

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

УТВЕРЖДАЮ

Заведующий кафедрой
функционального анализа
и операторных уравнений

 Каменский М.И.
подпись, расшифровка подписи
26.06.2018 г.

РАБОЧАЯ ПРОГРАММА УЧЕБНОЙ ДИСЦИПЛИНЫ

Б1.Б.13 Технология и методы программирования

- 1. Шифр и наименование направления подготовки / специальности:** 10.05.04
информационно-аналитические системы безопасности
- 2. Профиль подготовки / специализации**
- 3. Квалификация (степень) выпускника:** специалист
- 4. Форма образования:** очная
- 5. Кафедра, отвечающая за реализацию дисциплины:** функционального анализа
и операторных уравнений
- 6. Исполнители программы:** Груздев Денис Владиславович, математический фа-
культет, кафедра функционального анализа и операторных уравнений,
denjes@yandex.ru
- 7. Рекомендована:** НМС математического факультета протокол № 0500-07 от
03.07.18
- 8. Учебный год:** 2018-2019 **Семестр(ы):** третий

9. Цели и задачи учебной дисциплины:

В результате освоения дисциплины обучающийся должен:

Знать: основные приемы решения задач обработки текстовой и числовой информации, приемы алгоритмизации, традиционные структуры данных, основные требования методологии структурного программирования, как технологической основы разработки качественных программных компонентов, понятие статических и динамических данных, примеры базовых структур данных, подходы процедурного, модульного, программирования, реализацию вызова процедур в языках с блочной структурой, рекурсию.

Уметь: применять современные методы, средства разработки алгоритмов и программ для решения широкого круга задач, применять требования методологии структурного программирования при проектировании информационных моделей, формализовать поставленную задачу, тестировать и отлаживать программы в интегрированной среде разработки, опираясь на знания теоретических основ программирования, оптимизировать исходный код.

Владеть: навыками практического программирования конкретных задач в определенной языковой среде, применять средства структурного, модульного программирования для решения задач.

10. Место учебной дисциплины в структуре ООП: Дисциплина входит в базовую часть профессионального цикла.

11. Компетенции обучающегося, формируемые в результате освоения дисциплины:

| Компетенция | | Планируемые результаты обучения |
|-------------|---|--|
| Код | Название | |
| ОК-8 | способностью к самоорганизации и самообразованию | <p>знать: Принципы работы с инструментами для поиска и обработки больших объёмов информации</p> <p>уметь: Уметь писать доклады и делать презентации на заданную тему с использованием современных информационных технологий для поиска и обработки;</p> <p>владеть (иметь навык(и)): Основными инструментами для поиска и обработки больших объёмов информации по профилю деятельности в глобальных компьютерных системах, сетях, в библиотечных фондах и в иных источниках информации</p> |
| ОПК-4 | способностью применять в профессиональной деятельности языки и системы программирования, инструментальные средства разработки программного обеспечения, | <p>Знать: языки и системы программирования, инструментальные средства разработки программного обеспечения, современные методы и технологии программирования</p> <p>Уметь: применять в профессиональной де-</p> |

| | | |
|-------|--|--|
| | современные методы и технологии программирования | <p>ательности знания языков программирования.</p> <p>Владеть: современными методами и технологиями программирования и использовать их в решении задач</p> |
| ПК-12 | способностью разрабатывать программное и иные виды обеспечения специальных ИАС | <p>Знать: языки и системы программирования, инструментальные средства разработки программного обеспечения, современные методы и технологии программирования</p> <p>Уметь: разрабатывать программное и иные виды обеспечения специальных ИАС</p> <p>Владеть: способностью разрабатывать программное и иные виды обеспечения специальных ИАС</p> |

12. Объем дисциплины в зачетных единицах/часах в соответствии с учебным планом — 3/108.

Форма промежуточной аттестации зачёт с оценкой

13. Виды учебной работы

| Вид учебной работы | Трудоемкость (часы) | | |
|--------------------------------|---------------------|--------------------------------|--|
| | Всего | В том числе интерактивные часы | По семестрам |
| | | | сем.3 |
| Аудиторные занятия | 68 | | 68 |
| в том числе: | | | |
| лекции | 34 | | 34 |
| практические | 0 | | 0 |
| лабораторные | 34 | | 34 |
| контроль | 0 | | 0 |
| Самостоятельная работа | 40 | | 40 |
| Итого: | 108 | | 108 |
| Форма промежуточной аттестации | | | 1 контрольная работа, Зачет с оценкой |

13.1. Содержание разделов дисциплины

| № п/п | Наименование раздела дисциплины | Содержание раздела дисциплины |
|-------|---|---|
| 1. | Тема 1.1. Структурированные типы данных | Массивы. Одномерные массивы. Многомерные массивы. Строковый тип. Множества. Записи. Фиксированные записи. Записи с вариантами. Оператор присоединения. |
| 2. | Тема 2.1. Подпрограммы и их использование | Основные понятия. Классификация и вызов подпрограмм. Область действия идентификаторов при использовании подпрограмм. Структура процедур и функций. Классификация способов при передачи параметров. Передача параметров в Паскале. Списки формальных параметров. Параметры - значения. Параметры - переменные. Параметры - константы. Бестиповые параметры. Открытые параметры - массивы. Процедурные директивы. Рекурсия. Понятие ре- |

| | | |
|----|---|--|
| | | курсии. Формы рекурсивных процедур |
| 3. | Тема 2.2. Основные алгоритмы обработки данных | Вычисление по формулам, последовательный и бинарный поиск, сортировка, итерационные алгоритмы поиска корней уравнений, индуктивная обработка последовательностей данных, рекуррентные вычисления. Алгоритмы обработки данных, возникающих в задачах алгебры, математического анализа, математической статистики, задачах линейного программирования. Рекурсивные и итерационные алгоритмы обработки данных. Условия обеспечивающие завершение последовательности рекурсивных вызовов. Идеи реализации рекурсивных вызовов в подпрограммах. Инвариантная функция и инвариант цикла. Взаимосвязь итерации и рекурсии, индуктивное вычисление функций на последовательности данных. |
| 4. | Тема 3.1. Файловые структуры данных | Файловый тип. Операции над файлами. Установочные и завершающие операции. Ввод и вывод. Перемещения по файлу. Специальные операции. Операции с текстовыми файлами. Операции с типизированными файлами. Нетипизированные, или бестиповые файлы. Процедуры и функции для работы с бестиповыми файлами. |

13.2. Разделы дисциплины и виды занятий

| № п/п | Наименование раздела дисциплины | Виды занятий (часов) | | | | |
|--------|-------------------------------------|----------------------|----------|--------------|------------------------|-------|
| | | Лекции | Контроль | Лабораторные | Самостоятельная работа | Всего |
| 1. | Структурированные типы данных | 10 | 0 | 10 | 10 | 32 |
| 2. | Подпрограммы и их использование | 6 | 0 | 4 | 10 | 20 |
| 3. | Основные алгоритмы обработки данных | 8 | 0 | 10 | 10 | 28 |
| 4. | Файловые структуры данных | 10 | 0 | 10 | 10 | 28 |
| Итого: | | 34 | 0 | 34 | 40 | 108 |

14. Методические указания для обучающихся по освоению дисциплины

Аудиторные занятия, лекции и лабораторные занятия, предполагают самостоятельную работу студентов по данному курсу. Ряд тем выносятся для самостоятельного изучения, предлагаются темы для создания докладов с презентациями. Предусмотрены домашние задания и оформление отчетов выполнения лабораторных заданий, а также дополнительные задания для сильных студентов.

15. Учебно-методическое и информационное обеспечение дисциплины

а) основная литература:

| № п/п | Источник |
|-------|--|
| 1. | Рудалев, Валерий Геннадьевич. Технология визуального программирования : учебное пособие для вузов / В.Г. Рудалев ; Воронеж. гос. ун-т .— Воронеж |
| 2. | Иванова, Галина Сергеевна. Технология программирования : Учебник для студ. вузов, обуч. по направлению "Информатика и вычислительная техника", специальностям: "Вычислительные машины, комплексы, системы и сети", "Автоматизированные системы обработки информации и управления", "Программное обеспечение вычислительной техники и информационных систем" / Г.С. Иванова .— 2-е изд., стер. — М. : Изд-во МГТУ имени Н.Э. Баумана, 2003 .— 319 с. |

б) дополнительная литература:

| № п/п | Источник |
|-------|----------|
|-------|----------|

| | |
|----|---|
| 3. | Камаев, Валерий Анатольевич. Технологии программирования : учебник для вузов по направлению подгот. специалистов "Информатика и вычисл. техника" / В. А. Камаев, В. В. Костерин .— М. : Высш. шк., 2005 .— 358, [1] с. : ил. — Библиогр.: с. 354-355 |
| 4. | Кулямин, Виктор Вячеславович. Технологии программирования. Компонентный подход : учебное пособие / В.В. Кулямин .— М. : Бином.Лаборатория знаний : Интернет-Университет Информационных Технологий, 2007 .— 463 с. |

в) информационные электронно-образовательные ресурсы:

| № п/п | Источник |
|-------|---|
| 6. | Лекции по технологии программирования http://sp.cmc.msu.ru/info/3/techprog.htm |
| 7. | Дейтел, Х.М. Технологии программирования на Java 2 / Х.М. Дейтел, П.Д. Дейтел, С.И. Сантри .— М. : Бином, 2003 |

16. Перечень учебно-методического обеспечения для самостоятельной работы (учебно-методические рекомендации, пособия, задачки, методические указания по выполнению практических (контрольных) работ и др.)

| № п/п | Источник |
|-------|---|
| 1 | Завгородний, Михаил Григорьевич. Компьютерные сети [Электронный ресурс] : лабораторный практикум / М.Г. Завгородний, С.П. Майорова ; Воронеж. гос. ун-т; [ред. В.В. Юргелас] .— Электрон. текстовые дан. — Воронеж : Издательский дом ВГУ, 2014 .— Загл. с титул. экрана .— Свободный доступ из интрасети ВГУ .— Текстовые файлы .— Windows 2000 ; Adobe Acrobat Reader .— <URL: http://www.lib.vsu.ru/elib/texts/method/vsu/m14-134.pdf > |

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

Microsoft Visual Studio, Microsoft Office

18. Материально-техническое обеспечение дисциплины:

(при использовании лабораторного оборудования указывать полный перечень, при большом количестве оборудования можно вынести данный раздел в приложение к рабочей программе)

Компьютерный класс (14-15 компьютеров + программное обеспечение)

19. Фонд оценочных средств:

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

| Код и содержание компетенции | 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) |
|--|--|---|---------------------------|
| ОК-8 способность к самоорганизации и самообразованию | знать: Принципы работы с инструментами для поиска и обработки больших объёмов информации | Раздел 1. Структурированные типы данных Раздел 2. Подпрограммы и их использование | Контрольная работа |

| | | | |
|--|---|--|---------------------------|
| | уметь: Уметь писать доклады и делать презентации на заданную тему с использованием современных информационных технологий для поиска и обработки; | Раздел 1. Структурированные типы данных Раздел 2. Подпрограммы и их использование | Контрольная работа |
| | владеть (иметь навык(и)): Основными инструментами для поиска и обработки больших объёмов информации по профилю деятельности в глобальных компьютерных системах, сетях, в библиотечных фондах и в иных источниках информации | Раздел 1. Структурированные типы данных Раздел 2. Подпрограммы и их использование | Контрольная работа |
| ОПК-4 способностью применять в профессиональной деятельности языки и системы программирования, инструментальные средства разработки программного обеспечения, современные методы и технологии программирования. | Знать: языки и системы программирования, инструментальные средства разработки программного обеспечения, современные методы и технологии программирования. | Раздел 1. Структурированные типы данных Раздел 2. Подпрограммы и их использование | Контрольная работа, зачет |
| | Уметь: применять в профессиональной деятельности знания языков программирования. | Раздел 2. Подпрограммы и их использование Раздел 3. Основные алгоритмы обработки данных Раздел 4. Файловые структуры данных | Контрольная работа, зачет |
| | Владеть: современными методами и технологиями программирования и использовать их в решении задач. | Раздел 2. Подпрограммы и их использование Раздел 3. Основные алгоритмы обработки данных Раздел 4. Файловые структуры данных | Контрольная работа, зачет |
| ПК-12 способностью разрабатывать программное и иные виды обеспечения специальных ИАС | Знать: языки и системы программирования, инструментальные средства разработки программного обеспечения, современные методы и технологии программирования. | Раздел 2. Подпрограммы и их использование Раздел 3. Основные алгоритмы обработки данных Раздел 4. Файловые структуры данных | Контрольная работа, зачет |
| | Уметь: разрабатывать программное и иные виды | Раздел 2. Подпрограммы и их использование | Контрольная работа, зачет |

| | | | |
|--|---|--|---------------------------|
| | обеспечения специальных ИАС | Раздел 3. Основные алгоритмы обработки данных Раздел 4. Файловые структуры данных | |
| | Владеть: современными методами и технологиями программирования и использовать их в решении задач. | Раздел 2. Подпрограммы и их использование Раздел 3. Основные алгоритмы обработки данных Раздел 4. Файловые структуры данных | Контрольная работа, зачет |

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

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

Текущая аттестация проводится в форме лабораторных работ и контрольной работы.

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 с оценкой

оценка «отлично» - 5 баллов

оценка «хорошо» - 4 балла

оценка «удовлетворительно» - 3 балла

оценка «неудовлетворительно» - 2 балла.

| Критерии оценивания компетенций (зачет с оценкой) | Уровень сформированности компетенций | Шкала оценок |
|---|--------------------------------------|----------------|
| <i>Обучающийся в полной мере владеет понятийным аппаратом в области программирования и технологии работы на ЭВМ, способен иллюстрировать ответ примерами, фактами, применять теоретические знания для решения практических задач программирования, СУБД и сетевых технологий.</i> | <i>Повышенный уровень</i> | <i>Отлично</i> |
| <i>У обучающегося сформированы знания, умения и навыки про-</i> | <i>Базовый уро-</i> | <i>Хорошо</i> |

| | | |
|---|--------------------------|----------------------------|
| <i>граммирования и технологии работы на ЭВМ; он способен иллюстрировать ответ примерами, фактами, применять теоретические знания для решения практических задач; но допускает отдельные несущественные пробелы в своих знаниях, допускает ошибки при выполнении практических задач.</i> | <i>вень</i> | |
| <i>У обучающегося сформированы неполные знания, умения и навыки; он допускает отдельные существенные пробелы в своих знаниях, допускает существенные ошибки при выполнении практических задач.</i> | <i>Пороговый уровень</i> | <i>Удовлетворительно</i> |
| <i>Сформированы лишь фрагментарные знания, умения и навыки или знания, умения и навыки отсутствуют</i> | <i>–</i> | <i>Неудовлетворительно</i> |

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 Перечень вопросов к зачету:

1. Языки программирования. Классификация языков программирования.
2. Машинные команды. Языки программирования низкого уровня.
3. Языки программирования высокого уровня. Алгоритмические языки.
4. Объектно-ориентированное и обобщенное программирование.
5. Языки программирования C и C++ (создание, предназначение, возможности).
6. Инструментарий создания приложений (компиляторы, интерпретаторы, архиваторы, компоновщики, отладчики).
7. Интегрированная среда разработки приложений.
8. Препроцессор и макрообработка языка C++. Директивы препроцессора.
9. Приложение Microsoft Visual Studio 2010.
10. Этапы решения задач на компьютере.
11. Базовые средства языка C++ (состав языка, алфавит языка, идентификаторы, ключевые слова).
12. Вывод сообщений на экран. Комментарии. Примеры.
13. Данные (константы, переменные, типы данных).
14. Целые числа и их двоичное представление.
15. Вещественные числа с фиксированной и плавающей точкой. Вещественные числа однократной и двукратной точности.
16. Операторы объявления, присваивания и ввода данных с клавиатуры. Составные операторы присваивания. Примеры.
17. Арифметические операции. Примеры.
18. Операции отношений и логические операции. Примеры.
19. Оператор выбора if. Примеры.
20. Оператор «знак ?». Примеры.
21. Оператор выбора switch. Примеры.
22. Вложенные инструкции if и switch. "Лестничная" конструкция if-else-if
23. Итерационный оператор for. Использование нескольких управляющих переменных. Отсутствие элементов заголовка и циклы без тела. Примеры.
24. Итерационный оператор while. Примеры.

25. Итерационный оператор do-while. Примеры.
26. Инструкции break и continue. Оператор goto. Примеры.
27. Структурированный тип данных – массив (одномерный, двумерный, многомерный). Примеры.
28. Структурированный тип данных – строка. Ввод строк с клавиатуры. Примеры.
29. Библиотечные функции обработки строк и их использование. Примеры.
30. Инициализация массивов (безразмерных массивов). Массивы строк и их инициализация. Примеры.
31. Указатели. Операторы, используемые с указателями. Примеры.
32. Базовый тип указателя. Присваивание значений и проведение вычислений с помощью указателей. Примеры.
33. Арифметические и логические операции над указателями. Примеры.
34. Указатели массивов. Примеры.
35. Индексирование указателя. Строковые константы и указатели. Примеры.
36. Массивы указателей. Нулевые указатели. Многоуровневая непрямая адресация. Примеры.
37. Функции языка C++. Общий формат функций. Создание void-функции и использование ее аргументов. Примеры.
38. Создание функции, возвращающей значение, и использование ее в выражениях. Использование инструкции return. Примеры.
39. Правила действия областей видимости функций. Локальная область видимости. Примеры.
40. Объявление локальных переменных внутри блока. Скрытие имен. Примеры.
41. Глобальная область видимости. Примеры.
42. Передача функции указателя. Примеры.
43. Передача функции массива. Примеры.
44. Передача функции строк. Примеры.
45. Возвращение функцией указателя. Примеры.
46. Прототипы функций. Прототипы функций, содержащиеся в стандартных заголовках. Примеры.
47. Рекурсия. Функция main(). Примеры.
48. Сортировки массивов методом выбора и методом вставки. Примеры.
49. Сортировки массивов методом обмена (пузырька) и челночным методом. Примеры.
50. Сортировка массивов методом Шелла и быстрая сортировка (методом Хоара). Примеры.
51. Турнирная и пирамидальная сортировки массивов. Примеры.
52. Способы передачи аргументов функции. Примеры.
53. Ссылочные параметры функций. Примеры.
54. Перегрузка функций. Примеры.
55. Аргументы, передаваемые функции по умолчанию. Примеры.
56. Модификаторы типов данных. Спецификаторы типа const и volatile. Примеры.
57. Статические переменные (локальные и глобальные). Регистровые переменные. Примеры.
58. Перечисления. Примеры.

59. Ключевое слово `typedef`. Примеры.
60. Поразрядные операторы. Примеры.
61. Оператор “знак запятой”. Примеры.
62. Одновременное присваивание и составные операторы присваивания. Примеры.
63. Ключевое слово `sizeof`. Примеры.
64. Сводная таблица приоритетов C++-операторов. Примеры.
65. Системы свода-вывода. Потоки. Встроенные C++-потоки.
66. Текстовые файлы и «двоичные» файлы.
67. Работа с файлами. Текстовые потоки. Настройка открытия файла. Примеры.
68. Запись текстовой информации в дисковый файл. Примеры.
69. Чтение текстовой информации из дискового файла. Примеры.
70. Неявные и явные операции приведения типов. Примеры.
71. Структура. Примеры.

Примеры заданий для зачёта

1. Решить задачу, используя указатели. Дано целое число n (вводится с клавиатуры). Если оно кратно 10, разделить его на 10.

Задачи 2-4 решить, используя функции

2. Даны квадрат со стороной a и прямоугольник со сторонами b и c . Написав функцию, вычисляющую площадь прямоугольника по двум заданным сторонам, определить, площадь какой из фигур больше. (значения a , b и c вводятся с клавиатуры).

3. Дан символ (вводится с клавиатуры). Написать `void`-функцию, заменяющую этот символ на `*` в случае, если он является буквой.

4. Дан одномерный массив, все элементы которого различны. Поменять местами максимальный и минимальный элементы. (Написать функции поиска максимального и минимального элементов).

6. Дан целочисленный массив (считывается из текстового файла), среди элементов которого есть нулевые. Записать в другой текстовый файл их индексы.

7. Дан текст (считывается из текстового файла). Подсчитать количество букв в первом слове. Результат вывести на экран.

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

Текущий контроль представляет собой проверку усвоения учебного материала теоретического и практического характера, регулярно осуществляемую на занятиях. К основным формам текущего контроля можно отнести устный опрос, проверку домашних заданий, контрольные работы. Промежуточная аттестация предназначена для определения уровня освоения всего объема учебной дисциплины «Технология программирования и работа на ЭВМ» в форме зачетов и экзаменов. Промежуточная аттестация, как правило, осуществляется в конце семестра и может завершать изучение как отдельной дисциплины, так и ее разде-

лов. Промежуточная аттестация помогает оценить более крупные совокупности знаний и умений, в некоторых случаях даже формирование определенных профессиональных компетенций. Задания текущего контроля и проведение промежуточной аттестации должны быть направлены на оценивание уровня освоения теоретических и практических понятий, научных основ профессиональной деятельности; степени готовности обучающегося применять теоретические и практические знания и практически значимую информацию; приобретение умений профессионально значимых для профессиональной деятельности.