

Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования

«Курганский государственный университет»
(КГУ)

Кафедра «Безопасность информационных и автоматизированных систем»

УТВЕРЖДАЮ

Первый проректор



РАБОЧАЯ ПРОГРАММА УЧЕБНОЙ ДИСЦИПЛИНЫ

ОСНОВЫ ПРОГРАММИРОВАНИЯ

(наименование дисциплины)

образовательной программы высшего образования –
программы бакалавриата

44.03.05 – Педагогическое образование (с двумя профилями подготовки)

Направленность (Специализация): Математика и информатика

Формы обучения: заочная

Курган, 2019

Рабочая программа дисциплины «Основы программирования» составлена в соответствии с учебным планом по программе бакалавриата «Педагогическое образование (с двумя профилями подготовки)» (Математика и информатика), утвержденной:
- для заочной формы обучения 29 августа 2019 года.

Рабочая программа дисциплины одобрена на заседании кафедры «Безопасность информационных и автоматизированных систем» 27 сентября 2019 года, протокол № 2.

Рабочую программу составил:

канд. пед. наук, доцент каф. БИАС  /Никифорова Т.А./

Согласовано:

Зав. кафедрой «БИАС»

канд. пед. наук, доцент  /Е.Н. Полякова/

Зав. кафедрой «Методика обучения
естественным наукам и математики»

канд. ф-м. наук, доцент  /С.В. Косовских/

Начальник Управления
образовательной деятельности

 /С.Н. Синицын/

Специалист по учебно-методической
работе учебно-методического отдела

 /Г.В. Казанкова/

1. ОБЪЕМ ДИСЦИПЛИНЫ

Всего: 7 зачетных единиц трудоемкости (252 академических часа)

Заочная форма обучения

Вид учебной работы	На всю дисциплину	Семестр	
		6	7
Аудиторные занятия (контактная работа с преподавателем), (всего часов), в том числе:	16	8	8
лекции,	4	2	2
лабораторные работы,	12	6	6
практические занятия,			
Самостоятельная работа (всего часов), в том числе:	194	100	136
подготовка к зачету	18	18	
подготовка к экзамену	27		27
другие виды самостоятельной работы (самостоятельное изучение тем (разделов) дисциплины)	201	82	109
Вид промежуточной аттестации (зачет, экзамен):	Зач., Экз.	Зачет	Экзамен
Общая трудоемкость дисциплины и трудоемкость по семестрам в часах:	252	108	144

2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ОБРАЗОВАТЕЛЬНОЙ ПРОГРАММЫ

Дисциплина «Основы программирования» относится к вариативной части модуля Б.1 цикла дисциплин.

Краткое содержание дисциплины: Алгоритм. Свойства алгоритма. Способы описания алгоритма. Линейные алгоритмы. Алгоритмы с ветвлением. Циклические алгоритмы. Сложность алгоритма. Структурное (процедурное) программирование. Языки программирования. Синтаксис. Семантика. Трансляторы. Типы трансляторов и их особенности. Этапы компиляции. Алфавит языка. Лексемы. Идентификаторы. Типы данных. Служебные слова. Константы и переменные. Метки. Выражения. Операции. Основные функции. Структура программы. Основные структуры управления программой: ветвления, циклы, и др. Массивы. Алгоритмы обработки массивов. Сортировка массива. Различные алгоритмы сортировки. Строковые типы. Обработка строк. Процедуры и функции. Итерация и рекурсия. Файлы. Принципы работы с файлами. Ссылки. Динамические переменные.

Знания и умения, необходимые студентам для изучения дисциплины, формируются при изучении предшествующих дисциплин «Математическая логика и теория алгоритмов», «Информационные технологии», «Информатика», «Дискретная математика». Знания и умения, приобретенные студентами в результате изучения дисциплины, будут использоваться при изучении курсов «Информационные системы», «Объектно-ориентированное программирование», «Компьютерное моделирование», «Практикум по решению задач на ЭВМ», «Графические средства в языках высокого уровня», «Разработка компьютерных обучающих систем», «Разработка компьютерных систем оценки результатов обучения» и пр. Они могут быть полезны также при изучении других дисциплин, связанных с вопросами разработки, эксплуатации ИС, а также вопросами принятия решений.

Результаты обучения по дисциплине «Основы программирования» необходимы для выполнения разделов курсового проекта по дисциплине «Объектно-ориентированное программирование», а также выпускной квалификационной работы в части проектирования и разработки компьютерных обучающих систем.

Освоение следующих компетенций на уровне не ниже порогового: способен осваивать основы ИКТ-технологий и видеть перспективы направлений их развития (ПК-5).

3. ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ ОБУЧЕНИЯ

Целью изучения дисциплины «Основы программирования» является формирование обще-профессиональных и специальных компетентностей посредством знакомства студентов с базовыми понятиями программирования, с основами разработки алгоритмов, с основами оценки сложности алгоритма. Курс «Основы программирования» преследует несколько целей: дать студентам базовые представления о различных технологиях программирования, о процедурном или структурном программировании; сформировать у студента умение анализировать предметную область; на основе проведенного анализа студент должен уметь выбрать соответствующий язык программирования для решения практических задач в своей учебной и будущей профессиональной деятельности. Задачей изучения дисциплины «Основы программирования» является научить студентов квалифицированно использовать возможности языка программирования для решения задач.

Изучение дисциплины «Основы программирования» в соответствии с ФГОС ВО по направлению «44.03.05 – Педагогическое образование (с двумя профилями подготовки)» (Направленность: Математика и информатика) направлено на формирование следующих компетенций: – способен осваивать основы ИКТ-технологий и видеть перспективы направлений их развития (ПК-5).

В результате изучения дисциплины студенты должны:

Знать:

- историю создания и классификацию языков программирования (ПК-5);
- о технологиях программирования и о методах программирования (ПК-5);
- об этапах решения задач на компьютере с использованием языка программирования (ПК-5);
- базовые алгоритмические структуры: ветвление, циклы (ПК-5);
- основы языка структурного программирования: синтаксис, семантику, прагматику (ПК-5);
- о выборе типа данных и о выборе алгоритмической структуры для реализации на языке программирования конкретной математической модели (ПК-5);
- алгоритмы работы с массивами (ПК-5);
- методы поиска максимального и минимального элемента (ПК-5);
- алгоритмы сортировки массивов: метод сортировки обмeнами (частный случай – метод пузырька), метод последовательных минимумов, сортировка простыми вставками и бинарными вставками, метод быстрой сортировки или сортировки Хоора, алгоритм сортировки слияниями (ПК-5);
- методы перебора (ПК-5);
- рекурсию (ПК-5);
- методологию оценки сложности разработанного алгоритма (ПК-5).

Студент должен обладать умениями:

- построения моделей (дискретных и непрерывных, вероятностных и детерминированных) для различных предметных областей (ПК-5);
- анализа предметной области: выявления потребностей, выявления информационных объектов, выявления связей между объектами (ПК-5);
- на основе проведенного анализа уметь выбрать соответствующее программное обеспечение (в частности, язык программирования) для решения практических задач (ПК-5);
- конструировать программы (ПК-5);
- практической реализации модели, анализа адекватности компьютерной модели реальному объекту (ПК-5).

Владеть:

- 1) навыками проектирования программных продуктов для различных предметных областей (ПК-5);
- 2) навыками работы с компиляторами языка программирования, инструментальными средствами (ПК-5).

4. СОДЕРЖАНИЕ ДИСЦИПЛИНЫ

4.1. Учебно-тематический план

Зачная форма обучения

6 семестр

Ру-беж дисциплины	Номер раздела, темы дисциплины	Наименование раздела, темы дисциплины	Количество контактной работы с преподавателем		
			Лекции	Практические занятия	Лабораторные работы
<i>Модуль 1. Основы программирования на языке Pascal</i>					
Ру-беж 2	P1	Алфавит, синтаксис, семантика языка программирования Pascal. Идентификаторы. Служебные слова. Структура программы на Pascal: раздел заголовка, раздел декларации, операторная часть. Типы данных. Классификация типов данных. Константы и переменные. Метки. Выражения. Операции. Основные функции.	1		
	P2	Основные структуры управления программой.	1		1
	P3	Массивы. Алгоритмы обработки массивов.			1
	P4	Сортировка массива.			1
	P5	Строковые типы. Обработка строк.			1
	P6	Множества. Записи			1
	P7	Процедуры и функции. Итерация и рекурсия.			1
	P8	Файлы: типизированные (компонентные), текстовые и нетипизированные.			0
Всего			2		6

7 семестр

Ру-беж дисциплины	Номер раздела, темы дисциплины	Наименование раздела, темы дисциплины	Количество контактной работы с преподавателем		
			Лекции	Практические занятия	Лабораторные работы
<i>Модуль 2. Основы программирования на языке C++</i>					
Ру-беж 3	P9	Алфавит языка программирования C++. Лексемы. Константы. Выражения. Операции в C++. Приоритет операций. Скалярные типы данных C++. Объявление, описание и инициализация констант и переменных. Операции над скалярными типами данных C++. Структура программы на C++. Стандартная библиотека потокового ввода/вывода. Форматированный ввод/вывод.	1		
	P10	Основные управляющие операторы C++. Условные, циклические, селективные инструкции.	1		1
	P11	Указатели в C++			1
	P12	Массивы в C++. Сортировки массивов			1
	P13	Обработка символьных строк в C++. Стандартные функции обработки строк в библиотеке string			1
	P14	Структурный тип в C++.			1
	P15	Функции C++. Передача параметров и возвращение значения. Параметры структурных типов. Стандартная библиотека функций языка C++.			1
	P16	Файлы. Текстовые файлы. Двоичные файлы.			0
Всего			2		6

4.2. Содержание лекционных занятий

6 семестр

Шифр раздела, темы дисциплины	Наименование раздела, темы дисциплины	Наименование и содержание лекции	Трудоемкость,
-------------------------------	---------------------------------------	----------------------------------	---------------

лшш	Модуль 2. Основы программирования на языке Pascal		часы
р1	Алфавит, синтаксис, семантика языка программирования Pascal. Идентификаторы. Структура программы на Pascal. Типы данных. Классификация типов данных. Константы и переменные. Метки. Выражения. Операции. Основные функции.	Алфавит языка программирования Pascal	11
		Синтаксис, семантика языка программирования.	
		Идентификаторы	
		Зарезервированные слова Pascal.	
		Стандартные директивы.	
		Структура программы на Pascal 7.0: раздел заголовка, раздел декларации, операторная часть.	
		Типы данных. Понятие типа данных.	
		Классификация типов данных языка программирования Pascal.	
		Простые типы данных (порядковые и вещественные). Порядковые типы (целочисленные, логический, символьный, перечислимые, тип-диапазон).	
		Структурные типы (строковые, массивы, множества, записи, файлы).	
		Процедурные типы.	
		Указатели. Объектовые типы. Совместимость типов.	
		Константы и переменные. Идентификаторы. Допустимы значения переменных.	
		Метки. Описание меток. Использование меток. Оператор безусловного перехода.	
		Выражения. Понятие выражения.	
		Операции. Унарные. Мультипликативные. Аддитивные. Арифметические операции. Операции отношения. Логические операции. Приоритет операций.	
		р2	
Операции над символами. Операции и функции для работы с символами			
Основные структуры управления программой. Простой и составной операторы. Пустой оператор. Операторные скобки.			
Условный оператор.			
Оператор варианта.			
Операторы циклов с предусловием, с постусловием, с параметром. Оператор безусловного перехода. Оператор Break, Continue.			
Процедуры и функции. Модули.			
Описание типа «массив», переменной «массив», константы-массив.			
Ограничения, накладываемые на массивы.			
Организация доступа к элементам массива.			
Основные алгоритмы работы с массивами.			

7 семестр

Шифр раздела, темы дисциплины	Наименование раздела, темы дисциплины	Наименование и содержание лекции	Трудоемкость, часы
<i>Модуль 3. Основы программирования на языке C++</i>			
р9	Алфавит языка программирования C++. Лексемы. Константы. Выражения. Операции в C++. Приоритет операций. Скалярные типы данных C++. Объявление, описание и инициализация констант и переменных. Операции над скалярными типами данных C++. Структура программы на	Алфавит языка программирования C++	1
		Лексемы языка: идентификаторы; ключевые (служебные, зарезервированные) слова; константы; знаки операций; разделители (знаки пунктуации).	
		Константы	
		Выражения.	
		Унарные и бинарные операции.	
		Операции сдвига	
		Операции отношения (сравнения)	
		Логические бинарные операции	
		Операции присваивания	
		Операторы выбора компонентов структурированного объекта	
		Операции с компонентами классов	
		Условная операция	

	C++, Стандартная библиотека потокового ввода/вывода. Форматированный ввод/вывод	<p>Операция явного преобразования (приведения) типа</p> <p>Операции new и delete</p> <p>Скалярные типы и выражения</p> <p>char (символьный); short (короткий целый); int (целый); long (длинный целый); float (вещественный); double (вещественный с удвоенной точностью); void (отсутствие значения).</p> <p>Инициализация переменных</p> <p>Способ учета знакового разряда. Служебные слова unsigned (беззнаковый) и signed (знаковый)</p> <p>Явное задание способа размещения значения переменной</p> <p>Структура программы на C++.</p> <p>Основные операторы языка программирования Си++</p> <p>пустой оператор</p> <p>составной оператор</p> <p>Стандартная библиотека потокового ввода/вывода.</p> <p>Форматированный ввод/вывод</p>	
p10	Основные управляющие операторы C++. Условные, циклические, селективные инструкции.	<p>Условные конструкции (if ... else)</p> <p>Переключатель (оператор варианта) (switch).</p> <p>Цикл с известным заранее числом повторений for (<инициализация цикла>; <выражение-условие>; <список выражений>) <тело цикла>;</p> <p>Цикл с неизвестным числом повторений while (<условие или выражение>) <тело цикла>;</p> <p>Цикл с постусловием do <тело цикла> while (<выражение или условие>);</p> <p>Операторы передачи управления: оператор безусловного перехода (goto); оператор выхода из цикла или переключателя (break); оператор перехода к следующей итерации цикла (continue).</p> <p>Оператор возврата из функции (return)</p>	1

4.3. Лабораторные занятия 6 семестр

Шифр темы	Наименование раздела, темы дисциплины	Наименование и содержание практического занятия	Трудоемкость, часы
<i>Модуль 2. Основы программирования на языке Pascal</i>			
p2	Основные структуры управления программой.	<p>Основные структуры управления программой. Простой и составной операторы. Пустой оператор. Операторные скобки.</p> <p>Условный оператор.</p> <p>Оператор варианта.</p> <p>Оператор цикла с параметром</p> <p>Оператор цикла с предусловием</p> <p>Оператор цикла с постусловием.</p> <p>Оператор безусловного перехода. Оператор Break, Continue.</p>	1
p3	Массивы. Алгоритмы обработки массивов.	<p>Определение массива. Одномерный, двумерный и многомерный массивы.</p> <p>Описание типа «массив», переменной «массив», константы-массив.</p> <p>Ограничения, накладываемые на массивы.</p> <p>Организация доступа к элементам массива.</p> <p>Заполнение массива с клавиатуры, заполнение случайными числами, заполнение по формуле.</p> <p>Вывод элементов массива</p> <p>Поиск элемента массива</p> <p>Основные алгоритмы работы с массивами.</p>	1
p4	Сортировка массива.	<p>Сортировка обменов (частный случай – метод пузырька).</p> <p>Сортировка последовательных минимумов</p> <p>Сортировка простыми вставками и бинарными вставками</p> <p>Быстрая сортировки или сортировки Хоора</p> <p>Алгоритм сортировки слияниями</p>	1
p5	Строковые типы. Обработ-	Определение строкового типа.	1

	ка строк.	Описание строкового типа, строковой переменной, строковой константы. Функции преобразования строк Процедуры преобразования строк	
P6	Множества Записи	Определение множества. Описание типа «множество», переменной «множество», константы «множество». Ограничения, накладываемые на множество. Операции с множествами. Определение записи. Описание типа «запись», переменной «запись», константы «запись». Поля записи. Организация доступа к полям записи. Операции с записями. Записи с вариативными полями.	1
P7	Процедуры и функции. Итерация и рекурсия.	Понятия процедуры и функции. Основные отличия процедур и функций. Глобальные и локальные описания типов, переменных, констант, процедур и функций. Передача параметров по значению. Передача параметров по ссылке. Процедуры и функции как параметры. Оператор Exit, Halt.	1

7 семестр

Шифр темы	Наименование раздела, темы дисциплины	Наименование и содержание практического занятия	Трудоемкость, часы
Модуль 3. Основы программирования на языке C++			
P10	Основные управляющие операторы C++. Условные, циклические, селективные конструкции.	Условные конструкции (if ... else) Переключатель (оператор варианта) (switch). Цикл с известным заранее числом повторений for (<инициализация цикла>; <выражение-условие>; <список выражений>) <тело цикла>; Цикл с неизвестным числом повторений while (<условие или выражение>) <тело цикла>; Цикл с постусловием do <тело цикла> while (<выражение или условие>); Операторы передачи управления: оператор безусловного перехода (goto); оператор выхода из цикла или переключателя (break); оператор перехода к следующей итерации цикла (continue). Оператор возврата из функции (return)	1
P11	Указатели в C++	Определение указателя Инициализация указателя операция & Операции над указателями: операция разыменования или доступа по адресу (*); преобразование типов (приведение типов); присваивание; получение (взятие) адреса (&); сложение и вычитание (аддитивные операции); инкремент или автоувеличение (++); декремент или автоуменьшение (--); операции отношений (операции сравнения).	1
P12	Массивы в C++ Сортировки массивов	Определение одномерного массива Многомерный массив Массивы указателей Алгоритмы обработки массива Алгоритмы сортировки массива	1
P13	Обработка символьных строк в C++. Стандартные функции обработки строк в библиотеке string	Стандартные функции обработки строк в библиотеке string	1
P14	Структурный тип в C++.		1
P15	Функции C++. Передача параметров и возвращение значения	Функции Определения, описания и вызовы функций Функции с переменным количеством параметров	1

чения. Параметры структурных типов. Стандартная библиотека функций языка C++.	Рекурсивные функции
	Подставляемые (инлайн-) функции
	Функции и массивы

4.4. Контрольная работа

Контрольная работа не предусмотрена в учебном плане.

5. МЕТОДИЧЕСКИЕ УКАЗАНИЯ ОБУЧАЮЩИМСЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ

Дисциплина «Основы программирования» преподается в течении 3 и 4 семестра, в виде лекционных и лабораторных занятий, на которых происходит объяснение, усвоение, проверка материала; на заключительном этапе обучения программированию рекомендуется подготовка отчетов в виде докладов, сообщений, презентаций с их последующим обсуждением.

На лабораторных занятиях и лекциях рекомендуется использование иллюстративного материала (текстовой, графической и цифровой информации), программ для демонстрации алгоритмов, мультимедийных форм презентаций, также рекомендуется подготовка и проведение мозгового штурма.

В преподавании основ программирования применяются образовательные технологии: метод проблемного изложения материала; самостоятельное ознакомление студентов с источниками информации, использование иллюстративных материалов (видеофильмы, аудиозаписи, компьютерные презентации, готовые программы на языках программирования), демонстрируемых на современном оборудовании, общение в интерактивном режиме, составление алгоритмов в соответствии с этапами решения задач на компьютере, составление ментальных карт.

Самостоятельная работа студента, наряду с практическими аудиторными занятиями в группе выполняется (при непосредственном/опосредованном контроле преподавателя) по учебникам и учебным пособиям, оригинальной современной литературе по профилю.

При прослушивании лекций рекомендуется в конспекте отмечать все важные моменты, на которых заостряет внимание преподаватель, в частности те, которые направлены на качественное выполнение соответствующей лабораторной работы.

Залогом качественного выполнения лабораторных работ является самостоятельная подготовка к ним накануне путем повторения материалов лекций и разбором задач, рассмотренных на лекции. Рекомендуется подготовить вопросы по неясным моментам и обсудить их с преподавателем в начале лабораторной работы или в начале следующей лекции.

Преподавателем запланировано применение на лабораторных занятиях технологии разбора конкретных ситуаций. Поэтому приветствуется групповой метод выполнения лабораторных работ и защиты отчетов, а также взаимооценка и обсуждение результатов выполнения лабораторных работ.

Часть лабораторных работ выполняется с использованием таких программных продуктов, как Microsoft Office Word, Microsoft Visio, Borland Pascal, Borland C++. Рекомендуется повторить навыки использования указанных программных продуктов.

Рекомендации студентам перед тем как приступить к защите решения задачи по программированию (3 и 4 семестр) преподавателю:

1. Прочтите соответствующую разделу теории в лекциях, материалах сайта кафедры ИТ КГУ «Информатика и программирование: шаг за шагом», в рекомендованных преподавателем источниках.
2. Решите задачу в соответствии с основными этапами решения задач на компьютере. Не забудьте провести самотестирование.
3. Запустите программу. Проведите самотестирование.
4. Проведите тестирование Вашей программы под контролем преподавателя.
5. В случае правильной работы программы покажите преподавателю математическую модель, разработанные Вами тесты, расскажите алгоритм решения и ответьте на все вопросы преподавателя.
6. В случае успешной защиты Вам будет выставлена оценка по 10-тибалльной шкале. В случае неудачи вернитесь к пункту 1.
7. В случае успешной защиты приступайте к оформлению папки с решениями (для допуска к экзамену по программированию).
8. Предъявите преподавателю оформленное решение на подпись.

Выполнение самостоятельной работы подразумевает самостоятельное изучение разделов дисциплины, подготовку к лабораторным занятиям, к рубежным контролям, выполнение индивидуальной работы, подготовку к зачету и к экзамену.

Рекомендуемая трудоемкость самостоятельной работы представлена в таблице:

Рекомендуемый режим самостоятельной работы

<i>Виды самостоятельной работы студентов</i>	<i>Трудоемкость, часы</i>
Углубленное изучение разделов, тем дисциплины лекционного курса Динамические структуры данных Стандартные модули DOS и др в Pascal Объектно-ориентированный подход к программированию. Абстракция данных. Классы. Объектно-ориентированный подход к программированию. Инициализация и разрушение объектов. Конструкторы и деструкторы. Объектно-ориентированный подход к программированию. Члены класса: поля и методы. Ограничения доступа к членам класса. Друзья класса. Объектно-ориентированный подход к программированию. Уровни доступа к членам класса. Инкапсуляция. Наследование. Полиморфизм и способы его реализации. Шаблоны. Перегрузка операций. Организация ввода-вывода на языке C++. Потоки ввода-вывода. Шаблоны функций. Шаблоны классов. Библиотека STL. Другие библиотеки контейнерных классов. Контейнеры в C++ Обработка исключительных ситуаций	38
Подготовка к аудиторным занятиям (лабораторные занятия) (по 2 ч на каждое занятие)	6+6
Выполнение домашних, индивидуальных работ и т.д.	27+72
Подготовка к промежуточной аттестации по дисциплине (зачет)	18
Подготовка к промежуточной аттестации по дисциплине (экзамен)	27
Прочие виды	
<i>Итого</i>	<i>194</i>

6. ФОНД ОЦЕНОЧНЫХ СРЕДСТВ К АТТЕСТАЦИИ ПО ДИСЦИПЛИНЕ

6.1. Перечень оценочных средств

1. Индивидуальные задания для самостоятельной работы.
2. Отчеты студентов по лабораторным работам.
3. Банк практических заданий и теоретических вопросов к зачету.
4. Банк вопросов к экзамену.

6.2. Процедура оценивания результатов освоения дисциплины

Экзамен и зачет проводятся в форме беседы по вопросам. Экзаменационный билет состоит из 2 вопросов по теории и 1 практического задания. Ответы на каждый вопрос оцениваются до 10 баллов, выполнение практического задания оценивается до 10 баллов. Время, отводимое студенту на подготовку к ответу на теоретические вопросы экзаменационного билета, составляет 1 астрономический час. Задания практической части билета (написание кода на изучаемом языке программирования) выбираются студентом в соответствии со степенью сложности алгоритма.

Результаты зачета и экзамена заносятся преподавателем в экзаменационную ведомость, которая сдается в организационный отдел института в день экзамена, зачета, а также выставляются в зачетную книжку студента.

6.3. Примеры оценочных средств для зачета и экзамена

Список теоретических вопросов и практических заданий к зачету (6 и 7 семестр).

Теория.

1. Возникновение термина «алгоритм». Наивное определение понятия "алгоритм". Различные подходы к понятию «алгоритм». Свойства алгоритма.

2. *Основные понятия:* программа, алгоритм, языки программирования (высокого уровня и машинно-ориентированные), алфавит, синтаксис, семантика языка программирования.
3. Трансляторы: интерпретаторы и компиляторы. Процесс трансляции: лексический и грамматический анализ, генерация управляющей программы.
4. Инструментальные среды и системы программирования.
5. Жизненный цикл программного обеспечения.
6. Алгоритмы сортировки: сортировка обменов (частный случай – метод пузырька), сортировка последовательными минимумами, сортировка простыми вставками и бинарными вставками, быстрая сортировка или сортировка Хоора, алгоритм сортировки слияниями.
7. методы перебора (бинарное дерево, алгоритмы ветвей и границ),
8. Итерация и рекурсия.
9. Алгоритмы работы с массивами, методы поиска максимального и минимального элемента,
10. Методы решения уравнений, метод Монте-Карло и т.д.
11. Оценка сложности алгоритма.

Практика.

Задание 1. Реализовать следующие алгоритмы сортировки:

- сортировка обменами,
- сортировку Шелла,
- пирамидальную сортировку,
- быструю сортировку с одним рекурсивным вызовом,
- цифровую сортировку целых неотрицательных чисел.

Сравнить трудоемкости алгоритмов для различных длин входных массивов.

Список вопросов к зачету (6 семестр).

Основы программирования на языке Pascal.

1. Основные этапы решения задачи (основные этапы компьютерного моделирования).
2. Язык программирования. Классификация языков программирования. Символ, элементарные конструкции, выражения, операторы алгоритмического языка. Алфавит, синтаксис, семантика языка программирования. Трансляторы: интерпретаторы и компиляторы. Инструментальная среда программирования (редактор, компилятор, отладчик и т.д.).
3. Алфавит языка программирования Turbo Pascal 7.0. Идентификаторы. Примеры. Выражения в Turbo Pascal 7.0. Переменные и константы. Описание переменных, констант (типизированных и нетипизированных), типов. Глобальные и локальные переменные, константы, типы.
4. Структура программы на Turbo Pascal 7.0. Заголовок программы. Раздел описаний (раздел декларации) программы. Раздел программы (основной блок). Примеры.
5. Типы данных. Понятие типа данных. Классификация типов данных.
6. Типы данных. Простые типы данных: порядковые типы. Целочисленные типы данных. Диапазон допустимых значений. Размер в байтах. Процедуры для величин порядкового типа.
7. Типы данных. Простые типы данных: порядковые типы. Логический тип данных. Символьный тип данных. Диапазон допустимых значений. Размер в байтах. Перечисляемый тип данных. Тип-диапазон. Примеры.
8. Типы данных. Простые типы данных: вещественные типы. Диапазон допустимых значений. Количество цифр мантииссы. Размер в байтах.
9. Типы данных. Структурные типы: строковые, массивы, множества, записи, файлы.
10. Типы данных. Процедурные типы. Указатели.
11. Основные операции в Pascal: арифметические, логические, операции отношения. Приоритет операций. Примеры. Операции с битами информации.
12. Основные функции Pascal: арифметические, функции преобразования типа, функции для величин порядкового типа. Генерация псевдослучайных чисел.
13. Пустой оператор. Операторные скобки. Оператор присваивания. Операторы ввода и вывода данных. Форматный вывод данных. Комментарии. Примеры.
14. Условный оператор. Оператор варианта. Примеры.
15. Циклы. Оператор цикла с параметром. Особенности цикла с параметром. Примеры.

16. Циклы. Оператор цикла с предусловием. Особенности цикла с предусловием. Примеры.
17. Циклы. Оператор цикла с постусловием. Особенности цикла с постусловием. Примеры.
18. Циклы. Стандартные процедуры Break и Continue в операторах цикла. Примеры.
19. Массивы. Определение массива. Одномерный, двумерный и многомерный массивы. Описание типа «массив», переменной «массив», константы «массив». Размещение массива в памяти ЭВМ. Ограничения, накладываемые на массивы. Организация доступа к элементам одномерного и двумерного массива. Примеры.
20. Основные алгоритмы работы с массивами. Заполнение одномерного и двумерного массивов. Примеры.
21. Основные алгоритмы работы с массивами. Вывод элементов одномерного и двумерного массивов.
22. Основные алгоритмы работы с массивами. Сортировка методом «пузырька».
23. Основные алгоритмы работы с массивами. Сортировка методом простого выбора.
24. Основные алгоритмы работы с массивами. Сортировка методом Хоора (метод «быстрой сортировки»).
25. Строковый тип. Описание строкового типа. Описание строкового типа, строковой переменной, строковой константы. Размещение строки в памяти ЭВМ. Обращение к элементу строки. Примеры.
26. Строковый тип. Операции над строками: присваивание, конкатенации, сравнения. Примеры.
27. Строковый тип. Процедуры преобразования строк. Примеры.
28. Строковый тип. Функции преобразования строк. Примеры.
29. Множество. Определение множества. Описание типа «множество», переменной «множество», константы «множество». Ограничения, накладываемые на множество. Операции с множествами. Конструктор множества. Примеры. Процедуры Include() и Exclude().
30. Запись. Определение записи. Описание типа-«запись», переменной типа «запись», константы-«запись». Поля записи. Организация доступа к полям записи. Операции с записями. Примеры.
31. Вариативная (или вариантная) запись. Описание типа-«запись», переменной типа вариативная «запись». Поля вариативной записи. Организация доступа к полям вариативной записи. Примеры.
32. Процедуры и функции. Понятия процедуры, функции. Описания процедур и функций. Основные отличия процедур и функций. Глобальные и локальные описания типов, переменных, констант. Примеры.
33. Процедуры и функции. Формальные и фактические параметры. Передача параметров: параметры-значения (передача по значению) и параметры-переменные (передача по ссылке). Примеры.
34. Процедуры и функции. Формальные и фактические параметры. Параметры-константы. Параметры без типа. Массивы и строки открытого типа. Соответствующие директивы компилятору.
35. Процедуры и функции. Параметры-процедуры и параметры-функции. Соответствующие директивы компилятору. Процедуры Exit и Halt().
36. Итерация и рекурсия. Рекурсивные процедуры и функции. Примеры рекурсивных функций для: нахождения наименьшего общего делителя двух чисел по алгоритму Евклида; вычисления an ; вычисления $n!$; перевода натурального числа из десятичной системы счисления в N-ричную (по выбору преподавателя).
37. Файлы в Pascal. Виды файловых типов. Типизированные (компонентные) файлы. Текстовые файлы. Нетипизированные файлы. Особенности каждого вида файлов.
38. Файлы в Pascal. Описание файловой переменной. Общие установочные и завершающие операции с файлами: связывание файловой переменной, инициация файла и т.д. Режим проверки ошибок ввода-вывода. Примеры.
39. Файлы в Pascal. Запись в файл. Чтение из файла. Особенности чтения и записи в типизированные (компонентные), текстовые и нетипизированные файлы. Понятие буфера ввода-вывода. Примеры.
40. Файлы в Pascal. Последовательный и прямой доступ. Позиционирование в файле. Примеры.
41. Файлы в Pascal. Специальные операции: переименование, удаление и т.д. Примеры.
42. Текстовые файлы. Стандартные процедуры и функции для текстовых файлов.
43. Типизированные файлы. Стандартные процедуры и функции для типизированных файлов.
44. Нетипизированные файлы. Стандартные процедуры и функции для нетипизированных файлов.

45. Модуль в Pascal. Структура модуля: заголовок, раздел объявлений, раздел реализации, раздел инициализации. Доступ к объявленным в модуле объектам. Порядок подключения модуля. Компиляция модуля.
46. Система библиотечных модулей. Модуль System, Strings, Dos, Overlay, WinDos, Printer и т.д. Назначение каждого модуля.
47. Система библиотечных модулей. Модуль CRT. Назначение модуля. Работа с экраном: координаты символа, работа со строками, очистка экрана. Настройка цвета. Атрибуты символа. Вывод специальных символов. Позиционирование курсора. Подача звуковых сигналов. Опрос клавиатуры. Таймер. Примеры.
48. Графические возможности Pascal. Модуль Graph. Управление графическими режимами. Система координат и “текущий указатель”. Управление экраном, окнами. Функции и процедуры работы с точками. Рисование графических примитивов. Примеры.
49. Графические возможности Pascal. Модуль Graph. Управление цветом и палитрой. Установка шаблонов заливки. Примеры.
50. Графические возможности Pascal. Модуль Graph. Рисование графических фигур из линий и криволинейных фигур. Вывод и оформление текста в графическом режиме. Примеры.
51. Графические возможности Pascal. Модуль Graph. Активная и видимая страницы. Управление окнами. Создание простейшей анимации: управление страницами видеопамати, процедуры и функция обмена с памятью. Примеры.
52. Распределения памяти для выполняемого кода на Pascal.
53. Динамические переменные. Указатель. Типизированный и нетипизированный указатель. Разыменование указателя. Допустимые операции. Работа с динамической областью памяти с помощью процедур New() и Dispose(); с помощью процедур GetMem() и FreeMem(); с помощью процедур Mark() и Release(). Примеры.

Список вопросов к экзамену (7 семестр).

Основы программирования на языке C++.

1. Классификация языков программирования (компилируемые, исполняемые на виртуальных машинах и интерпретируемые ЯП). Язык программирования C++. Стандарт языка. Стандартная библиотека.
2. Элементы языка C. Константы, идентификаторы, ключевые слова.
3. Базовые типы C++. Переменные, константы и литералы. Приведение типов.
4. Типы данных и их объявление. Целые и вещественные типы. Перечисляемый тип. Хранение целых и вещественных чисел в памяти компьютера.
5. Типы данных и их объявление. Указатели. Операции разадресации и адреса. Адресная арифметика.
6. Выражения. Операнды и операции (унарные, бинарные, тернарные). Правила преобразования типов.
7. Структура простейшей программы на C++. Процесс компиляции.
8. Операторы языка C. Оператор выражение, составной оператор. Оператор присваивания. Арифметические операторы. Операторы сравнения и логические операторы. Порядок действий (приоритет операторов).
9. Организация ввода-вывода в языке C. Форматный ввод-вывод. Стандартные потоки ввода/вывода. Средства работы с потоками ввода/вывода. Специальные символы (символ перевода строки, символ табуляции, символ конца строки, ноль-терминатор).
10. Псевдослучайные числа. Генерация псевдослучайных чисел на C++.
11. Использование библиотечных функций на примере заголовочного файла cmath.
12. Ветвления в программе. Операторы условного перехода: условный оператор и оператор множественного выбора.
13. Организация циклических вычислительных процессов с помощью операторов for, while, do while.
14. Массивы в C++. Индексные выражения. Хранение в памяти одномерных и многомерных массивов.
15. Массивы в C++. Основные алгоритмы их обработки. Ввод и вывод, поиск экстремума.
16. Массивы в C++. Алгоритмы сортировки.
17. C-строки (символьные массивы). Многомерные массивы.

18. Ссылки в C++. Указатели в C++. Операция разыменования. Константные указатели и указатели на константы.
19. Указатели и массивы. Операции над указателями.
20. Статическая, автоматическая и динамическая память. Типичные ошибки, возникающие при использовании динамической памяти («утечка памяти»).
21. Структуры и объединения. Вариантные структуры. Поля битов.
22. Правила определения переменных и типов. Инициализация данных.
23. Функции в C++. Прототип и описание функции.
24. Определение и вызов функций. Формальные и фактические параметры. Передача параметров в функцию по значению, по ссылке, по указателю. Параметры функций со значениями по умолчанию. Перегрузка функций.
25. Определение и вызов функций. Передача массивов и указателей на функции.
26. Определение и вызов функций. Предварительная инициализация параметров, функции с переменным числом параметров. Передача параметров функции main().
27. Время жизни и область видимости программных объектов. Классы памяти. Инициализация глобальных и локальных переменных.
28. Рекурсия. Некоторые подходы к замеру производительности программ.
- 29.
30. Динамические объекты. Способы выделения и освобождения памяти. Линейный односвязный список.
31. Динамические массивы. Особенности выделения и освобождения памяти для многомерных массивов.
32. Директивы препроцессора. Макроопределения.

Пример заданий для индивидуальной работы.

Каждый студент получает индивидуальный вариант для прорешивания задач на языке Pascal, C++. Например,

РАЗДЕЛ 1. Вычисления

Два вкладчика положили в Сбербанк одинаковые суммы. Первый из них взял по истечении m месяцев и получил r рублей, а второй, взяв вклад по истечении n месяцев и получил s рублей. Сколько каждый из них положил в Сбербанк и сколько процентов выплачивает Сбербанк?

РАЗДЕЛ 2. Ветвления

Напишите программу, которая определит, не приведет ли перемножение двух заданных целых чисел к переполнению.

РАЗДЕЛ 3-4. Циклы

3. Найти корни уравнения $\sin \ln x - \cos \ln x + 2 \cdot \ln x = 0$ методом половинного деления на отрезке $[1; 3]$ с точностью $0,001$.

4. При некоторых заданных x , N и ε , определяемых вводом, вычислите: 1) сумму первых n слагаемых указанного ниже вида

$$a_i = \frac{(-1)^i \cdot \cos^{i/2} x \cdot i!}{(i+1)!}, \quad \text{где } i! = \begin{cases} 1 \cdot 3 \cdot 5 \cdot 7 \cdot \dots \cdot i, & \text{если } i - \text{нечетное.} \\ 2 \cdot 4 \cdot 6 \cdot \dots \cdot i, & \text{если } i - \text{четное.} \end{cases}$$

не больше числа ε .

РАЗДЕЛ 5. Строки

16.1. *Шифрация*. Один из методов шифрации называется наложением гаммы. Делается это следующим образом: берется некоторое случайное число в диапазоне от 127 до 255 — гамма, и код каждого символа строки заменяется кодом, получающимся в результате операции: новый код = старый код XOR гамма. Написать программу, реализующую: а) данный метод шифрации; б) дешифрацию строки при заданной гамме.

Входные данные: шифруемая строка.

Выходные данные: * гамма; * зашифрованная строка.

16.2. *Кодировщик*. Написать программу, перекодирующую строку в кодировке KOI в строку в кодировке Windows-1251 и обратно.

РАЗДЕЛ 6. Массивы

16. Массивом `chars [M] [N]` кодируется поле, на котором расположено несколько прямоугольников. Каждый состоит из целого числа клеток, прямоугольники не накладываются друг на друга и не соприкасаются. Разные прямоугольники могут состоять из разных символов. Один и тот же прямоугольник не может состоять из различных символов. Пустые квадраты поля

```

### . . . ? ? . . + .
### . . = . ? ? . . + .
### . . . . . ? . . + .
. . . . ? ? ? . . . .
? ? ? . . . . . = =
? ? ? . . . . . = =
? ? ? . . . . . = =

```

кодируются символом «точка». Подсчитать число прямоугольников разных типов. Пример: Для этого программа должна выдать ответ:

- # - прямоугольников: 2
- ? - прямоугольников: 3
- + - прямоугольников: 1
- = - прямоугольников: 2

РАЗДЕЛ 7. Использование процедур и функций

Общая. В программе использовать механизм формальных параметров.

7.16. Текст закодирован в виде сетки из 0 и 1, где цифрой 0 обозначено отверстие. Для того чтобы декодировать сообщение (криптограмму), нужно наложить сетку на текст так, чтобы в отверстия можно было видеть символы закодированного текста. Первый раз сетка накладывается так, чтобы сторона, отмеченная знаком «+», была верхней, затем сетка поворачивается по часовой стрелке на 90° , читается следующий набор символов и так далее до полного оборота сетки на 360° . Создайте программу а) для ввода сетки; б) для кодирования текста и расшифровки закодированного текста с помощью заданной в файле сетки. Поворот сетки осуществите в процедуре.

Пример	цифры рамки	сообщения
	ж е у н р и в н я е к к о б р с т у р л к т и р а о о с е н ь е т	Пуримглазобироедваиснежынекруги
Сетка	+ 1 0 1 0 1 0 1 1 1 1 0 1 1 1 0 1 1 1 1 0 1 1 0 1 1 1 1 1 1 1 1 1 1 0 1 1	

РАЗДЕЛ 8. Рекурсия

16. Пусть имеется n дорог. Некоторые из них соединены дорогами известной длины. При помощи рекурсии найдите: кратчайший маршрут, начинающийся в заданной городе n и проходящий через остальные.

РАЗДЕЛ 9. Множество (только для Pascal)

16. Имеется список класса (все имена различны). Определить, есть ли в классе человек, который побывал в гостях у всех. (Для каждого ученика составить множество побывавших у него в гостях друзей, сам ученик в это множество не входит.)

РАЗДЕЛ 10. Записи и файлы

Общая: Программа должна предоставить пользователю следующий выбор: 1) создание файла, 2) просмотр записей из файла, 3) добавление записей в файл, 4...n-1) просмотр решения задачи и n) выход из программы. Указанные выше действия оформить в виде подпрограмм.

16. Описать переменную круг, в которой содержатся все данные для построения круга в декартовой системе координат. а) Определить координаты центра, радиус, площадь и длину окружности круга минимального радиуса, который будет содержать внутри себя все заданные круги. б) Рассматривая окружности попарно, определить координаты точек пересечения или точки касания для каждой пары, либо вывести сообщение об отсутствии общих точек.

РАЗДЕЛ 11. Текстовый файл

16. Текстовый файл содержит запись многочленов некоторой степени с одной переменной x , коэффициенты многочлена — целые. Например, многочлена $5x^4 - 3x^3 + 15x^2 - 4$ записан в файле как $5x^4 - 3x^3 + 15x^2 - 4$. Указать степень многочлена, его коэффициенты. Допisać в указанный файл таблицу значений каждого многочлена на данном отрезке $[a, b]$.

РАЗДЕЛ 12. Нетипизированные файлы

16. Из графических примитивов (режим SVGA 640x480x256) создать спрайты (16x16), сохранить спрайты на диске в нетипизированный файл, подключить их из файла к демонстрационной программе перемещения спрайтов.

РАЗДЕЛ 13. Многоразрядные числа. Длинная арифметика

Общая: Разработать две программы выполнения указанных (см. вариант) арифметических операций над многоразрядными числами (количество цифр превышает 25) при следующих способах хранения:

- 1 способ: с использованием массива.
- 2 способ: с использованием строки.

16. Составить программу для вычисления точного значения многочлена $a_n x^n + a_{n-1} x^{n-1} + \dots + a_1 x + a_0$, где a_i и x — целые числа больше 10^{11} .

РАЗДЕЛ 14. Действия с числами

16.1. Даны натуральные числа n и k , $n > 1$. Напечатать k десятичных знаков числа $1/n$. Программа должна использовать только целые переменные.

16.2. Дано натуральное число n . Переставить его цифры так, чтобы образовалось максимальное число, записанное теми же цифрами.

РАЗДЕЛ 15. Графика

16.1. «Графики». Составить программу, которая предлагает пользователю некоторый список функций для построения графиков (например, $y = ax^2 + bx + c$, $y = a \sin x + b$ и т.д. до 10 наименований). После выбора соответ-

вующей функции, задания коэффициентов и отрезка, на котором выполняется построение, программа строит указанный график. Затем значение коэффициентов и положение графика можно менять (например, с помощью клавиш управления курсором), после чего график перестраивается и отображается обновленное уравнение соответствующей кривой.

РАЗДЕЛ 16. Модули.

Общая: Создать модуль, разработать программу для демонстрации работы процедур, функций модуля.

16.1. Создать модуль `DigitsP` для выполнения арифметических операций с использованием прямого, обратного и дополнительного кодов вещественных чисел в 2-ичной системе счисления со следующим набором подпрограмм:

- 1) перевод числа из десятичной системы счисления в 2-ичную;
- 2) перевод числа из 2-ичной системы счисления в десятичную;
- 3) получение прямого, обратного и дополнительного кодов числа;
- 4) функции проверки правильности записи прямого, обратного и дополнительного кодов числа;
- 5) сложение с использованием обратного и дополнительного кодов чисел.

P-ичное число представить следующим типом:

```
Type TNumber = Array[0..31] Of 0..1;
```

Пример заданий для выполнения на практических занятиях (3 семестр и 4 семестр)

Каждая из лабораторных работ сопровождается подробными пошаговыми инструкциями выполнения заданий. В конце текста каждой лабораторной работы приведен список вопросов для самоконтроля и контроля преподавателя. Каждая выполненная лабораторная работа должна быть проверена преподавателем и защищена студентом с учетом высказанных замечаний. По итогам защиты студент получает определенное количество баллов. Допуск к семестровой аттестации по дисциплине «Основы программирования» осуществляется только при выполнении всех лабораторных работ.

Каждый студент получает индивидуальный вариант для прорешивания задач на языке Pascal и C++. На занятиях будут прорешаны общие задания.

6.5. Фонд оценочных средств

Полный банк заданий для текущего, рубежных контролей и промежуточной аттестации по дисциплине, показатели, критерии, шкалы оценивания компетенций, методические материалы, определяющие процедуры оценивания образовательных результатов, приведены в учебно-методическом комплексе дисциплины.

7. ОСНОВНАЯ И ДОПОЛНИТЕЛЬНАЯ УЧЕБНАЯ ЛИТЕРАТУРА

7.1. Основная учебная литература

По программированию

1. Кнут Д. Искусство программирования: Пер. с англ.: В 3 т. – М.: Мир, 1976. Т. 1. Основные алгоритмы. – М.: Мир, 1976. – 736 с. Т. 2. Получисленные алгоритмы. – М.: Мир, 1977. – 728 с. Т. 3. Сортировка и поиск. – М.: Мир, 1978. – 848 с. [Электронный ресурс] <https://bookree.org/reader?file=536014>
2. Вирт Н. Алгоритмы + структуры данных = программа. – М.: Мир, 1985. – 368с. [Электронный ресурс] <https://bookree.org/reader?file=437196>
3. Мейер Б., Бодуэн К. Методы программирования. – М.: Мир, 1982. Т. 1. – 356 с. Т. 2. – 368 с. [Электронный ресурс] <https://bookree.org/reader?file=486551>

для изучения C

4. Подбельский В.В. Язык Си++: Учебное пособие. - М.: Финансы и статистика, 1996. - 560 с. [Электронный ресурс] <https://bookree.org/reader?file=523028>
5. Керниган Б., Ритчи Д., Фьюэр А. Язык программирования Си. Задачи по языку Си. М.: Финансы и статистика, 2009. - 279 с. [Электронный ресурс] <https://bookree.org/reader?file=652860>

7.2. Дополнительная литература

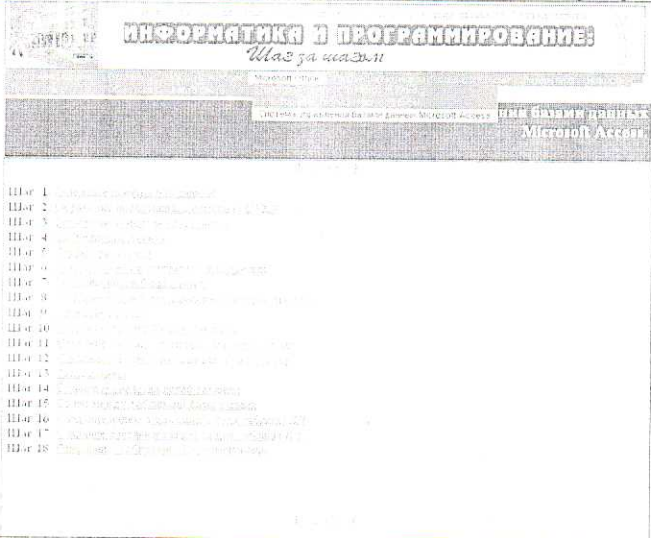
1. Керниган Б., Ритчи Д., Фьюэр А. Язык программирования Си. Задачи по языку Си. М.: Финансы и статистика, 1985. - 279 с.
2. Керниган Б., Ритчи Д. Язык программирования Си. - М.: Финансы и статистика, 1992. - 272 с.
3. Уэйт М., Прата С., Мартин Д. Язык Си. М.: Мир, 1988. - 512 с.
4. Болски М.И. Язык программирования Си. Справочник. М.: Радио и связь, 1988. - 96 с.
5. Берри Р., Микинз Б. Язык Си. Введение для программистов. - М.: Финансы и статистика, 1988. - 191 с.
6. Джехани Н. Программирование на языке Си. - М.: Радио и связь, 1988. - 272 с.
7. Хенкок Л., Кригер М. Введение в программирование на языке Си. - М.: Радио и связь, 1986. - 192 с.
8. Романовская Л.М., Русс Т.В., Свитковский С.Г. Программирование в среде Си для ПЭВМ ЕС. - М.: Финансы и статистика, 1991. - 352 с.

9. Дансмур М., Дейвис Г. Операционная система UNIX и программирование на языке Си. - М.: Радио и связь, 1989. - 192 с.
10. Бочков С.О., Субботин Д.М. Язык программирования Си для персонального компьютера. - М.: Радио и связь, 1990. - 384 с.
11. Хендрикс Д. Компилятор языка Си для микроЭВМ. - М.: Радио и связь, 1989. - 240 с.

8. УЧЕБНО-МЕТОДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ САМОСТОЯТЕЛЬНОЙ РАБОТЫ ОБУЧАЮЩИХСЯ

1. Медведев А.А Основы программирования на Pascal. Курган: Изд-во Курганского ун-та., 2000.– 168 с.
2. Раздел Программирование сайта кафедры ПОАС «Информатика и программирование. Шаг за шагом» [Электронный ресурс] <http://it.kgsu.ru>

9. РЕСУРСЫ СЕТИ «ИНТЕРНЕТ», НЕОБХОДИМЫЕ ДЛЯ ОСВОЕНИЯ ДИСЦИПЛИНЫ

№	Интернет-ресурс	Краткое описание
1	http://it.kgsu.ru/	Сайт кафедры ИТиМПИ «Информатика и программирование: Шаг за шагом»
	 <p>The screenshot shows the website header with the title 'ИНФОРМАТИКА И ПРОГРАММИРОВАНИЕ: Шаг за шагом'. Below the header is a navigation menu with 18 numbered items, likely representing the steps of a course or program.</p>	
2	http://www.intuit.ru/department/pl/cpp/	Электронный курс А. Л. Фридман «Язык программирования C++»

10. ИНФОРМАЦИОННЫЕ СПРАВОЧНЫЕ СИСТЕМЫ

При чтении лекций используются слайдовые презентации.

Минимальные требования к операционной системе и программному обеспечению компьютера, используемого при показе слайдовых презентаций: Windows XP, Microsoft PowerPoint 2007.

Для организации лабораторных занятий используется Borland Pascal, Borland C++.

11. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

Для материально-технического обеспечения дисциплины требуется наличие компьютерной техники с установленным соответствующим программным обеспечением и другого оборудования, поддерживающего проведение презентаций, построение проектной документации, выход в сеть Интернет. Также требуется обеспечение литературой, которую в достаточном объеме может предложить книжный фонд библиотеки Курганского госуниверситета. На кафедре БИАС имеются 4 специализированные аудитории, мультимедийные средства; при чтении лекций будут использоваться наборы слайдов; для тестирования – средства мониторинга и т.д.

Аннотация к рабочей программе дисциплины

«Основы программирования»

образовательной программы высшего образования –
программы бакалавриата

44.03.05 – Педагогическое образование (с двумя профилями подготовки)

Направленность: Математика и информатика

Трудоемкость дисциплины: 7 з.е. (252 академических часа)

Семестр: 6, 7 (заочная форма обучения)

Форма промежуточной аттестации: зачет, экзамен

Содержание дисциплины. Основные разделы.

Алгоритм. Свойства алгоритма. Способы описания алгоритма. Линейные алгоритмы. Алгоритмы с ветвлением. Циклические алгоритмы. Сложность алгоритма.

Структурное (процедурное) программирование. Языки программирования. Синтаксис. Семантика. Трансляторы. Типы трансляторов и их особенности. Этапы компиляции. Алфавит языка. Лексемы. Идентификаторы. Типы данных. Служебные слова. Константы и переменные. Метки. Выражения. Операции. Основные функции. Структура программы. Основные структуры управления программой: ветвления, циклы, и др. Массивы. Алгоритмы обработки массивов. Сортировка массива. Различные алгоритмы сортировки. Строковые типы. Обработка строк. Процедуры и функции. Итерация и рекурсия. Файлы. Принципы работы с файлами. Ссылки. Динамические переменные.