

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

Государственное образовательное учреждение высшего профессионального образования
**Ивановский государственный энергетический университет
имени В.И.Ленина**

УТВЕРЖДАЮ

Декан _____

_____ (Ф.И.О.)

“ ____ ” _____ 201__

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

«ПРОГРАММИРОВАНИЕ»

Направление подготовки 230100 "Информатика и вычислительная техника"

Квалификация (степень) выпускника бакалавр
(бакалавр, магистр)

Профиль подготовки "Высокопроизводительные вычислительные системы на базе больших ЭВМ"

Форма обучения очная
(очная, заочная и др.)

Выпускающая кафедра ВВС

Кафедра-разработчик РПД ПОКС

Семестр	Трудоем- кость з.е./ час.	Лек- ций, час.	Практич. занятий, час.	Лаборат. работ, час.	Курсовое проекти- рование, час	СРС, час	Форма промежуточного (рубежного) контроля (экзамен/зачет)
1	4,5 / 162	34		28		64	экзамен
2	5,5 / 198	34		28	14	86	экзамен
Итого	10 / 360	68		56	14	150	72

Иваново 2011

Рабочая программа дисциплины (РПД) составлена в соответствии с требованиями ФГОС ВПО по направлению подготовки 230100 "Информатика и вычислительная техника" с учетом рекомендаций ПрООП по профилю подготовки "Высокопроизводительные вычислительные системы на базе больших ЭВМ"

Программу составили:
кафедра ПОКС

Алыкова А.Л., доцент _____ *Ф.И.О., ученое звание*

Рецензент(ы):

(для дисциплин общенаучного цикла – выпускающие кафедры, для дисциплин профессионального цикла – представители работодателей)

Программа одобрена на заседании кафедры (УМС):

_____ *Наименование кафедры (УМС)*
(протокол № от _____)

Председатель цикловой методической комиссии по направлению:

_____ *(Ф.И.О., ученое звание, подпись)*

СОДЕРЖАНИЕ

1. Цели освоения дисциплины.
2. Место дисциплины в структуре ООП ВПО.
3. Структура и содержание дисциплины.
4. Формы контроля освоения дисциплины.
5. Учебно-методическое и информационное обеспечение дисциплины.
6. Материально-техническое обеспечение дисциплины.

Приложения

- Приложение 1. Аннотация рабочей программы.
Приложение 2. Технологии и формы преподавания.
Приложение 3. Технологии и формы обучения.
Приложение 4. Оценочные средства и методики их применения.

1. ЦЕЛИ ОСВОЕНИЯ ДИСЦИПЛИНЫ

Целью освоения дисциплины является достижение следующих результатов обучения (РО):

- знания:
 - на уровне представлений: методика проектирования программ из встроенных и абстрактных типов данных и типовых алгоритмических структур, включающая нисходящее проектирование программы, модульное программирование, структурное кодирование;
 - на уровне воспроизведения: принципы построения алгоритма на основе типовых логических структур
 - на уровне понимания: представление данных в памяти компьютера, порядок работы отдельных операторов языка программирования.
- умения:
 - теоретические (см. предыдущий пункт)
 - практические: находить метод решения задачи, разрабатывать эффективный алгоритм и реализовывать его на языке программирования Си, разрабатывать систему тестов и выполнять отладку программы;
- навыки:
 - работа в интегрированной среде программирования, отладка программ с помощью встроенного отладчика.

Перечисленные РО являются основой для формирования следующих компетенций:

общекультурных

- владеет культурой мышления, способен к обобщению, анализу, восприятию информации, постановке цели и выбору путей ее достижения (ОК-1);
 - готов к кооперации с коллегами, работе в коллективе (ОК-3);
 - способен находить организационно-управленческие решения в нестандартных ситуациях и готов нести за них ответственность (ОК-4);
 - осознает социальную значимость своей будущей профессии, обладает высокой мотивацией к выполнению профессиональной деятельности (ОК-8);
 - осознает сущность и значение информации в развитии современного общества; владеет основными методами, способами и средствами получения, хранения, переработки информации (ОК-11);
 - имеет навыки работы с компьютером как средством управления информацией (ОК-12);
- профессиональных

- осваивать методики использования программных средств для решения практических задач (ПК-2);
- разрабатывать интерфейсы "человек - электронно-вычислительная машина" (ПК-3);
- разрабатывать модели компонентов информационных систем, включая модели баз данных (ПК-4);
- разрабатывать компоненты программных комплексов и баз данных, использовать современные инструментальные средства и технологии программирования (ПК-5);
- обосновывать принимаемые проектные решения, осуществлять постановку и выполнять эксперименты по проверке их корректности и эффективности (ПК-6);
- готовить презентации, научно-технические отчеты по результатам выполненной работы, оформлять результаты исследований в виде статей и докладов на научно-технических конференциях (ПК-7).

2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ООП ВПО

Дисциплина "Программирование" относится к циклу профессиональных дисциплин.

Необходимыми условиями для освоения дисциплины являются: знание школьного курса информатики и математики, умение работать с клавиатурой, владение элементарными навыками работы в OS Windows.

В таблице приведены предшествующие и последующие дисциплины, направленные на формирование компетенций, заявленных в разделе «Цели освоения дисциплины»:

№ п/п	Наименование компетенции	Предшествующие дисциплины	Последующие дисциплины (группы дисциплин)
<i>Общекультурные компетенции</i>			
ОК-1	владеет культурой мышления, способен к обобщению, анализу, восприятию информации, постановке цели и выбору путей ее достижения	информатика	Защита информации Параллельное программирование Технологии параллельного программирования Системы искусственного интеллекта Нейрокомпьютерные системы Моделирование технических систем на МВС Моделирование процессов в сплошных средах Многопоточное и распределенное программирование
ОК-3	готов к кооперации с коллегами, работе в коллективе	информатика	Моделирование технических систем на МВС Моделирование процессов в сплошных средах
ОК-4	способен находить организационно-управленческие решения в нестандартных ситуациях и готов нести за них ответственность	информатика	Системы искусственного интеллекта Нейрокомпьютерные системы Защита информации
ОК-8	осознает социальную значимость своей будущей профессии, обладает высокой мотивацией к выполнению профессиональной деятельности	информатика	Моделирование технических систем на МВС Многопоточное и распределенное программирование Параллельное программирование Технологии параллельного программирования
ОК-11	осознает сущность и значение информации в развитии современного общества; владеет основными методами, способами и средствами получения, хранения, переработки информации	информатика	Базы данных

ОК-12	имеет навыки работы с компьютером как средством управления информацией	информатика	Базы данных Защита информации Системы искусственного интеллекта
<i>Профессиональные компетенции</i>			
ПК-2	осваивать методики использования программных средств для решения практических задач	информатика	Моделирование технических систем на МВС Многопоточное и распределенное программирование Параллельное программирование Технологии параллельного программирования Системы искусственного интеллекта Нейрокомпьютерные системы Защита информации
ПК-3	разрабатывать интерфейсы "человек - электронно-вычислительная машина"	информатика	Базы данных
ПК-4	разрабатывать модели компонентов информационных систем, включая модели баз данных	информатика	Базы данных Моделирование технических систем на МВС Моделирование процессов в сплошных средах
ПК-5	разрабатывать компоненты программных комплексов и баз данных, использовать современные инструментальные средства и технологии программирования	информатика	Базы данных Моделирование технических систем на МВС Многопоточное и распределенное программирование Параллельное программирование Технологии параллельного программирования Системы искусственного интеллекта Нейрокомпьютерные системы Защита информации
ПК-6	обосновывать принимаемые проектные решения, осуществлять постановку и выполнять эксперименты по проверке их корректности и эффективности	информатика	Моделирование технических систем на МВС Многопоточное и распределенное программирование Параллельное программирование
ПК-7	готовить презентации, научно-технические отчеты по результатам выполненной работы, оформлять результаты исследований в виде статей и докладов на научно-технических конференциях	информатика	НИР Итоговая государственная аттестация

3. СТРУКТУРА И СОДЕРЖАНИЕ ДИСЦИПЛИНЫ

Общая трудоемкость дисциплины составляет 10 зачетных единиц, 360 часов.

№ модуля образовательной программы	№ раздела	Наименование раздела дисциплины	Виды учебной нагрузки и их трудоемкость, часы					
			Лекции	Практические занятия	Лабораторные работы	Курсовое проектирование	СРС	Всего часов
1	1	Вводная часть	6	-	2	-		8
	2	Основные операторы языка и простые типы данных	12	-	8	-	24	44

2	3	Структурирование программ и структурированные типы данных	24	-	30	-	48	102
3	4	Алгоритмы решения типовых задач	8	-	6	-	12	26
	5	Динамические структуры данных	4	-	6	-	10	20
4	6	Технология разработки структурных программ	8	-	4	14	56	82
	7	Дополнительные возможности	6	-	-	-	-	6
ИТОГО:			68	0	56	14	150	288

3.1. Лекции

№ п/п	Номер раздела дисциплины	Объем, часов	Тема лекции
1	1	2	Введение: – Связь программирования с другими дисциплинами. – Архитектура компьютера. – Языки программирования. – Компиляторы и интерпретаторы.
2	1	2	Разработка программ: – Основные этапы решения задач на ЭВМ. – Структурное программирование.
3	1	2	Понятие об алгоритмах и способах их представления. Стиль программирования.
4	2	4	Написание программы на языке Си: – Структура программы. – Понятие типа данных. Арифметика: – Числовые константы и переменные. – Оператор присваивания. – Арифметическое выражение. – Стандартные арифметические функции. – Система ввода/вывода. Пример простейшей программы.
5	2	4	Программирование условий: – Условный оператор. – Логическое выражение. – Вложенные условные операторы. – Условная операция. – Переключатель SWITCH CASE. Пример разветвляющейся программы.
6	2	4	Циклы: – Классификация. – Циклы с условиями: WHILE и DO- WHILE – Цикл FOR. – Сравнение циклов. – Вложенные циклы. – Операторы goto, break и continue. Примеры циклических программ.

7	3	4	<p>Массивы:</p> <ul style="list-style-type: none"> – Понятие массива. – Одномерные, двумерные, многомерные массивы. – Декларация массивов. – Доступ к элементу массива, ввод/вывод массива. – Пример задачи с использованием массивов.
8	3	4	<p>Адреса и указатели:</p> <ul style="list-style-type: none"> – Размещение данных в памяти компьютера. – Адреса программных объектов и указатели как тип данных. – Типы указателей и операции над ними. – Адресная арифметика. – Динамические массивы. – Пример задачи с использованием многомерного динамического массива
9	3	4	<p>Подпрограммы: определение функций:</p> <ul style="list-style-type: none"> – Проектирование: модульность и структурное кодирование. – Функции, определяемые пользователем, контекстно-независимые функции. – Область видимости идентификаторов. – Механизм обмена данными между подпрограммой и программой. – Использование функций. – Прототип функции. – Пример на использование функций.
10	3	4	<p>Символы и строки:</p> <ul style="list-style-type: none"> – Символьный тип данных, кодовая таблица ASCII. – Функции для работы с символами. – Пример функции преобразования символов. – Строковый тип данных, отличие строки от массива символов. – Функции для работы со строками. – Пример функции удаления подстроки из строки. – Пример программы на обработку символьной информации.
11	3	4	<p>Файлы:</p> <ul style="list-style-type: none"> – Понятие файла и файловой системы. – Поточковые переменные. – Текстовые и бинарные файлы. – Функции для работы с файлами. – Пример программы с использованием файлов.
12	3	4	<p>Структуры и объединения:</p> <ul style="list-style-type: none"> – Структуры: декларация, использование. – Объединения: декларация, использование. – Объединение на основе структур.
13	4	4	<p>Рекурсия:</p> <ul style="list-style-type: none"> – Определение. – Реализация. – Примеры рекурсивных решений.
14	4	4	<p>Комбинаторные алгоритмы:</p> <ul style="list-style-type: none"> – Полный перебор и его сокращение, примеры. – Алгоритмы выборки, примеры рекурсивной и нерекурсивной реализации. – Алгоритмы перестановки, примеры рекурсивной и нерекурсивной реализации.

15	5	4	Динамические структуры данных: – Списки односвязные и двусвязные, пример. – Деревья: построение, обход, удаление вершин, примеры.
16	6	4	Критерии качества программ: – Жизненный цикл программы. – Эффективность: оценка затрат по времени и памяти на примере некоторых алгоритмов сортировки. – Надежность, правильность.
17	6	4	Проектирование диалоговых программ: – Виды диалогов, организация диалога типа "меню". – Работа с экраном в текстовом режиме, использование системы окон, символов псевдографики. – Обработка функциональной клавиатуры. Основы машинной графики: – Видеорежимы. – Инициализация. – Формирование графических изображений.
18	7	6	Функции, дополнительные возможности: – Функции с переменным числом параметров, пример. – Функции с параметрами со значениями по умолчанию. – Перегрузка функций. – Шаблоны функций.
Итого:		68	

3.2. Лабораторные работы

№ п/п	Номер раздела дисциплины	Наименование лабораторной работы	Наименование лаборатории	Трудоемкость, часов
1	1	Освоение среды программирования	Компьютерный класс	2
2	2	Программирование условий	Компьютерный класс	4
3	2	Программирование циклов	Компьютерный класс	4
4	3	Массивы	Компьютерный класс	6
5	3	Динамические массивы	Компьютерный класс	2
6	3	Функции	Компьютерный класс	6
7	3	Обработка символьной информации	Компьютерный класс	6
8	3	Работа с файлами	Компьютерный класс	4
9	3	Структуры	Компьютерный класс	6
10	4	Рекурсия и комбинаторика	Компьютерный класс	6
11	5	Динамические структуры	Компьютерный класс	6
12	6	Сортировки	Компьютерный класс	4
Итого:				56

3.3. Самостоятельная работа студента

Раздел дисциплины	№ п/п	Вид СРС	Трудоемкость, часов
Раздел 2	1	Подготовка к лабораторным работам	18
	2	Оформление отчетов	6
Раздел 3	1	Подготовка к лабораторным работам	32
	2	Оформление отчетов	16
Раздел 4	1	Подготовка к лабораторным работам	8
	2	Оформление отчетов	4

Раздел 5	1	Подготовка к лабораторным работам	6
	2	Оформление отчетов	4
Раздел 6	1	Подготовка к лабораторным работам	6
	2	Оформление отчетов	4
Итого:			104

3.4. Курсовые работы по дисциплине

Цель: Разработка законченного программного продукта, имеющего хороший пользовательский интерфейс и реализующего эффективный алгоритм решения поставленной задачи. Трудоемкость составляет 14 часов аудиторных консультаций и 46 часов самостоятельной работы.

Темы курсовых работ:

1. Разработка программы раскрашивания карты минимальным количеством цветов
2. Разработка программы поиска маршрута в метрополитене
3. Разработка программного комплекса численного решения алгебраических уравнений, заданных в виде полинома.
4. Разработка программного комплекса вычисления определенных интегралов
5. Разработка программы игры в «крестики-нолики»
6. Разработка программы определения оптимального маршрута
7. Разработка программы составления кроссвордов
8. Разработка программы составления расписания занятий
9. Разработка программы игры в слова
10. Разработка программы проектирования системы мостов на архипелаге
11. Разработать программу игры в «пятнадцать»
12. Разработать программу автоматического построения лабиринтов
13. Разработать программу игры «Угадай слово»
14. Разработать программу игры «Морской бой»
15. Разработать программу игры «Алфавит»
16. Разработать программу игры «100 спичек»
17. Разработать программу игры «Белое – черное»
18. Разработать программу игры «Уголки»
19. Разработка программного комплекса численного решения систем линейных алгебраических уравнений
20. Разработка программы построения графика произвольной функции
21. Разработать программу игры «Реверси»
22. Разработать программу игры «Йога»
23. Разработать программу игры «Ним»
24. Разработать программу игры «Балда»
25. Разработать программу игры в шашки
26. Разработать программу игры в нарды
27. Разработать программу игры «Mancala»
28. Разработать программу хранения и обработки данных в соответствии с заданной предметной областью

4. ФОРМЫ КОНТРОЛЯ ОСВОЕНИЯ ДИСЦИПЛИНЫ

Контроль освоения дисциплины производится в соответствии с ПОЛОЖЕНИЕМ о системе РИТМ в ИГЭУ.

Текущий контроль студентов производится в дискретные временные интервалы (в соответствии с приказом ректора о проведении ТК и ПК по системе РИТМ в ИГЭУ) преподавателями, ведущими лабораторные работы по дисциплине в форме выполнения и защиты лабораторных работ. Кроме того, учитывается посещаемость, активность и самостоятельность на занятиях.

Промежуточный контроль по дисциплине проходит в форме экзамена по окончании семестра, включает в себя решение задачи.

Рубежный (итоговый) контроль знаний студентов производится по завершении изучения дисциплины, проходит в форме экзамена.

5. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

а) основная литература:

1. В. В. Подбельский, Язык СИ++. Учебное пособие. – М: Финансы и статистика. – 2001.
2. Т. А. Павловская, Татьяна Александровна. С/С++. Программирование на языке высокого уровня.— СПб.: Питер, 2004.
3. Б. Страуструп, Язык программирования С++. Специальное издание. – СПб.: БИНОМ. – 2001.
4. Д.Хенкеманс, М.Ли, Программирование на С++. СПб.: Символ-Плюс. – 2004.
5. Алыкова А.Л. Основы программирования. Учеб.-метод. пособие/ ГОУВПО "Ивановский государственный энергетический университет им. В.И. Ленина". – Иваново, 2009.
6. Алыкова А.Л. Алгоритмы решения типовых задач. Учеб.-метод. пособие/ ГОУВПО "Ивановский государственный энергетический университет им. В.И. Ленина". – Иваново, 2004.

б) дополнительная литература:

1. Фигурнов В.Э. IBM PC для пользователя. - М.: Финансы и статистика, 1994. - 368 с.
2. Тассел Д.В. Стил, разработка, эффективность, отладка и испытание программ. - М.: Мир, 1985. - 332 с.
3. Кнут Д. Искусство программирования для ЭВМ. - М.: Мир. - Т. 3: Сортировка и поиск, 1978.

в) программное обеспечение: Среда программирования Borland C++.

6. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

1. Лабораторные работы:

лаборатория Б-303, оснащенная 12 компьютерами

2. Прочее:

рабочее место преподавателя, оснащенное компьютером.

АННОТАЦИЯ РАБОЧЕЙ ПРОГРАММЫ ДИСЦИПЛИНЫ «ПРОГРАММИРОВАНИЕ» (Б3.Б.11)

Дисциплина “Программирование” является частью профессионального цикла дисциплин подготовки студентов по направлению 230100 "Информатика и вычислительная техника".

Дисциплина реализуется на факультете информатики и вычислительной техники кафедрой программного обеспечения компьютерных систем.

Дисциплина нацелена на формирование следующих компетенций:

общекультурных

- владеет культурой мышления, способен к обобщению, анализу, восприятию информации, постановке цели и выбору путей ее достижения (ОК-1);
- готов к кооперации с коллегами, работе в коллективе (ОК-3);
- способен находить организационно-управленческие решения в нестандартных ситуациях и готов нести за них ответственность (ОК-4);
- осознает социальную значимость своей будущей профессии, обладает высокой мотивацией к выполнению профессиональной деятельности (ОК-8);
- осознает сущность и значение информации в развитии современного общества; владеет основными методами, способами и средствами получения, хранения, переработки информации (ОК-11);
- имеет навыки работы с компьютером как средством управления информацией (ОК-12);

профессиональных

- осваивать методики использования программных средств для решения практических задач (ПК-2);
- разрабатывать интерфейсы "человек - электронно-вычислительная машина" (ПК-3);
- разрабатывать модели компонентов информационных систем, включая модели баз данных (ПК-4);
- разрабатывать компоненты программных комплексов и баз данных, использовать современные инструментальные средства и технологии программирования (ПК-5);
- обосновывать принимаемые проектные решения, осуществлять постановку и выполнять эксперименты по проверке их корректности и эффективности (ПК-6);
- готовить презентации, научно-технические отчеты по результатам выполненной работы, оформлять результаты исследований в виде статей и докладов на научно-технических конференциях (ПК-7).

Содержание дисциплины охватывает круг вопросов, связанных с разработкой программ в технологии структурного программирования: жизненный цикл программ, принципы структурного программирования, понятие алгоритма и его свойства, структура Си-программы, основные типы и структуры данных, операторы языка Си, стандартный ввод-вывод, функции, передача параметров, указатели и ссылки, адресная арифметика, динамические массивы, символы и строки, файлы, структуры, эффективность программ в координатах время-память, типовые алгоритмы обработки данных, комбинаторные алгоритмы, динамические структуры, способы организации пользовательского интерфейса.

Преподавание дисциплины предусматривает следующие формы организации учебного процесса: лекции, лабораторные работы, самостоятельная работа студента, консультации, курсовое проектирование.

Программой дисциплины предусмотрены следующие виды контроля: текущий и промежуточный контроль успеваемости в форме сдачи лабораторных работ и рубежный (итоговый) контроль в форме экзамена

Общая трудоемкость освоения дисциплины составляет 10 зачетных единиц, 360 часов. Программой дисциплины предусмотрены лекционные (68 часов), лабораторные (56 часов) занятия, курсовое проектирование (14 часов), самостоятельная работа студента (150 часов).

ТЕХНОЛОГИИ И ФОРМЫ ПРЕПОДАВАНИЯ

Рекомендации по организации и технологиям обучения для преподавателя

I. Образовательные технологии

Преподавание дисциплины ведется с применением следующих видов образовательных технологий:

Информационные технологии: использование электронных ресурсов при подготовке к лекциям и лабораторным занятиям.

II. Виды и содержание учебных занятий

Раздел 1. Вводная часть

Теоретические занятия (лекции) - 6 часов.

Лекция 1. Информационная
Структура лекции.

Введение:

- Связь программирования с другими дисциплинами.
- Архитектура компьютера.
- Языки программирования.
- Компиляторы и интерпретаторы

Лекция 2. Информационная
Структура лекции.

Разработка программ:

- Основные этапы решения задач на ЭВМ.
- Структурное программирование.

Лекция 3. Информационная
Структура лекции.

- Понятие об алгоритмах и способах их представления.
- Стил программирования.

Лабораторные работы - 2 часа, 1 работа.

Все лабораторные работы выполняются индивидуально на компьютере в среде программирования Borland C++.

1. Освоение среды программирования

Раздел 2. Основные операторы языка и простые типы данных

Теоретические занятия (лекции) - 12 часов.

Лекция 4. Информационная
Структура лекции.

Написание программы на языке Си:

- Структура программы.
- Понятие типа данных.

Арифметика:

- Числовые константы и переменные.
- Оператор присваивания.
- Арифметическое выражение.
- Стандартные арифметические функции.
- Система ввода/вывода.

Пример простейшей программы.

Лекция 5. Информационная Структура лекции.

Программирование условий:

- Условный оператор.
- Логическое выражение.
- Вложенные условные операторы.
- Условная операция.
- Переключатель SWITCH CASE.

Пример разветвляющейся программы.

Лекция 6. Информационная Структура лекции.

Циклы:

- Классификация.
- Циклы с условиями: WHILE и DO- WHILE
- Цикл FOR.
- Сравнение циклов.
- Вложенные циклы.
- Операторы goto, break и continue.

Примеры циклических программ.

Лабораторные работы - 8 часов, 2 работы.

1. Программирование условий
2. Программирование циклов

Управление самостоятельной работой студента - 24 часов.

Консультации по выполнению лабораторных работ.

Раздел 3. Структурирование программ и структурированные типы данных

Теоретические занятия (лекции) - 24 часов.

Лекция 7. Информационная Структура лекции.

Массивы:

- Понятие массива.
- Одномерные, двумерные, многомерные массивы.
- Декларация массивов.
- Доступ к элементу массива, ввод/вывод массива.
- Пример задачи с использованием массивов.

Лекция 8. Информационная Структура лекции.

Адреса и указатели:

- Размещение данных в памяти компьютера.
- Адреса программных объектов и указатели как тип данных.
- Типы указателей и операции над ними.
- Адресная арифметика.
- Динамические массивы.
- Пример задачи с использованием многомерного динамического массива

Лекция 9. Информационная Структура лекции.

Подпрограммы: определение функций:

- Проектирование: модульность и структурное кодирование.
- Функции, определяемые пользователем, контекстно-независимые функции.
- Область видимости идентификаторов.
- Механизм обмена данными между подпрограммой и программой.
- Использование функций.
- Прототип функции.
- Пример на использование функций.

Лекция 10. Информационная Структура лекции.

Символы и строки:

- Символьный тип данных, кодовая таблица ASCII.
- Функции для работы с символами.
- Пример функции преобразования символов.
- Строковый тип данных, отличие строки от массива символов.
- Функции для работы со строками.
- Пример функции удаления подстроки из строки.
- Пример программы на обработку символьной информации.

Лекция 11. Информационная Структура лекции.

Файлы:

- Понятие файла и файловой системы.
- Поточковые переменные.
- Текстовые и бинарные файлы.
- Функции для работы с файлами.
- Пример программы с использованием файлов.

Лекция 12. Информационная Структура лекции.

Структуры и объединения:

- Структуры: декларация, использование.
- Объединения: декларация, использование.
- Объединение на основе структур.

Лабораторные работы - 30 часов, 6 работ.

1. Массивы
2. Динамические массивы
3. Функции
4. Обработка символьной информации

5, Работа с файлами

6. Структуры

Управление самостоятельной работой студента - 48 часов.

Консультации по выполнению лабораторных работ.

Раздел 4. Алгоритмы решения типовых задач

Теоретические занятия (лекции) - 8 часов.

Лекция 13. Информационная

Структура лекции.

Рекурсия:

- Определение.
- Реализация.
- Примеры рекурсивных решений.

Лекция 14. Информационная

Структура лекции.

Комбинаторные алгоритмы:

- Полный перебор и его сокращение, примеры.
- Алгоритмы выборки, примеры рекурсивной и нерекурсивной реализации.
- Алгоритмы перестановки, примеры рекурсивной и нерекурсивной реализации.

Лабораторные работы - 6 часов, 1 работа.

1. Рекурсия и комбинаторика

Управление самостоятельной работой студента - 12 часов.

Консультации по выполнению лабораторных работ.

Раздел 5. Динамические структуры данных

Теоретические занятия (лекции) - 4 часов.

Лекция 15. Информационная

Структура лекции.

Динамические структуры данных:

- Списки односвязные и двусвязные, пример.
- Деревья: построение, обход, удаление вершин, примеры.

Лабораторные работы - 6 часов, 1 работа.

1. Динамические структуры

Управление самостоятельной работой студента - 10 часов.

Консультации по выполнению лабораторных работ.

Раздел 6. Технология разработки структурных программ

Теоретические занятия (лекции) - 8 часов.

Лекция 16. Информационная

Структура лекции.

Критерии качества программ:

- Жизненный цикл программы.

- Эффективность: оценка затрат по времени и памяти на примере некоторых алгоритмов сортировки.
- Надежность, правильность.

Лекция 17. Информационная Структура лекции.

Проектирование диалоговых программ:

- Виды диалогов, организация диалога типа "меню".
- Работа с экраном в текстовом режиме, использование системы окон, символов псевдографики.
- Обработка функциональной клавиатуры.

Основы машинной графики:

- Видеорежимы.
- Инициализация.
- Формирование графических изображений.

Лабораторные работы - 4 часов, 1 работ.

1. Сортировки

Управление самостоятельной работой студента - 14 часов.

Консультации по выполнению лабораторных работ.

Раздел 7. Дополнительные возможности

Теоретические занятия (лекции) - 6 часов.

Лекция 18. Информационная Структура лекции.

Функции, дополнительные возможности:

- Функции с переменным числом параметров, пример.
- Функции с параметрами со значениями по умолчанию.
- Перегрузка функций.
- Шаблоны функций.

Курсовые работы

Трудоемкость выполнения работы – 60 час.

Цель: Разработка завершеного программного продукта, имеющего хороший пользовательский интерфейс и реализующего эффективный алгоритм решения поставленной задачи.

Темы курсовых проектов:

1. Разработка программы раскрашивания карты минимальным количеством цветов
2. Разработка программы поиска маршрута в метрополитене
3. Разработка программного комплекса численного решения алгебраических уравнений, заданных в виде полинома.
4. Разработка программного комплекса вычисления определенных интегралов
5. Разработка программы игры в «крестики-нолики»
6. Разработка программы определения оптимального маршрута
7. Разработка программы составления кроссвордов
8. Разработка программы составления расписания занятий
9. Разработка программы игры в слова
10. Разработка программы проектирования системы мостов на архипелаге
11. Разработать программу игры в «пятнашки»

12. Разработать программу автоматического построения лабиринтов
13. Разработать программу игры «Угадай слово»
14. Разработать программу игры «Морской бой»
15. Разработать программу игры «Алфавит»
16. Разработать программу игры «100 спичек»
17. Разработать программу игры «Белое – черное»
18. Разработать программу игры «Уголки»
19. Разработка программного комплекса численного решения систем линейных алгебраических уравнений
20. Разработка программы построения графика произвольной функции
21. Разработать программу игры «Реверси»
22. Разработать программу игры «Йога»
23. Разработать программу игры «Ним»
24. Разработать программу игры «Балда»
25. Разработать программу игры в шашки
26. Разработать программу игры в нарды
27. Разработать программу игры «Mancala»
28. Разработать программу хранения и обработки данных в соответствии с заданной предметной областью

Пример задания на курсовой проект (работу).

Тема: Разработка программы игры в «крестики-нолики»

Игровое поле представляет собой квадрат размером 19×19 клеток. Играющие расставляют в свободные клетки крестики или нолики. Выигрывает тот, кто первым составит цепочку длиной из пяти «своих символов» - горизонтальную, вертикальную или диагональную.

Требования к функциональным характеристикам:

- возможность играть двум игрокам между собой;
- возможность играть одному игроку с компьютером;
- случайный выбор игрока, начинающего игру;
- удобный для игрока способ расстановки крестиков или ноликов;
- контроль корректности заполнения клеток поля;
- контроль игровой ситуации и определение победителя.

Требования к оформлению пояснительной записки:

Пояснительная записка должна содержать:

- задание на проектирование;
- краткий анализ задачи;
- описание математической модели;
- методы решения задачи;
- описание и блок-схемы основных алгоритмов;
- общие характеристики входной и выходной информации;
- набор тестовых примеров с результатами их выполнения;
- руководство пользователя;
- прокомментированные тексты программ.

Этапы выполнения:

1. Проектирование программы:

- разработка структуры программы (выделение подзадач);
 - определение структур данных;
 - разработка алгоритмов;
2. Реализация программы на алгоритмическом языке, отладка и тестирование.
 3. Документирование проекта.
- Сроки выполнения устанавливает преподаватель.

ТЕХНОЛОГИИ И ФОРМЫ ОБУЧЕНИЯ

Рекомендации по освоению дисциплины для студента

Трудоемкость освоения дисциплины составляет 360 часов, из них 138 часов аудиторных занятий и 150 часов, отведенных на самостоятельную работу студента.

Рекомендации по распределению учебного времени по видам самостоятельной работы и разделам дисциплины приведены в таблице.

Контроль освоения дисциплины осуществляется в соответствии с ПОЛОЖЕНИЕМ о системе РИТМ в ИГЭУ.

Вид работы	Содержание (перечень вопросов)	Трудоемкость, час.	Рекомендации
Раздел №2. «Основные операторы языка и простые типы данных»			
Подготовка к лабораторной работе №2	Изучение теоретического материала, разработка метода решения, разработка алгоритма	9	Алыкова А.Л. Основы программирования. Учеб.-метод. пособие/ ГОУВПО "Ивановский государственный энергетический университет им. В.И. Ленина". – Иваново, 2009.
Оформление отчета по лабораторной работе №2		3	Алыкова А.Л. Основы программирования. Учеб.-метод. пособие/ ГОУВПО "Ивановский государственный энергетический университет им. В.И. Ленина". – Иваново, 2009.
Подготовка к лабораторной работе №3	Изучение теоретического материала, разработка метода решения, разработка алгоритма	9	Алыкова А.Л. Основы программирования. Учеб.-метод. пособие/ ГОУВПО "Ивановский государственный энергетический университет им. В.И. Ленина". – Иваново, 2009.
Оформление отчета по лабораторной работе №3		3	Алыкова А.Л. Основы программирования. Учеб.-метод. пособие/ ГОУВПО "Ивановский государственный энергетический университет им. В.И. Ленина". – Иваново, 2009.
Итого по разделу		24	
Раздел №3. «Структурирование программ и структурированные типы данных»			
Подготовка к лабораторной работе №4	Изучение теоретического материала, разработка метода решения, разработка алгоритма	6	Алыкова А.Л. Основы программирования. Учеб.-метод. пособие/ ГОУВПО "Ивановский государственный энергетический университет им. В.И. Ленина". – Иваново, 2009.
Оформление отчета по лабораторной работе №4		2	Алыкова А.Л. Основы программирования. Учеб.-метод. пособие/ ГОУВПО "Ивановский государственный

			энергетический университет им. В.И. Ленина". – Иваново, 2009.
Подготовка к лабораторной работе №5	Изучение теоретического материала, разработка метода решения, разработка алгоритма	6	Алыкова А.Л. Основы программирования. Учеб.-метод. пособие/ ГОУВПО "Ивановский государственный энергетический университет им. В.И. Ленина". – Иваново, 2009.
Оформление отчета по лабораторной работе №5		2	Алыкова А.Л. Основы программирования. Учеб.-метод. пособие/ ГОУВПО "Ивановский государственный энергетический университет им. В.И. Ленина". – Иваново, 2009.
Подготовка к лабораторной работе №6	Изучение теоретического материала, разработка метода решения, разработка алгоритма	5	Алыкова А.Л. Основы программирования. Учеб.-метод. пособие/ ГОУВПО "Ивановский государственный энергетический университет им. В.И. Ленина". – Иваново, 2009.
Оформление отчета по лабораторной работе №6		3	Алыкова А.Л. Основы программирования. Учеб.-метод. пособие/ ГОУВПО "Ивановский государственный энергетический университет им. В.И. Ленина". – Иваново, 2009.
Подготовка к лабораторной работе №7	Изучение теоретического материала, разработка метода решения, разработка алгоритма	6	Алыкова А.Л. Основы программирования. Учеб.-метод. пособие/ ГОУВПО "Ивановский государственный энергетический университет им. В.И. Ленина". – Иваново, 2009.
Оформление отчета по лабораторной работе №7		3	Алыкова А.Л. Основы программирования. Учеб.-метод. пособие/ ГОУВПО "Ивановский государственный энергетический университет им. В.И. Ленина". – Иваново, 2009.
Подготовка к лабораторной работе №8	Изучение теоретического материала, разработка метода решения, разработка алгоритма	5	Алыкова А.Л. Основы программирования. Учеб.-метод. пособие/ ГОУВПО "Ивановский государственный энергетический университет им. В.И. Ленина". – Иваново, 2009.
Оформление отчета по лабораторной работе №8		3	Алыкова А.Л. Основы программирования. Учеб.-метод. пособие/ ГОУВПО "Ивановский государственный энергетический университет им. В.И. Ленина". – Иваново, 2009.
Подготовка к лабораторной работе №9	Изучение теоретического материала, разработка метода решения, разработка алгоритма	4	Алыкова А.Л. Основы программирования. Учеб.-метод. пособие/ ГОУВПО "Ивановский государственный энергетический университет им. В.И. Ленина". – Иваново, 2009.
Оформление отчета по		3	Алыкова А.Л. Основы программирования. Учеб.-метод.

лабораторной работе №9			пособие/ ГОУВПО "Ивановский государственный энергетический университет им. В.И. Ленина". – Иваново, 2009.
Итого по разделу		48	
Раздел №4. «Алгоритмы решения типовых задач»			
Подготовка к лабораторной работе №10	Изучение теоретического материала, разработка метода решения, разработка алгоритма	8	Алыкова А.Л. Алгоритмы решения типовых задач. Учеб.-метод. пособие/ ГОУВПО "Ивановский государственный энергетический университет им. В.И. Ленина". – Иваново, 2004.
Оформление отчета по лабораторной работе №10		4	Алыкова А.Л. Алгоритмы решения типовых задач. Учеб.-метод. пособие/ ГОУВПО "Ивановский государственный энергетический университет им. В.И. Ленина". – Иваново, 2004.
Итого по разделу		12	
Раздел №5. «Динамические структуры данных»			
Подготовка к лабораторной работе №11	Изучение теоретического материала, разработка метода решения, разработка алгоритма	6	Алыкова А.Л. Алгоритмы решения типовых задач. Учеб.-метод. пособие/ ГОУВПО "Ивановский государственный энергетический университет им. В.И. Ленина". – Иваново, 2004.
Оформление отчета по лабораторной работе №11		4	Алыкова А.Л. Алгоритмы решения типовых задач. Учеб.-метод. пособие/ ГОУВПО "Ивановский государственный энергетический университет им. В.И. Ленина". – Иваново, 2004.
Итого по разделу		10	
Раздел №6. «Технология разработки структурных программ»			
Подготовка к лабораторной работе №12	Изучение теоретического материала, разработка метода решения, разработка алгоритма	6	Алыкова А.Л. Алгоритмы решения типовых задач. Учеб.-метод. пособие/ ГОУВПО "Ивановский государственный энергетический университет им. В.И. Ленина". – Иваново, 2004.
Оформление отчета по лабораторной работе №12		4	Алыкова А.Л. Алгоритмы решения типовых задач. Учеб.-метод. пособие/ ГОУВПО "Ивановский государственный энергетический университет им. В.И. Ленина". – Иваново, 2004.
Выполнение курсовой работы	1. Проектирование программы: – разработка структуры программы (выделение подзадач); – определение структур данных; – разработка алгоритмов.	16	Алыкова А.Л. Основы программирования. Учеб.-метод. пособие/ ГОУВПО "Ивановский государственный энергетический университет им. В.И. Ленина". – Иваново, 2009.
		3	
		3	
		10	Алыкова А.Л. Алгоритмы решения типовых задач. Учеб.-метод. пособие/ ГОУВПО "Ивановский

	2. Реализация программы на алгоритмическом языке, отладка и тестирование. 3. Документирование проекта.	22 8	государственный энергетический университет им. В.И. Ленина". – Иваново, 2004.
Итого по разделу		56	
ВСЕГО		150	

ОЦЕНОЧНЫЕ СРЕДСТВА И МЕТОДИКИ ИХ ПРИМЕНЕНИЯ

Оценивание уровня учебных достижений студента осуществляется в виде текущего, промежуточного и рубежного (итогового) контроля в соответствии с ПОЛОЖЕНИЕМ о системе РИТМ в ИГЭУ.

Фонды оценочных средств

Фонды оценочных средств, позволяющие оценить РО по данной дисциплине, включают в себя

- комплект задач по разделам 2, 3 дисциплины размещен в Алыкова А.Л. Основы программирования. Учеб.-метод. пособие/ ГОУВПО "Ивановский государственный энергетический университет им. В.И. Ленина". – Иваново, 2009.
- комплект задач по разделам 4, 5 дисциплины размещен в Алыкова А.Л. Алгоритмы решения типовых задач. Учеб.-метод. пособие/ ГОУВПО "Ивановский государственный энергетический университет им. В.И. Ленина". – Иваново, 2004.
- варианты заданий к курсовому проекту, приведены в УМКД.

Критерии оценивания

В соответствии с Положением о системе РИТМ в каждом семестре проводится 2 текущих, 2 промежуточных и один рубежный контроль. Оценивание производится по результатам выполнения лабораторных работ и курсового проекта. Оценке подлежат выбор способа решения, качество и эффективность предложенного алгоритма, качество программной реализации, качество программной документации (отчет по лабораторной работе, пояснительная записка к курсовой работе)

Лабораторные работы

Отдельных требований по допуску к лабораторным работам не предъявляется.

Отчет по лабораторной работе представляется в печатном или рукописном виде, содержание отчета оговорено в пособии: Алыкова А.Л. Основы программирования. Учеб.-метод. пособие/ ГОУВПО "Ивановский государственный энергетический университет им. В.И. Ленина". – Иваново, 2009.

При сдаче лабораторной работы студент должен обосновать выбор способа решения, объяснить алгоритм, выбор способа представления данных и программную реализацию, доказать работоспособность программы.

Основаниями для снижения оценки являются:

- неэффективное решение;
- плохой стиль программирования;
- недружественный интерфейс;
- небрежное выполнение отчета.

Лабораторная работа не может быть принята, если студент не может дать объяснений по решению задачи.

Подготовка и защита курсового проекта

Курсовая работа включает программный продукт и пояснительную записку.

Программный продукт представляется на машинном носителе в виде исходного кода и исполняемого файла. Пояснительная записка должна документировать программный продукт.

Процедура защиты работы включает демонстрацию программного продукта с одновременной устной презентацией, тестирование программного продукта, ответы на вопросы преподавателя и студентов.

Требования, предъявляемые к обучающимся в ходе защиты: студент обосновать выбор способа решения, объяснить алгоритм и программную реализацию, доказать самостоятельность выполнения курсовой работы.

Критерии оценивания:

- соответствие функциональности программного продукта поставленной задаче – 10 баллов;
- выбор способа решения – 20 баллов;
- эффективность решения – 20 баллов;
- качество программного кода – 20 баллов;
- качество пользовательского интерфейса – 20 баллов;
- качество оформления пояснительной записки – 10 баллов.

Студент должен набрать в ходе защиты проекта не менее 60 баллов.