1]+3*i;
writeln(a[10]);
```

(В ответ введите число)

11 Какое число будет выведено на экран в результате выполнения следующей программы?

```
Const
        N={x};
Var
        j, i, s:integer;
        a:array [1..n, 1..n] of integer;
begin
    s:=0;
    for i:=1 to n do for j:=1 to n do a[i, j]:=i+j;
    for i:=1 to n do s:=s+a[i, {y}];
    writeln(s);
end.
```

12 В файл byte_12.dan записаны целые числа типа byte. Найдите их сумму.

13 Какое описание массива содержит ошибку?

```
Const
    n=10;
    m=12;
Type
    Figura=(elephant, rook, horse, queen, king, pawn);
```

4 Методические материалы

Для обеспечения процесса обучения и решения задач обучения используются следующие материалы:

Основная литература:

1. Гураков А. В., Мещерякова О. И., Мещеряков П. С. Информатика II: учебное пособие. [Электронный ресурс]: – Томск: ФДО, 2015. – 112 с. – URL: https://edu.tusur.ru/training/publications/5546

Дополнительная литература:

Согласована на портале № 9559

1. Фаронов В.В. Турбо Паскаль 7.0: Практика программирования: Учебное пособие - 7-е изд., перераб. - М.: Нолидж, 2001. - 416 с.: ил. (3 экз. в библ.)
2. Вирт Н. Алгоритмы и структуры данных./Пер.с англ. – 2-е изд. – СПб.: Невский Диалект, 2001. -352 с. (1 экз. в библ.)
3. Епанешников, Алексей Михайлович. Программирование в среде Turbo Pascal 7.0: учебное пособие. - М.: Диалог-МИФИ, 1993. - 282, [6] с.: ил. (17 экз. в библ.)
4. Федоров, Алексей. Особенности программирования на Borland Pascal: учебное пособие/ - Киев: Диалектика, 1994. - 144 с. (5 экз. в библ.)

Учебно-методическое и программное обеспечение:

- Гураков А.В., Методические указания к лабораторным работам и самостоятельной работе по дисциплинам «Информатика» и «Информационные технологии» [Электронный ресурс]: – Томск: ФДО, 2015. – 18с. – URL: https://edu.tusur.ru/training/publications/5563
- Интегрированная среда разработки программ Free Pascal.
- Набор презентаций для информационной поддержки курса «Информационные технологии», в формате OpenOffice.org Impress.

Согласована на портале № 9559