

Минобрнауки России

**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ  
ВЫСШЕГО ОБРАЗОВАНИЯ  
«ВОРОНЕЖСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»  
(ФГБОУ ВО «ВГУ»)**

**УТВЕРЖДАЮ**



Заведующий кафедрой  
Алгазинов Эдуард Константинович  
Кафедра информационных систем

31.08.2018

**РАБОЧАЯ ПРОГРАММА УЧЕБНОЙ ДИСЦИПЛИНЫ**

Б1.В.ДВ.04.01 Язык программирования Си

**1. Код и наименование направления подготовки/специальности:**

09.03.02 ИНФОРМАЦИОННЫЕ СИСТЕМЫ И ТЕХНОЛОГИИ

**2. Профиль подготовки/специализация:**

Информационные системы в телекоммуникациях, Информационные системы и технологии в управление предприятием, Информационные системы и сетевые технологии, Обработка информации и машинное обучение, Программная инженерия в информационных системах

**3. Квалификация (степень) выпускника:**

Бакалавриат

**4. Форма обучения:**

Очная

**5. Кафедра, отвечающая за реализацию дисциплины:**

Кафедра информационных систем

**6. Составители программы:**

Фертиков Вадим Валериевич, кандидат физ.-мат. наук, доцент

**7. Рекомендована:**

Научно-методическим советом ФКН, протокол НМС №6 от 25.06.2018

**8. Учебный год:**

2018-2019

**Семестр(ы):**

2

**9. Цели и задачи учебной дисциплины:**

формирование профессиональных компетенций будущих бакалавров в области информационных систем и технологий через изучение основ языка программирования Си, практических приемов его применения для решения вычислительных задач и при реализации приложений, работающих со структурированными данными. В процессе освоения учебных материалов студент получит знание синтаксиса; умение применять языковые конструкции для решения практических задач: алгоритмирование, структурирование программного кода и данных; навыки разработки, тестирования и отладки приложений с использованием современных интегрированных средств. По окончании курса студент должен уметь: определять структуры данных при проектировании алгоритмов в процессе решения задач; разбивать решение сложной задачи на последовательность

более простых задач; использовать библиотеки стандартных программ, которые включены в язык программирования.

#### 10. Место учебной дисциплины в структуре ООП:

дисциплина выборного блока вариативной части (Б1.В.ДВ); входные знания в объеме обязательных курсов, предусмотренных учебным планом для изучения в первом семестре: «Введение в программирование», «Теоретические основы информатики». В свою очередь, знание языка программирования Си необходимо студентам для изучения общих профессиональных и специальных дисциплин: «Язык программирования C++», «Параллельные алгоритмы обработки данных», «Языки и системы программирования», «Технологии программирования».

№ п/п	Наименование дисциплин учебного плана, с которым организована взаимосвязь дисциплины рабочей программы	№ № разделов дисциплины рабочей программы, связанных с указанными дисциплинами
1	Язык программирования C++	9-14
2	Параллельные алгоритмы обработки данных	1-8
3	Технологии программирования	11-13

#### 11. Планируемые результаты обучения по дисциплине/модулю (знания, умения, навыки), соотнесенные с планируемыми результатами освоения образовательной программы (компетенциями выпускников):

Код	Название	Знать	Уметь	Владеть
ПК-12	способностью разрабатывать средства реализации информационных технологий (методические, информационные, математические, алгоритмические, технические и программные)	синтаксические конструкции языка программирования и их семантику	применять языковые конструкции для решения практических задач, использовать библиотеки стандартных функций, поставляемых с языком программирования	навыками разработки приложений с использованием современных интегрированных средств

Код	Название	Знать	Уметь	Владеть
ПК-17	способность использовать технологии разработки объектов профессиональной деятельности в областях: машиностроение, приборостроение, техника, образование, медицина, административное управление, юриспруденция, бизнес, предпринимательство, коммерция, менеджмент	общие приемы структурирования программного кода и обрабатываемых данных	определять структуры данных при проектировании алгоритмов, разбивать решение сложной задачи на последовательность более простых задач	навыками тестирования и отладки программных модулей, реализованных на языке Си

## 12. Объем дисциплины в зачетных единицах/час:

3/108

## Форма промежуточной аттестации:

Зачет

## 13. Виды учебной работы

Вид учебной работы	Семестр 2	Всего
Аудиторные занятия	32	32
Лекционные занятия	16	16
Практические занятия		0
Лабораторные занятия	16	16
Самостоятельная работа	76	76
Курсовая работа		0
Промежуточная аттестация	0	0
Часы на контроль		0
Всего	72	72

### 13.1. Содержание дисциплины

п/п	Наименование раздела дисциплины	Содержание раздела дисциплины
1	Особенности построения программных проектов на языке Си. Представление данных.	Составные части программных проектов - модули. Этапы сборки проекта: препроцессорная обработка, компиляция, объектная компоновка. Лексические основы, константы, описание переменных. Базовые типы данных.

п/п	Наименование раздела дисциплины	Содержание раздела дисциплины
2	Основные встроенные операции языка. Выражения	Операции: унарные, бинарные, условная операция, операция "запятая". Соглашения о приоритетах и ассоциативности операций, примеры выражений.
3	Последовательные операторы	Правила написания последовательных операторов, область действия идентификаторов
4	Операторы, управляющие ходом выполнения программы	Условный оператор и переключатель; операторы циклов: с предусловием, с постусловием, итерационный; операторы передачи управления
5	Тип данных указатель	Указатели, операции размещения и удаления динамических переменных
6	Массивы и указатели	Представление массивов. Массивы и операции над указателями. Размещение динамических массивов. Полезные приемы управления многомерными динамическими массивами.
7	Функции, описание и вызов	Описание функции, использование параметров различных типов: значение, указатель, указатель на массив, указатель на функцию. Полезные приемы передачи динамических матриц в качестве параметров функции.
8	Рекурсия	Примеры использования прямой и косвенной рекурсии.
9	Структуры	Описание структуры, обращение к элементам, возможности обмена между функциями, массив структур - простейшая модель базы данных.
10	Объединения	Примеры использования: несколько способов доступа к памяти; объединение как элемент структуры (структура с вариантами).
11	Простейшие динамические структуры данных	Пример построения динамической структуры данных: односвязный список. Операции вставки, удаления и просмотра элементов списка.
12	Обзор динамических структур и способов их построения	Полезные приемы и примеры построения динамических структур с применением комбинаций типов структур, объединений и указателей: очередь, кольцо, двусвязные список и кольцо, дерево.
13	Двоичное дерево поиска	Пример построения и использования динамической структуры данных: двоичное дерево поиска. Операции вставки, удаления и поиска элементов дерева.

п/п	Наименование раздела дисциплины	Содержание раздела дисциплины
14	Способы определения имен типов	Директива определения имен типов. Структурный тип и тип объединений. Прототип функции - описание функционального типа. Перечислимый тип.
15	Особенности применения данных функционального типа. Управление распределением статической памяти.	Особенности применения динамических структур для хранения данных функционального типа (указателей на функции). Полезный пример: построение контекстно-зависимого меню. Представление констант и изменяющихся переменных (volatile); спецификаторы класса памяти; битовые поля структур.
16	Функции с переменным количеством параметров. Препроцессорные средства.	Цели использования конструкции, приемы программирования и полезный пример. Этапы препроцессорной обработки; способы построения макросов; использование директивы включения текстов и правила оформления заголовочных файлов; условная компиляция и др.

### 13.2. Темы (разделы) дисциплины и виды занятий

№ п/п	Наименование темы (раздела) дисциплины	Лекционные занятия	Практические занятия	Лабораторные занятия	Самостоятельная работа	Всего
1	<i>Особенности построения программных проектов на языке Си. Представление данных.</i>	1		1	4	6
2	<i>Основные встроенные операции языка. Выражения</i>	1		1	4	6
3	<i>Последовательные операторы</i>	1		1	4	6
4	<i>Операторы, управляющие ходом выполнения программы</i>	1		1	4	6
5	<i>Тип данных указатель</i>	1		1	5	7

№ п/п	Наименование темы (раздела) дисциплины	Лекционные занятия	Практические занятия	Лабораторные занятия	Самостоятельная работа	Всего
6	<i>Массивы и указатели</i>	1		1	5	7
7	<i>Функции, описание и вызов</i>	1		2	5	8
8	<i>Рекурсия</i>	1		2	5	8
9	<i>Структуры</i>	1		2	5	8
10	<i>Объединения</i>	1		2	5	8
11	<i>Простейшие динамические структуры данных</i>	1		2	5	8
12	<i>Обзор динамических структур и способов их построения</i>	1			5	6
13	<i>Двоичное дерево поиска</i>	1			5	6
14	<i>Способы определения имен типов</i>	1			5	6
15	<i>Особенности применения данных функционального типа. Управление распределением статической памяти.</i>	1			5	6
16	<i>Функции с переменным количеством параметров. Препроцессорные средства.</i>	1			5	6
		16	0	16	76	108

#### **14. Методические указания для обучающихся по освоению дисциплины**

**При использовании дистанционных образовательных технологий** и электронного обучения выполнять все указания преподавателей, вовремя подключаться к online занятиям, ответственно подходить к заданиям для самостоятельной работы.

**Внеаудиторная самостоятельная работа** студентов включает проработку материалов лекций, изучение рекомендованной литературы, подготовку к лабораторным работам и их защитами, подготовку к периодическому компьютерному тестированию, подготовку к устному опросу и зачету.

**Самостоятельная работа в аудитории** выполняется под непосредственным руководством преподавателя. Для повышения эффективности руководства при проведении лабораторных занятий, призванных обеспечить выборочное использование лекционного материала для более глубокого изучения вариантов использования отдельных синтаксических конструкций и приемов программирования при решении соответствующих практических задач, необходимо ссылаться на материалы лекций, показывая место решаемых задач в общем плане изложения. Проверка результатов выполнения работы должна включать требования хорошего стиля программирования, специфического для изучаемого языка. Примерный перечень задач с рекомендациями относительно содержания лабораторного занятия приводится ниже.

- Изучение особенностей реализации консольных приложений при помощи популярных сред разработки. Индивидуальный выбор студентами среды для дальнейшего использования на лабораторных занятиях.
- Индивидуальное задание №1: расчет значения числового ряда (произведения, интеграла). Основная цель – освоение специфических для языка операций и приемов составления выражений.
- Индивидуальное задание №2: определенное преобразование одномерного массива (символьной строки). Основная цель – освоение приемов манипуляции элементами массивов при помощи операций над указателями.
- Индивидуальное задание №3: определенное преобразование двумерной матрицы реализовать в виде функции. Цель: освоение приемов передачи динамических матриц в качестве параметров.
- Индивидуальное задание №4: реализация определенного рекурсивного алгоритма. Выполнение дополнительных заданий с целью углубленного изучения материала, в частности, приемов использования косвенной рекурсии.
- Индивидуальное задание №5: реализация приложения – модели СУБД (данные структурируются в виде массива структур с вариантами).
- Индивидуальное задание №6: реализация приложения – модели СУБД (данные структурируются в виде односвязного списка структур с вариантами). Выполнение дополнительных заданий с целью углубленного изучения материала, в частности, оформление проекта модели СУБД в нескольких модулях, реализация контекстно-зависимого меню.

Учитывая разницу темпов выполнения индивидуальных заданий, преподаватель обеспечивает выполнение студентами дополнительных заданий (общих для студентов, решивших основную задачу) с целью углубленного изучения материала.

#### **15. Перечень основной и дополнительной литературы, ресурсов интернет, необходимых для освоения дисциплины**

а) основная литература:

№ п/п	Источник
1	Фридман, А.Л. Язык программирования Си++ : [16+] / А.Л. Фридман. – 2-е изд., исправ. – Москва : Национальный Открытый Университет «ИНТУИТ», 2016. – 219 с. – (Основы информационных технологий). – Режим доступа: по подписке. – URL: <a href="https://biblioclub.ru/index.php?page=book&amp;id=578114">https://biblioclub.ru/index.php?page=book&amp;id=578114</a>
2	Волкова, Т.И. Введение в программирование : учебное пособие / Т.И. Волкова. – Москва ; Берлин : Директ-Медиа, 2018. – 139 с. : ил., схем., табл. – Режим доступа: по подписке. – URL: <a href="https://biblioclub.ru/index.php?page=book&amp;id=493677">https://biblioclub.ru/index.php?page=book&amp;id=493677</a>

б) дополнительная литература:

№ п/п	Источник
1	Прата Стивен. Язык программирования С++ : Лекции и упражнения : учебник / Стивен Прата. — Москва : DiaSoft, 2005. – 1097 с.
2	Язык С++ / И.Ф. Астахова [и др.]. — Минск : Новое знание, 2003. — 203 с.
3	Язык С++ : Пробное учебное пособие / И.Ф. Астахова [и др.]. — Воронеж : Воронеж. гос. ун-т, 2001. — 150 с.
4	Керниган Б. Язык программирования Си. Задачи по языку Си / Б. Керниган, Д. Ритчи, А. Фьюэр. — Москва : Финансы и статистика, 1985. – 279 с.
5	Подбельский В.В. Язык СИ++ / В.В. Подбельский. — Москва. : Финансы и статистика, 2007. — 559 с.
6	Уэйт М. Язык Си. Руководство для начинающих / М. Уэйт, С. Прата, Д. Мартин. – Москва : Мир, 1988. — 512 с.
7	Белецкий Я. Энциклопедия языка Си / Я. Белецкий. — Москва : Мир, 1992.— 686 с.

в) информационные электронно-образовательные ресурсы:

№ п/п	Ресурс
1	<a href="http://www.lib.vsu.ru">http://www.lib.vsu.ru</a> ЗНБ ВГУ
2	<a href="https://edu.vsu.ru">https://edu.vsu.ru</a> Образовательный портал "Электронный университет ВГУ"

**16. Перечень учебно-методического обеспечения для самостоятельной работы**

№ п/п	Источник
1	Рекомендованная программой литература.
2	Конспекты лекций.



№ п/п	Источник
3	Автоматизированная система контроля знаний собственной разработки.

**17. Информационные технологии, используемые для реализации учебной дисциплины, включая программное обеспечение и информационно-справочные системы (при необходимости):**

В качестве интегрированной среды разработки при проведении лабораторных работ студентам рекомендуется использование установленного в компьютерных классах факультета компьютерных наук лицензионного программного обеспечения Microsoft Visual Studio. Для текущего контроля успеваемости используется автоматизированная система контроля знаний собственной разработки. Система построена по технологии генерации учебно-тренировочных задач (УТЗ), позволяющей автоматически формировать условия и эталонные результаты задач. Исходным материалом для построения текста УТЗ служат заранее разработанные модели (шаблоны), предусматривающие введение в автоматическом режиме совокупности случайных параметров. Текст УТЗ включает фрагменты программ на языке Си (конструкции, реализующие типовые алгоритмические структуры), а ответ требует предъявления определенных результатов работы фрагментов.

Внеаудиторная самостоятельная работа, как правило, предполагает использование студентами среды разработки Microsoft Visual Studio, официальная процедура установки которой на собственные компьютеры студентов обеспечена факультетом компьютерных наук. Тем не менее, студенты никак не ограничиваются требованиями к их выбору средств разработки для использования в самостоятельной работе.

При реализации дисциплины могут использоваться технологии электронного обучения и дистанционные образовательные технологии на базе портала edu.vsu.ru, а также другие доступные ресурсы сети Интернет.

**18. Материально-техническое обеспечение дисциплины:**

компьютерные классы факультета компьютерных наук, лицензионное программное обеспечение Microsoft Visual Studio, программное обеспечение для проведения электронного тестирования собственной разработки.

**19. Фонд оценочных средств:**

**19.1 Перечень компетенций с указанием этапов формирования и планируемых результатов обучения**

Код и содержание компетенции (или ее части)	Планируемые результаты обучения (показатели достижения заданного уровня освоения компетенции посредством формирования знаний, умений, навыков)	Этапы формирования компетенции (разделы (темы) дисциплины или модуля и их наименование)	ФОС* (средства оценивания)
ПК-12 способностью разрабатывать средства реализации информационных технологий (методические, информационные, математические, алгоритмические, технические и программные)	Знать синтаксические конструкции языка программирования и их семантику Уметь применять языковые конструкции для решения практических задач, использовать библиотеки стандартных функций, поставляемых с языком программирования	Разделы 1 - 5 Особенности построения программных проектов на языке Си. Представление данных. Основные встроенные операции языка. Выражения. Последовательные операторы. Операторы, управляющие ходом выполнения программы.	Комплекты задач №1,2 Комплекты тестов №1-3
	Владеть навыками разработки приложений с использованием современных интегрированных средств	Разделы 6 - 9 Тип данных указатель. Массивы и указатели. Функции, описание и вызов. Рекурсия	Комплекты задач №3,4 Комплекты тестов №4,5

Код и содержание компетенции (или ее части)	Планируемые результаты обучения (показатели достижения заданного уровня освоения компетенции посредством формирования знаний, умений, навыков)	Этапы формирования компетенции (разделы (темы) дисциплины или модуля и их наименование)	ФОС* (средства оценивания)
ПК-17 способностью использовать технологии разработки объектов профессиональной деятельности в областях: машиностроение, приборостроение, техника, образование, медицина, административное управление, юриспруденция, бизнес, предпринимательство, коммерция, менеджмент	Знать общие приемы структурирования программного кода и обрабатываемых данных Уметь определять структуры данных при проектировании алгоритмов, разбивать решение сложной задачи на последовательность более простых задач Владеть навыками тестирования и отладки программных модулей, реализованных на языке Си	Разделы 10 – 17 Структуры. Объединения. Простейшие динамические структуры данных. Обзор динамических структур и способов их построения. Двоичное дерево поиска. Способы определения имен типов. Особенности применения данных функционального типа. Управление распределением статической памяти. Функции с переменным количеством параметров. Препроцессорные средства.	Комплекты задач №5,6 Комплекты тестов №6,7
Промежуточная	аттестация		Комплект КИМ

## 19.2 Описание критериев и шкалы оценивания компетенций (результатов обучения) при промежуточной аттестации

Для оценивания результатов обучения на зачете используются следующие показатели (ЗУНЫ из 19.1):

1. знание синтаксических конструкций языка программирования и их семантики;
2. умение применять языковые конструкции для решения практических задач, использовать библиотеки стандартных функций, поставляемых с языком программирования;
3. владение навыками разработки приложений с использованием современных интегрированных средств;
4. знание общих приемов структурирования программного кода и обрабатываемых данных;
5. умение определять структуры данных при проектировании алгоритмов, разбивать решение сложной задачи на последовательность более простых задач;
6. владение навыками тестирования и отладки программных модулей, реализованных на языке

Си.

Для оценивания результатов обучения на зачете используется – зачтено, не зачтено.

Соотношение показателей, критериев и шкалы оценивания результатов обучения.

Критерии оценивания компетенций	Уровень сформированности компетенций	Шкала оценок
<p><i>Сформированные знания о синтаксических конструкциях языка программирования и их семантике. Сформированное умение применять языковые конструкции для решения практических задач, использовать библиотеки стандартных функций, поставляемых с языком программирования. Сформированные навыки разработки приложений с использованием современных интегрированных средств. Сформированные знания об общих приемах структурирования программного кода и обрабатываемых данных. Сформированное умение определять структуры данных при проектировании алгоритмов, разбивать решение сложной задачи на последовательность более простых задач. Сформированные навыки тестирования и отладки программных модулей, реализованных на языке Си</i></p>	<p><i>Повышенный уровень</i></p>	<p><i>Отлично</i></p>
<p><i>Сформированные, но содержащие отдельные пробелы представления о синтаксических конструкциях языка программирования и их семантике. Успешное, но содержащее отдельные пробелы умение применять языковые конструкции для решения практических задач, использовать библиотеки стандартных функций, поставляемых с языком программирования. Сформированные, но содержащие отдельные пробелы навыки разработки приложений с использованием современных интегрированных средств. Сформированные, но содержащие отдельные пробелы представления об общих приемах структурирования программного кода и обрабатываемых данных. Успешное, но содержащее отдельные пробелы умение определять структуры данных при проектировании алгоритмов, разбивать решение сложной задачи на последовательность более простых задач. Сформированные, но содержащие отдельные пробелы навыки тестирования и отладки программных модулей, реализованных на языке Си.</i></p>	<p><i>Базовый уровень</i></p>	<p><i>Хорошо</i></p>

<p><i>Неполное представление о синтаксических конструкциях языка программирования и их семантике. Успешное, но не системное умение применять языковые конструкции для решения практических задач, использовать библиотеки стандартных функций, поставляемых с языком программирования. Неполное владение навыками разработки приложений с использованием современных интегрированных средств. Неполное представление об общих приемах структурирования программного кода и обрабатываемых данных. Успешное, но не системное умение определять структуры данных при проектировании алгоритмов, разбивать решение сложной задачи на последовательность более простых задач. Неполное владение навыками тестирования и отладки программных модулей, реализованных на языке Си.</i></p>	<p><i>Пороговый уровень</i></p>	<p><i>Удовлетворительно</i></p>
<p><i>Фрагментарные знания или отсутствие знаний. Фрагментарные умения или отсутствие умений. Фрагментарные навыки или отсутствие навыков.</i></p>	<p><i>-</i></p>	<p><i>Неудовлетворительно</i></p>

### **19.3 Типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений, навыков и (или) опыта деятельности, характеризующие этапы формирования компетенций в процессе освоения образовательной программы**

#### **19.3.1. Примерный перечень практических заданий**

Индивидуальное задание №1 Расчет значения числового ряда (произведения, интеграла). Основная цель – освоение специфических для языка операций и приемов составления выражений. Оценка «зачтено» выставляется, если разработанная согласно индивидуальному заданию программа выдает результат расчета с допустимой точностью, которая задается оператором.

Индивидуальное задание №2 Определенное преобразование одномерного массива (символьной строки). Основная цель – освоение приемов манипуляции элементами массивов при помощи операций над указателями. Оценка «зачтено» выставляется, если разработанная согласно индивидуальному заданию программа выполняет необходимые действия по преобразованию одномерного массива с использованием операций над указателями.

Индивидуальное задание №3 Определенное преобразование двумерной матрицы реализовать в виде функции. Цель: освоение приемов передачи динамических матриц в качестве параметров. Оценка «зачтено» выставляется, если разработанная согласно индивидуальному заданию программа выполняет необходимые действия по преобразованию двумерной матрицы, хранение которой реализовано в динамической памяти в форме массива указателей на одномерные массивы-строки; собственно преобразование должно быть реализовано в виде функции с передачей матриц в качестве параметров.

Индивидуальное задание №4 Реализация определенного рекурсивного алгоритма. Выполнение дополнительных заданий с целью углубленного изучения материала, в частности, приемов использования косвенной рекурсии. Оценка «зачтено» выставляется, если разработанная согласно индивидуальному заданию программа выдает результат расчета с допустимой точностью, которая задается оператором.

Индивидуальное задание №5 Реализация приложения – модели СУБД (данные структурируются в виде массива структур с вариантами). Оценка «зачтено» выставляется, если разработанная согласно индивидуальному заданию программа реализует действия по хранению/преобразованию данных в последовательности, которая задается оператором.

Индивидуальное задание №6 Реализация приложения – модели СУБД (данные структурируются в виде односвязного списка структур с вариантами). Выполнение дополнительных заданий с целью углубленного изучения материала, в частности, оформление проекта модели СУБД в нескольких модулях, реализация контекстно-зависимого меню. Оценка «зачтено» выставляется, если разработанная согласно индивидуальному заданию программа реализует действия по хранению/преобразованию данных в последовательности, которая задается оператором.

### **19.3.2. Тестовые задания**

Для генерации практически неограниченного количества тестовых заданий комплектов 1-7 используется автоматизированная система контроля знаний собственной разработки. Система построена по технологии генерации учебно-тренировочных задач (УТЗ), позволяющей автоматически формировать условия и эталонные результаты задач. Исходным материалом для построения текста УТЗ служат заранее разработанные модели (шаблоны), предусматривающие введение в автоматическом режиме совокупности случайных параметров. Текст УТЗ включает фрагменты программ на языке Си (конструкции, реализующие типовые алгоритмические структуры), а ответ требует предъявления определенных результатов работы фрагментов.

Оценка «зачтено» выставляется, если безошибочно выполнено не менее 66% простых заданий либо не менее 50% заданий повышенной сложности

Тест №1. Выражения Цель контроля: специфические для языка операции и приемы составления выражений. Обучаемому предлагается на выбор 2 набора из 10 сгенерированных случайным образом тестовых вопросов каждый: набор относительно простых задач и набор задач повышенной сложности. Общее количество вариантов вопросов теста неограниченно.

Тест №2. Операторы Цель контроля: специфические для языка операторы передачи управления. Обучаемому предлагается на выбор 2 набора из 10 сгенерированных случайным образом тестовых вопросов каждый: набор относительно простых задач и набор задач повышенной сложности. Общее количество вариантов вопросов теста неограниченно.

Тест №3. Указатели Цель контроля: использование указателей на статические и динамические переменные внутри операторов. Обучаемому предлагается на выбор 2 набора из 10 сгенерированных случайным образом тестовых вопросов каждый: набор относительно простых задач и набор задач повышенной сложности. Общее количество вариантов вопросов теста неограниченно.

Тест №4. Массивы Цель контроля: приемы манипуляции элементами массивов при помощи операций над указателями. Обучаемому предлагается на выбор 2 набора из 10 сгенерированных случайным образом тестовых вопросов каждый: набор относительно простых задач и набор задач повышенной сложности. Общее количество вариантов вопросов теста неограниченно.

Тест №5. Функции Цель контроля: способы передачи параметров и возврата значений функций, реализация рекурсивных функций, включая варианты косвенной рекурсии. Обучаемому предлагается на выбор 2 набора из 10 сгенерированных случайным образом тестовых вопросов каждый: набор относительно простых задач и набор задач повышенной сложности. Общее количество вариантов вопросов теста неограниченно.

Тест №6. Структуры Цель контроля: структурирование данных при помощи конструкций «структура» и «объединение», включая варианты вложенного использования конструкций. Обучаемому предлагается на выбор 2 набора из 10 сгенерированных случайным образом тестовых вопросов каждый: набор относительно простых задач и набор задач повышенной сложности. Общее количество вариантов вопросов теста неограниченно.

Тест №7. Списки Цель контроля: реализация динамических структур данных (списки, очереди, стеки, деревья). Обучаемому предлагается на выбор 2 набора из 10 сгенерированных случайным образом тестовых вопросов каждый: набор относительно простых задач и набор задач повышенной сложности.

сложности. Общее количество вариантов вопросов теста неограниченно.

#### **19.4. Методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций**

Оценка знаний, умений и навыков, характеризующая этапы формирования компетенций в рамках изучения дисциплины осуществляется в ходе текущей и промежуточной аттестаций.

Текущая аттестация проводится в соответствии с Положением о текущей аттестации обучающихся по программам высшего образования Воронежского государственного университета. Текущая аттестация проводится в формах: устного опроса (индивидуальный опрос, фронтальная беседа); защиты лабораторных работ; а также тестирования с использованием автоматизированной системы контроля знаний собственной разработки. Система построена по технологии генерации учебно-тренировочных задач (УТЗ), позволяющей автоматически формировать условия и эталонные результаты задач. Исходным материалом для построения текста УТЗ служат заранее разработанные модели (шаблоны), предусматривающие введение в автоматическом режиме совокупности случайных параметров. Текст УТЗ включает фрагменты программ на языке Си (конструкции, реализующие типовые алгоритмические структуры), а ответ требует предъявления определенных результатов работы фрагментов. При оценивании результатов устного опроса и защиты лабораторных работ используется качественная шкала оценок. Оценивание результатов тестирования предполагает использование количественной шкалы.

Промежуточная аттестация проводится в соответствии с Положением о промежуточной аттестации обучающихся по программам высшего образования. Промежуточная аттестация по итогам освоения дисциплины проводится в форме зачета. Условиями для положительной итоговой оценки являются: выполнение всех лабораторных работ, перечисленных в приложении под наименованием «индивидуальное задание», а также успешное прохождение всех этапов компьютерного тестирования.

При оценивании используется следующая шкала:

5 баллов ставится, если обучающийся демонстрирует полное соответствие знаний, умений, навыков приведенным в таблицах показателям, свободно оперирует приобретенными знаниями, умениями, применяет их при решении практических задач;

4 балла ставится, если обучающийся демонстрирует соответствие знаний, умений, навыков приведенным в таблицах показателям, но допускает незначительные ошибки, неточности, испытывает затруднения при решении практических задач;

3 балла ставится, если обучающийся демонстрирует неполное соответствие знаний, умений, навыков приведенным в таблицах показателям, допускает значительные ошибки при решении практических задач;

2 балла ставится, если обучающийся демонстрирует явное несоответствие знаний, умений, навыков приведенным в таблицах показателям.

При сдаче зачета

«зачтено» - 3-5 баллов

«не зачтено» - 2 балла.