

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное учреждение
высшего профессионального образования

«УФИМСКИЙ ГОСУДАРСТВЕННЫЙ АВИАЦИОННЫЙ
ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

Кафедра вычислительной математики и кибернетики

АННОТАЦИЯ РАБОЧЕЙ ПРОГРАММЫ

УЧЕБНОЙ ДИСЦИПЛИНЫ

«ПРОГРАММИРОВАНИЕ»

Уровень подготовки

бакалавриат

(высшее образование - бакалавриат; высшее образование – специалитет, магистратура)

Направление подготовки (специальность)

09.03.04 Программная инженерия

(код и наименование направления подготовки, специальности)

Направленность подготовки (профиль, специализация)

Разработка программно-информационных систем

(наименование профиля подготовки, специализации)

Квалификация (степень) выпускника

бакалавр

Форма обучения

очная

Уфа 2015

Исполнители:

доцент каф. ВМиК

должность



подпись

Валиахметова Ю. И.

расшифровка подписи

Заведующий кафедрой

ВМиК

наименование кафедры



личная подпись

Юсупова Н. И.

расшифровка подписи

Место дисциплины в структуре образовательной программы

Дисциплина «Программирование» является дисциплиной базовой части.

Рабочая программа составлена в соответствии с требованиями Федерального государственного образовательного стандарта высшего образования по направлению подготовки 09.03.04 «Программная инженерия», утвержденного приказом Министерства образования и науки Российской Федерации от "12" марта 2015 г. № 229.

Целью освоения дисциплины является формирование у студентов представления о программировании как прикладной науке, позволяющей решать на ЭВМ задачи из различных областей человеческой деятельности, овладение языком программирования высокого уровня и средствами создания надежных и эффективных программ.

Задачи: овладение основными навыками алгоритмизации и методами построения программ на языках высокого уровня (C++); научить студентов разрабатывать модели задач для составления программ, разрабатывать и отлаживать программы на C++, проводить тестирование.

Перечень результатов обучения

Процесс изучения дисциплины направлен на формирование элементов следующих компетенций.

Планируемые результаты обучения по дисциплине

№	Формируемые компетенции	Код	Знать	Уметь	Владеть
1	готовность применять основные методы и инструменты разработки программного обеспечения	ПК-1	Основные подходы к написанию алгоритмов и программ; операторы, циклы, синтаксис команд; как выбирать наиболее подходящие структуры данных для решения поставленной задачи, современные методы построения программ, их отладки и тестирования.	Создавать и отлаживать программный код; выбирать наиболее подходящий язык программирования; обнаружить и исправить ошибки в программе; протестировать программу.	Навыками: написания кода, алгоритмизации и формализации условий задач, расшифровки легенд; разработки математических моделей задач; выбора подходящих структур данных; написания алгоритмов решения задач; программирования алгоритмов; отладки и тестирования программ; составления документации на программы.
	владением навыками чтения, понимания и выделения главной идеи прочитанного исходного кода, документации	ПК-21		алгоритмизировать и формализовать задачи; работать в локальных коллективах; обосновывать, отстаивать и нести ответственность за принимаемые решения;	Навыками: написания кода, алгоритмизации и формализации условий задач, расшифровки легенд; разработки математических моделей задач; выбора подходящих

				подбирать типы данных, используемые в языках высокого уровня;	структур данных; написания алгоритмов решения задач; программирования алгоритмов; отладки и тестирования программ; составления документации на программы.
--	--	--	--	---	---

Содержание и структура дисциплины (модуля)

Общая трудоемкость дисциплины составляет 9 зачетных единиц (324 часа).

Трудоемкость дисциплины по видам работ

Вид работы	Трудоемкость, час.	
	1 семестр	2 семестр
Лекции (Л)	20	20
Практические занятия (ПЗ)	12	12
Лабораторные работы (ЛР)	28	28
КСР	4	5
Курсовая проект работа (КР)	-	-
Расчетно - графическая работа (РГР)	-	-
Самостоятельная работа (проработка и повторение лекционного материала и материала учебников и учебных пособий, подготовка к лабораторным и практическим занятиям, коллоквиумам, рубежному контролю и т.д.)	71	79
Подготовка и сдача экзамена	-	36
Подготовка и сдача зачета	9	-
Вид итогового контроля (зачет, экзамен)	зачет	экзамен

Содержание разделов и формы текущего контроля

№	Наименование и содержание раздела	Количество часов						Литература, рекомендуемая студентам*	Виды интерактивных образовательных технологий**
		Аудиторная работа				СРС	Всего		
		Л	ПЗ	ЛР	КСР				
1	Алгоритм и его свойства. Способы записи алгоритма. Программа. Языки программирования. Примеры алгоритмов и программ. Структура программы на языке C++. Примеры. Этапы создания исполняемой программы. Состав языка C++. Константы и переменные C++. Типы данных в C++. Выражения. Знаки операций. Основные операторы C++ (присваивание, составные, выбора, циклов, перехода). Синтаксис, семантика, примеры	4	2	4	1	12	23	Проблемное обучение; контекстное обучение; лекция-визуализация.	
2	Этапы решения задачи. Виды ошибок. Тестирование. Массивы (определение, инициализация, способы перебора). Сортировка массивов (простой обмен, простое включение, простой выбор). Поиск в одномерных массивах (дихотомический и линейный).	4	2	8	1	14	29	Контекстное обучение; лекция-визуализация.	
3	Указатели. Операции с указателями. Примеры Динамические переменные. Операции new и delete. Примеры. Ссылки. Примеры.	4	2	8	1	16	31	Проблемное обучение.	
4	Одномерные массивы и указатели. Примеры. Многомерные массивы и указатели. Примеры. Динамические массивы. Примеры.	4	4	4	1	16	29	Контекстное обучение; лекция-визуализация.	
5	Символьная информация и строки. Функции	4	2	4		13	27		

	для работы со строками (библиотечный файл string.h). Функции ввод-вывода (scanf(), printf(), puts(), gets(), putchar(), getchar()).								
6	<p>Функции в C++. Формальные и фактические параметры. Передача параметров по адресу и по значению. Локальные и глобальные переменные. Примеры.</p> <p>Прототип функции. Библиотечные файлы. Директива препроцессора #include.</p> <p>Передача одномерных массивов в функции. Примеры.</p> <p>Передача многомерных массивов в функции. Примеры.</p> <p>Передача строк в функции. Примеры.</p> <p>Функции с умалчиваемыми параметрами. Примеры.</p> <p>Подставляемые функции. Примеры.</p> <p>Функции с переменным числом параметров. Примеры.</p> <p>Перегрузка функции. Шаблоны функций. Примеры.</p> <p>Указатели на функции. Примеры.</p> <p>Ссылки на функции. Примеры.</p>	8	4	8	2	26	44		Работа в команде
7	<p>Типы данных, определяемые пользователем (переименование типов, перечисление, структуры, объединения). Примеры.</p> <p>Структуры. Определение, инициализация, присваивание структур, доступ к элементам структур, указатели на структуры, битовые поля структур.</p> <p>Динамические структуры данных (однонаправленные и двунаправленные</p>	6	4	8	1	27	46		Работа в команде

	<p>списки).</p> <p>Создание списка, печать, удаление, добавление элементов (на примере однонаправленных и двунаправленных списков).</p> <p>Потоковый ввод-вывод в C++. Открытие и закрытие потока. Стандартные потоки ввода-вывода.</p>								
8	<p>Символьный, строковый, блоковый и форматированный ввод-вывод.</p> <p>Прямой доступ к файлам.</p> <p>Создание бинарных и текстовых файлов, удаление, добавление, корректировка элементов, печать файлов.</p>	6	4	12	2	26	50		Работа в команде

Занятия, проводимые в интерактивной форме, составляют 30% от общего количества аудиторных часов по дисциплине программирование.

Лабораторные работы

№ ЛР	№ раздела	Наименование лабораторных работ	Кол-во часов
1	1	Изучение среды C++. Составление и отладка первых программ на языке C++.	4
2	2	Битовые операции и битовые массивы.	8
3	3	Работа с массивами. Методы сортировки массивов.	8
4	4	Работа со строками. Циклы. Использование указателей и кучи. Тестирование программ.	4
5	5	Задачи на графах. Алгоритм Дейкстры.	4
6	6	Работа с двумерными массивами. Инициализация, развертка, копирование, преобразование двумерных	4
7	6	Графика. Изучение графических функций. Движение объекта в графическом режиме под управлением спец. клавиш.	4
8	7	Длинная арифметика.	8
9	8	Методы решения оптимизационных задач выбора: задача о загрузке рюкзака.	4
10	8	Методы динамического программирования при решении задач оптимизации.	4
11	8	структуры данных- стек, очередь, списки с одной и двумя связями, бинарное поисковое дерево.	4

Практические занятия (семинары)

№ занятия	№ раздела	Тема	Кол-во часов
1	1	Свойства алгоритма. Способы записи алгоритма. Программа. Этапы создания исполняемой программы. Типы данных в C++. Выражения. Знаки операций. Основные операторы C++ (присваивание, составные, выбора, циклов, перехода). Синтаксис, семантика, примеры	2
2	2	Массивы (определение, инициализация, способы перебора). Сортировка массивов (простой обмен, простое включение, простой выбор). Поиск в одномерных массивах (дихотомический и линейный).	2

№ занятия	№ раздела	Тема	Кол-во часов
3	3	Указатели. Операции с указателями. Примеры Динамические переменные. Операции new и delete. Примеры.	2
4	4	Одномерные массивы и указатели. Примеры. Многомерные массивы и указатели. Примеры.	2
5	4	Динамические массивы. Примеры.	2
6	5	Символьная информация и строки. Функции для работы со строками. Функции ввод-вывода	2
7	6	Функции в С++. Формальные и фактические параметры. Передача параметров по адресу и по значению. Локальные и глобальные переменные. Примеры. Прототип функции. Библиотечные файлы. Директива препроцессора #include. Передача одномерных и многомерных массивов в функции. Примеры.	2
8	6	Передача строк в функции. Функции с умалчиваемыми параметрами. Подставляемые функции. Функции с переменным числом параметров. Перегрузка функции. Шаблоны функций. Указатели на функции. Ссылки на функции.	2
9	7	Типы данных, определяемые пользователем. Структуры. Динамические структуры данных (однонаправленные и двунаправленные списки). Определение, инициализация, присваивание структур, доступ к элементам структур, указатели на структуры, битовые поля структур.	2
10	7	Создание списка, печать, удаление, добавление элементов (на примере однонаправленных и двунаправленных списков). Потоковый ввод-вывод в С++. Открытие и закрытие	2
11	8	Символьный, строковый, блоковый и форматированный ввод-вывод. Прямой доступ к файлам.	2
12	8	Создание бинарных и текстовых файлов, удаление, добавление, корректировка элементов, печать файлов.	2

Учебно-методическое и информационное обеспечение дисциплины (модуля)

Основная литература

1. Иванова Г. С. Технология программирования: учебник / Г. С. Иванова .— 3-е изд., стер. — Москва : Кнорус, 2013 .— 336 с.
2. Павловская, Т. А. С/С++. Структурное программирование. Программирование на языке высокого уровня : учебное пособие пособие для вузов : практикум / Т. А. Павловская, Ю. А. Щупак .— СПб. : Питер, 2003 .— 240 с.
3. Балдин К. В. Математическое программирование: учебник / К. В. Балдин, Н. А. Брызгалов, А. В. Рукосуев – Москва: Дашков и К, 2014 – 220 с.
4. Давыдова Н.А. Программирование: учеб. пособие / Н.А.Давыдова - Москва: Бином. Лаборатория знаний, 2012 - 238 с.

5. Окулов С. М. Динамическое программирование / С. М. Окулов, О. А. Пестов – Москва: БИНОМ. Лаборатория знаний, 2014 – 296 с.

Дополнительная литература

1. Давыдова Н.А. Программирование: учеб. пособие / Н.А.Давыдова - Москва: Бинум. Лаборатория знаний, 2012 - 238 с.

2. Задачи по программированию / С. М. Окулов [и др.]; под ред. С. М. Окулова - Москва: БИНОМ, 2014 - 823 с.

3. Кауфман В. Ш. Языки программирования. Концепции и принципы [Электронный ресурс]: [пособие для студентов вузов] / В. Ш. Кауфман - Москва: ДМК ПРЕСС, 2010 - 464 с.

Интернет-ресурсы (электронные учебно-методические издания, лицензионное программное обеспечение)

На сайте библиотеки УГАТУ <http://library.ugatu.ac.ru/> в разделе «Информационные ресурсы», подраздел «Доступ к БД» размещены ссылки на интернет-ресурсы.

Материально-техническое обеспечение дисциплины

Учебные аудитории университета для лекционных, практических занятий и самостоятельной работы, в том числе лекционные аудитории с современными средствами демонстрации 9-103, 9-202, 6-415, 6-416. Учебные лаборатории, обеспечивающие реализацию ОПОП ВО 6-313,6-409,6-413,6-414,6-417,6-419.

Адаптация рабочей программы для лиц с ОВЗ

Адаптированная программа разрабатывается при наличии заявления со стороны обучающегося (родителей, законных представителей) и медицинских показаний (рекомендациями психолого-медико-педагогической комиссии). Для инвалидов адаптированная образовательная программа разрабатывается в соответствии с индивидуальной программой реабилитации.