

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ
ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«РЯЗАНСКИЙ ГОСУДАРСТВЕННЫЙ РАДИОТЕХНИЧЕСКИЙ
УНИВЕРСИТЕТ ИМЕНИ В.Ф. УТКИНА»

КАФЕДРА
«ВЫЧИСЛИТЕЛЬНАЯ И ПРИКЛАДНАЯ МАТЕМАТИКА»

**ОЦЕНОЧНЫЕ МАТЕРИАЛЫ
«НИЗКО-УРОВНЕВОЕ ПРОГРАММИРОВАНИЕ»**

Направление подготовки

09.03.04 «Программная инженерия»

Направленность (профиль) подготовки

Программная инженерия

Квалификация выпускника – бакалавр

Форма обучения – очная

Рязань

ОБЩИЕ ПОЛОЖЕНИЯ

Оценочные материалы – это совокупность учебно-методических материалов и процедур, предназначенных для оценки качества освоения обучающимися данной дисциплины как части основной образовательной программы.

Цель – оценить соответствие знаний, умений и уровня приобретенных компетенций, обучающихся целям и требованиям основной образовательной программы в ходе проведения текущего контроля и промежуточной аттестации.

Основная задача – обеспечить оценку уровня сформированности общекультурных и профессиональных компетенций, приобретаемых обучающимся в соответствии с этими требованиями.

Контроль знаний обучающихся проводится в форме текущего контроля и промежуточной аттестации. Текущий контроль успеваемости и промежуточная аттестация проводятся с целью определения степени усвоения учебного материала, своевременного выявления и устранения недостатков в подготовке обучающихся, организации работы обучающихся в ходе учебных занятий и оказания им индивидуальной помощи.

К контролю текущей успеваемости относятся проверка знаний, умений и навыков обучающихся на лабораторных работах по результатам выполнения и защиты обучающимися индивидуальных заданий, по результатам выполнения контрольных работ и тестов, по результатам проверки качества конспектов лекций и иных материалов.

В качестве оценочных средств на протяжении семестра используется устные и письменные ответы студентов на индивидуальные вопросы, письменное тестирование по теоретическим разделам курса, реферат. Дополнительным средством оценки знаний и умений студентов является отчет о проведении лабораторных работ и их защита.

По итогам курса обучающиеся сдают зачет с оценкой. Форма проведения – устный ответ с письменным подкреплением по утвержденным билетам, сформулированным с учетом содержания дисциплины. В билет для зачета включается два теоретических вопроса. В процессе подготовки к устному ответу студент должен составить в письменном виде план ответа.

ОПИСАНИЕ ПОКАЗАТЕЛЕЙ И КРИТЕРИЕВ ОЦЕНИВАНИЯ КОМПЕТЕНЦИЙ

Сформировать каждой компетенции в рамках освоения данной дисциплины оценивается по трехуровневой шкале:

- пороговый уровень (удовлетворительный) является обязательным для всех обучающихся по завершении освоения дисциплины;
- продвинутый уровень (хороший) характеризуется превышением минимальных характеристик сформированности компетенций по завершении освоения дисциплины;
- эталонный уровень (отличный) характеризуется максимально возможной выраженностью компетенций и является важным качественным ориентиром для самосовершенствования обучаемого.

При достаточном качестве освоения более 81% приведенных знаний, умений и навыков преподаватель оценивает освоение данной компетенции в рамках настоящей дисциплины на эталонном уровне, при освоении более 61% приведенных знаний, умений и навыков — на продвинутом, при освоении более 41% приведенных знаний, умений и навыков — на пороговом уровне. При освоении менее 40% приведенных знаний, умений и навыков компетенция в рамках настоящей дисциплины считается неосвоенной.

Уровень сформированности каждой компетенции на различных этапах ее формирования в процессе освоения дисциплины оценивается в ходе текущего контроля успеваемости и представлено различными видами оценочных средств.

Учитываются:

- уровень усвоения материала, предусмотренного программой курса;
- умение анализировать материал и устанавливать причинно-следственные связи;
- ответы на вопросы: полнота, аргументированность, убежденность, качество ответа (его общая композиция, логичность, общая эрудиция);
- использование основной и дополнительной литературы при подготовке, и принимаются во внимание знания, умения, навыки, перечисленные в п. 2 рабочей программы дисциплины.

Критерии оценивания уровня сформированности компетенции в процессе выполнения контрольных заданий:

41%-60% правильных ответов соответствует пороговому уровню сформированности компетенции на данном этапе ее формирования;

61%-80% правильных ответов соответствует продвинутому уровню сформированности компетенции на данном этапе ее формирования;

81%-100% правильных ответов соответствует эталонному уровню сформированности компетенции на данном этапе ее формирования.

Сформированность уровня компетенций не ниже порогового является основанием для допуска обучающегося к промежуточной аттестации по данной дисциплине. Формой промежуточной аттестации по данной дисциплине является зачет с оценкой, оцениваемый по принятой в ФГБОУ ВО «РГРТУ» четырехбалльной системе: «неудовлетворительно», «удовлетворительно», «хорошо» и «отлично».

«ОТЛИЧНО»	<i>студент должен:</i> продемонстрировать глубокое усвоение материала; исчерпывающе, последовательно, грамотно и логически стройно изложить теоретический материал; правильно формулировать определения; уметь делать выводы по излагаемому материалу; безупречно ответить не только на вопросы билета, но и на дополнительные вопросы в рамках рабочей программы дисциплины; продемонстрировать умение правильно выполнять предусмотренные
«хорошо»	<i>студент должен:</i> продемонстрировать достаточно полное знание материала; продемонстрировать знание основных теоретических понятий; достаточно последовательно, грамотно и логически стройно изложить материал; уметь сделать достаточно обоснованные выводы; ответить на все вопросы билета; продемонстрировать умение правильно выполнять практические задания, при этом возможны не принципиальные ошибки;

«удовлетворительно»	<i>студент должен:</i> продемонстрировать общее знание материала; знать основную рекомендуемую учебную литературу; уметь строить ответ в соответствии со структурой излагаемого вопроса; показать общее владение понятийным аппаратом дисциплины; уметь устранять допущенные ошибки в ответе на теоретические вопросы и при выполнении практических заданий, либо (при неправильном выполнении практического задания) по указанию преподавателя выполнить другие практические задания того же раздела дисциплины;
«неудовлетворительно»	<i>ставится в случае:</i> незнания значительной части программного материала; не владения понятийным аппаратом; существенных ошибок при изложении учебного материала; неумения строить ответ в соответствии со структурой излагаемого вопроса; неумения делать выводы.
	Как правило, такая оценка ставится студентам, которые не могут продолжить обучение поданной образовательной программе, а также, если студент после начала экзамена отказался его сдавать, или нарушил правила сдачи экзамена (списывал, подсказывал, обманом пытался получить более высокую оценку и т.д.).

ФОРМИРУЕМЫЕ КОМПЕТЕНЦИИ

ПК-1: Способен разрабатывать требования, проектировать и выполнять программную реализацию программного обеспечения

ПК-1.1. Анализирует требования к программному обеспечению

ПК-1.3. Проектирует программное обеспечение и выполняет его программную реализацию

ПК-3: Способен разрабатывать компоненты системных программных продуктов

ПК-3.1. Разрабатывает системные утилиты программного обеспечения

ПК-3.2. Создает компоненты инструментальных средств программирования

ТИПОВЫЕ КОНТРОЛЬНЫЕ ЗАДАНИЯ ИЛИ ИНЫЕ МАТЕРИАЛЫ

4.1. Типовые задания текущего контроля для практических занятий и при защите лабораторных работ

1. Составить функцию нахождения среднего арифметического элементов вектора L . Используя данную функцию, найти средние арифметические в строках матриц $M1$ и $M2$ и сформировать из них соответственно вектора $V1$, $V2$. Написать функцию сортировки векторов $V1$ и $V2$ по возрастанию. 2. Составить функцию проверки упорядоченности символьных элементов вектора L по алфавиту. Используя данную функцию, проанализировать строки символьных матриц $M1$ и $M2$. Для строк, которые не отсортированы по алфавиту, выделить их элементы в отдельный вектор и написать функцию, удаляющую все буквы из них, с нечётным номером в алфавите. 3. Составить функцию, подсчитывающую количество слов вектора, которые начинаются и оканчиваются одной и той же буквой. Используя данную функцию, найти суммы числа слов в каждой строке в матрицах $M1$ и $M2$ и записать их все в один суммарный вектор V . Найти с помощью отдельной функции минимальный и максимальный элемент вектора V и удалить эти элементы из него.
4. Составить функцию, которая перемещает в начало вектора L все четные элементы, а в конец вектора - нечетные элементы. С использованием данной функции преобразовать все строки матриц $M1$ и $M2$. Далее написать функцию нахождения максимального и минимального элемента в матрицах $M1$ и $M2$, а после этого удалить данные элементы из них.
5. Составить функцию, проверяющую на равенство значения элементов векторов $L1$ и $L2$ и возвращающую вектор одинаковых элементов, присутствующих в обоих из них. Используя функцию, проанализировать соответствующие строки матриц $M1$ и $M2$, и из полученных векторов одинаковых элементов создать матрицу $M3$, где все несуществующие элементы заменены значением «бесконечность». Написать функцию, подсчитывающую кол-во значений «бесконечность» в матрице $M3$.
6. Составить функцию, определяющую включения вектора $L1$ в вектор $L2$ и наоборот (варианты возвращаемого значения: 0 – вектор $L1$ включает $L2$, 1 – вектор $L2$ включает $L1$, 2 – вектора полностью идентичны, 3 – вектора не включают друг друга). Используя функцию, проанализировать соответствующие строки матриц $M1$ и $M2$, составить суммарный вектор из всех не включающих друг друга строк. Написать функцию, замещающую в

данном векторе все дублирующийся элементы значением «бесконечность».

7. Составить функцию вставки элемента E после каждого элемента вектора, превышающего некоторое значение P . Используя данную функцию обработать все строки матриц $M1$ и $M2$. Определить на сколько данные матрицы увеличились, уменьшились или остались прежнего размера. После написать функцию, находящую количество элементов E с хотя бы одним чётным индексом в матрицах $M1$ и $M2$.

8. В векторе натуральных чисел переставить элементы по следующему правилу: если текущий элемент больше некоторого числа P , то поместить следующий за ним элемент в конец вектора; если текущий элемент меньше или равен числу P , перенести в начало вектора текущий элемент (первый оставить без изменения). Используя данную функцию обработать все строки матриц $M1$ и $M2$. Далее написать функцию, которая удаляет из матриц $M1$ и $M2$ строки с большим количеством отрицательных чисел, чем положительных. Определить на сколько данные матрицы увеличились, уменьшились или остались прежнего размера.

9. Написать функцию создания вектора $L1$ из вектора $L2$, расположив в нём только положительные элементы вектора $L2$ в обратном порядке. С помощью данной функции обработать строки матриц $M1$ и $M2$. Написать функцию обработки матриц, которая замещает все чётные элементы значением «бесконечность» и применить её к матрицам $M1$ и $M2$. 10.

Написать функцию определяющую, входит ли элемент E в вектор L , подсчитать количество вхождений данного элемента в вектор и вставить первый элемент вектора после каждого вхождения E . С помощью данной функции обработать строки матриц $M1$ и $M2$. После этого написать функцию удаления элемента из матрицы и с помощью неё удалить из матриц $M1$ и $M2$ элемент E . Определить на сколько данные матрицы увеличились, уменьшились или остались прежнего размера, относительно обработанных ранее матриц.

11. Написать функцию возвращающую вектор L , упорядочив его по убыванию, из четных элементов вектора $L1$ и нечетных элементов вектора $L2$. С помощью данной функции сформировать матрицу M , где каждая строка матрицы формируется из элементов строк матрицы $M1$ и строк матрицы $M2$. После написать функцию, заполняющую отсутствующие элементы матрицы M минимальным элементом данной строки в чётных строках, и максимальным элементом в нечётных строках.

12. Написать функцию, формирующую два вектора L1 и L2 из вектора L по следующему правилу: в вектор L1 занести положительные элементы, а в вектор L2 – отрицательные. С помощью данной функции сформировать построчно матрицы M1 и M2 из строк матрицы M. После написать функцию, находящую вектор минимальных элементов каждой строки матрицы M1 и максимальных элементов строк матрицы M2.

13. Описать массив структур и поместить в него сгенерированные сведения о N книгах. Предусмотреть такие сведения как название книги, жанр, дата издания (отдельный объект), количество экземпляров, ФИО автора (отдельным объектом), количество страниц. Написать функцию выдачи списка книг по фамилии автора, жанру или диапазону годов издания. Написать функцию удаления сведений о количестве страниц, если количество страниц менее заданного числа. Написать функцию добавления информации о возрасте книги, найденную по дате её издания.

14. Описать массив структур и поместить в него сгенерированные сведения о N работниках. Предусмотреть такие сведения как ФИО работника (отдельным объектом), дата рождения (отдельный объект), номер телефона, место работы (отдельный объект со сведениями о названии организации, должности и стаже). Написать функцию выдачи списка работников по названию организации, должности или диапазону стажа. Написать функцию удаления сведений о дате рождения, если стаж менее заданного числа. Написать функцию добавления информации о районе проживания работника, найденного по первым двум цифрам телефона.

15. Описать массив структур и поместить в него сгенерированные сведения о N студентах. Предусмотреть такие сведения как ФИО студента (отдельным объектом), дата поступления (отдельный объект), номер телефона, результаты сессии (отдельный массив структур с информацией о названии предметов и полученных оценках). Написать функцию удаления сведений о дате поступления, если год поступления старше заданного. Написать функцию добавления информации о среднем балле студента, найденного по оценкам сессии. Написать функцию выдачи списка студентов отсортированному по убыванию среднего балла.

4.2. Типовые контрольные задания итогового контроля

1. Описать массив структур и поместить в него сгенерированные сведения о N деталях. Предусмотреть такие сведения как наименование

детали, габаритные размеры (отдельный объект), материал, масса детали, список поставщиков деталей (отдельный массив структур из названий организации и контактного телефона). Написать функцию удаления сведений о материале, если масса детали менее указанной величины. Написать функцию добавления информации о габаритном объёме детали, найденного по габаритным размерам. Написать функцию выдачи списка деталей отсортированному по убыванию массы.

2. Описать массив структур и поместить в него сгенерированные сведения о N сданных экзаменационных сессий. Предусмотреть такие сведения сессии как номер курса, дата начала сессии (отдельный объект), дата конца сессии (отдельный объект), список предметов (массив структур со сведениями о названии предмета и полученной оценки). Написать функцию удаления сведений о номере курса, если номер сессии нечётный. Написать функцию добавления информации о средней оценке сессии, найденного по списку оценок предметов. Написать функцию выдачи списка предметов и оценок лучшей сессии и худшей сессии.

3. Описать массив структур и поместить в него сгенерированные сведения о N людях. Предусмотреть такие сведения как ФИО человека (отдельным объектом), пол, дата рождения (отдельный объект), номер телефона, адрес проживания (отдельный объект содержащий сведения о городе, улице, номере дома и номере квартиры). Написать функцию выдачи списка людей по городу, полу или диапазону годов рождения. Написать функцию удаления сведений о дате рождения, если год рождения более указанного. Написать функцию добавления информации о районе проживания работника, найденного по первым двум цифрам телефона. 4.

Описать массив структур и поместить в него сгенерированные сведения о N работниках. Предусмотреть такие сведения как ФИО работника (отдельным объектом), дата рождения (отдельный объект), номер цеха, трудовая информация (отдельный объект со сведениями о должности, разряде, стаже). Написать функцию выдачи списка работников по должности, разряду или диапазону стажа. Написать функцию удаления сведений о дате рождения, если стаж менее заданного числа. Написать функцию добавления информации о возрасте работника, найденного по году рождения.

5. Описать массив структур и поместить в него сгенерированные сведения о N сотрудниках. Предусмотреть такие сведения как ФИО

сотрудника (отдельным объектом), дата рождения (отдельный объект), должность, стаж, зарплата (отдельный объект со сведениями о окладе, премии, оплате интенсивности, оплате переработки). Написать функцию удаления сведений о дате рождения, если стаж менее заданного числа. Написать функцию добавления информации о суммарном доходе работника, найденного как сумма всей составляющей зарплаты минус 13%. Написать функцию выдачи списка работников, отсортированных по убыванию дохода.

6. Описать массив структур и поместить в него сгенерированные сведения о плане выпуска N наименований. Предусмотреть такие сведения как название изделия, шифр, единица измерения, план выпуска (отдельный объект из плана выпуска и сколько фактически выпущено), список заказчиков (отдельный массив структур из названий организации и количества закупаемого наименования). Написать функцию удаления сведений о единице измерения, если план выпуска менее заданного числа. Написать функцию добавления информации о проценте выполнения плана, найденного как соотношение фактического выпуска от плана выпуска. Написать функцию выдачи списка изделий, с перевыполнением плана, списка изделий с невыполнением плана.

7. Описать массив структур и поместить в него сгенерированные сведения о N спортсменах. Предусмотреть такие сведения как ФИО спортсмена (отдельным объектом), дата рождения (отдельный объект), страна, вид соревнования, результаты соревнований (отдельный объект со сведениями о названии соревнования, дате проведения, результате спортсмена). Написать функцию выдачи списка спортсменов по названию соревнования, стране или диапазону годов рождения. Написать функцию удаления сведений о дате рождения, если год рождения менее заданного числа. Написать функцию добавления информации о среднем результате спортсмена по всем соревнованиям.

8. Описать массив структур и поместить в него сгенерированные сведения о N футболистах. Предусмотреть такие сведения как ФИО футболиста (отдельным объектом), дата рождения (отдельный объект), количество голов, команда (отдельный объект со сведениями о названии команды, стране, дате вступления (отдельный объект), зарплате футболиста). Написать функцию выдачи списка футболистов по названию команды, стране или диапазону забитых голов. Написать функцию удаления сведений о дате рождения, если год рождения менее заданного числа. Написать

функцию добавления информации о количестве лет нахождения в команде, рассчитанной по году вступления в команду.

9. Описать массив структур и поместить в него сгенерированные сведения о инвентаризационной ведомости из N наименований.

Предусмотреть такие сведения как название наименования, инвентарный номер, дата принятия на учёт (отдельный объект), количество, место хранения (отдельный объект из полей номер корпуса, номер этажа, номер помещения). Написать функцию удаления сведений о дате принятия на учёт, если год принятия является текущим. Написать функцию добавления информации о сроке службы наименования по текущей дате и дате принятия его на учёт. Написать функцию выдачи списка наименований по номеру корпуса, номеру этажа или с указанным диапазоном сроков службы. 10.

«Комплексное число» – Complex. Разработать структуру комплексных чисел. Структура должна работать с функциями для изменения и получения значения действительной и мнимой части, для реализации операций сложения, вычитания, умножения, деления, присваивания комплексных чисел. Предусмотреть функцию toString. Создать 2 массива структур и с помощью них поэлементно показать работу всех операций. 11. «Дробь» – Fraction. Разработать структуру в виде пары целых положительных чисел (m,n) а также отдельно знак дроби. Структура должна работать с функциями для изменения и получения значения числителя и знаменателя, сложения, вычитания, умножения, деления и присваивания дробей. Предусмотреть функцию toString. Создать 2 массива структур и с помощью них поэлементно показать работу всех операций. 12.

«Вектор» – Vector. Разработать структуру вектора размерности n . Реализовать функции для изменения и получения значения компонента вектора, вычисления длины вектора, скалярного произведения, сложения, умножения, умножения на скаляр. Предусмотреть функцию toString. Создать 2 массива структур и с помощью них поэлементно показать работу всех операций.

13. «Квадратная матрица» – Matrix. Разработать структуру квадратной матрицы $n \times n$. Реализовать функции для изменения и получения значения элемента матрицы, сложения, вычитания, умножения матриц; вычисления индексов максимального и минимального элемента матрицы. Предусмотреть функцию toString. Создать 2 массива структур и с помощью них поэлементно показать работу всех операций. 14. «Многочлен» – Polynom.

Разработать структуру полинома степени n .

Реализовать функции для изменения и получения значения указанного коэффициента, вычисления значения полинома; сложения, вычитания, умножения полиномов. Предусмотреть функцию toString. Создать 2 массива структур и с помощью них поэлементно показать работу всех операций. 15. «Фигуры» – Shapes. Разработать структуру для описания плоских фигур: круг, прямоугольник, треугольник. Включить функции для получения и изменения параметров фигур, перемещения на плоскости, вращения, нахождения площади и периметра фигуры. Предусмотреть функцию toString. Выполнить тестирование модуля, создав массив структур и показав на его примере работу всех функций.

16. «Множество целых чисел» – Set. Разработать структуру множества целых чисел мощности n . Реализовать функции для определения принадлежности заданного элемента множеству, добавление\удаление элемента, пересечения, объединения, разности двух множеств. Предусмотреть функцию toString. Создать 2 массива структур и с помощью них поэлементно показать работу всех операций.

17. «Массив строк» – StringArray. Разработать структуру для представления массива строк. Реализовать функции для добавления\удаления строк, для поэлементной конкатенации двух массивов, упорядочения строк по длине, слияния двух массивов строк с удалением повторяющихся строк, а также формирование массива количества слов в каждой строке. Предусмотреть функцию toString. Создать 2 структуры и с помощью них поэлементно показать работу всех функций.

18. «Массив бит» – BitArray. Разработать структуру представляющий собой массив битов длины n . Реализовать функции для установки и получения значения бита на заданной позиции, изменения размера массива (справа и слева), сдвиг битов вправо\влево на заданное число позиций, битовые операции and и or для двух массивов. Предусмотреть функцию toString. Создать 2 массива структур и с помощью них поэлементно показать работу всех операций.

19. «Булева матрица» – BoolMatrix. Разработать структуру представляющий собой матрицу булевых значений размерности $n \times m$. Реализовать функции для изменения и получения значения указанного элемента, логического сложения, умножения и инверсии матриц. Реализовать функцию для подсчета количества true и false значений в матрице.

Предусмотреть функцию toString. Создать 2 массива структур и с помощью них поэлементно показать работу всех операций.

20. «Односвязный список» – LinkedList. Разработать структуру для работы с односвязным списком с целыми числами. Реализовать функции добавления элемента на заданную позицию, удаление всех элементов с заданным значением, получение значения по заданному индексу, объединение двух списков, разделение списка на два с указанной позиции, реверс списка.

Предусмотреть функцию toString. Создать 2 массива структур и с помощью них поэлементно показать работу всех операций.

21. «Бинарное дерево» – BinaryTree. Разработать структуру для работы с бинарным деревом, узлы которого содержат натуральные числа. Реализовать функции добавления и удаления узлов, получения массива узлов с заданным значением, определения высоты и количества листьев у дерева.

Предусмотреть функцию toString. Создать массив структур и с помощью них поэлементно показать работу всех операций.

4.3. Вопросы к зачёту с оценкой по дисциплине

1. Что такое языки программирования «высокого» и «низкого» уровня? Что такое уровни абстракции?
2. Каково назначение языка C? Зачем нужна функция main?
3. Что такое ассемблер? Что такое регистры процессора и какие их виды бывают?
4. Что такое команда пересылки и как осуществить сложение и вычитание в ассемблере?
5. Что такое команда пересылки и как производится умножение и деление в ассемблере?
6. Что такое ассемблерная вставка? Что такое расширенная ассемблерная вставка? Формат записи?
7. Формат объявления констант и переменных в языке C? Что такое область видимости в языке C? Какие области видимости бывают? 8.

Какие есть целочисленные и вещественные типы данных в языке С?

9. Что такое автоматическое приведение типов? Каковы правила автоматического приведения типов? Что такое явное преобразование типа?
10. Формат функции printf? Что такое специальные символы? Приведите примеры.
11. Формат функции scanf? Что такое спецификаторы формата? Приведите примеры.
12. Какие унарные, бинарные и тернарные операторы существуют в языке С? Приведите примеры.
13. Что такое выражение? Что такое приоритет операций? Какие существуют основные математические функции?
14. Формат функций rand и srand? Для чего они нужны? Приведите примеры использования.
15. Для чего нужны и каков формат условного оператора и оператора выбора?
16. Для чего нужны и каков формат цикла с постусловием и с предусловием?
17. Формат цикла с параметром? Когда выполняются команды, описанные у него в заголовке? Для чего нужны операторы break и continue?
18. Что такое массив? Формат объявления и инициализации массивов? Какими понятиями характеризуется массив?
19. Приведите пример объявления и инициализации одномерного и двумерного массива.
20. Что такое функция? Что такое чистая функция?
21. Формат объявления и вызова функции? Что нужно сделать, когда вызов функции предшествует её объявлению?
22. Как объявить функцию, не принимающую аргументов или не возвращающую значение? Для чего применяется ключевое слово

- inline? 23. Что такое константные параметры функции и для чего применяются?
24. Как передаётся одномерный и двумерный массив в функцию через параметр? Приведите пример.
25. Что такое указатель? Как объявляется указатель? Какие операции определены для указателей?
26. Что такое арифметика указателей? Как связаны массивы и указатели? Что такое нулевой указатель и для чего он применяется?
27. Какие существуют способы передачи параметров функции? В каких случаях какой способ необходимо выбирать?
28. Что такое константный указатель, указатель на константу и для чего он применяется? Приведите примеры объявления.
29. Как представляются одномерные и двумерные массивы в памяти компьютера?
30. Как передавать массив в функцию через указатель? Что такое пустой указатель, какие отличия от типизированного, и для чего он применяется? 31. Как объявить указатель на функцию? Для чего он может применяться?
32. На какие области делится память, выделяемая для процесса? Что такое область инициализированных и неинициализированных данных?
33. На какие области делится память, выделяемая для процесса? Что такое область стека и область кучи?
34. Что такое автоматическая память? Какие преимущества и недостатки даёт автоматическое выделение памяти?
35. Что такое динамическая память? Какие преимущества и недостатки даёт динамическое выделение памяти?
36. Что такое статическая память? Какие преимущества и недостатки существуют у статического выделения памяти?
37. Формат и назначение функции `malloc` и `free`? Как функция `free` «узнаёт» о размере освобождаемого участка памяти?
38. Формат и назначение функции `calloc` и `realloc`?

39. Формат и назначение функции `memcpy` и `memset`?
40. Как можно создать двумерный динамический массив? Что такое «утечка памяти»?
41. Что такое строковый тип данных в языке C? Как задаётся строковая константа, а как отдельный символ?
42. Как инициализировать строку через строковую константу и через массив символов? Как задаётся массив символьных строк?
43. Каков формат и назначение функций `puts` и `gets` и в чём их отличие от `printf` и `scanf`?
44. Каков формат и назначение функции `fgets`?
45. Каков формат и назначение функций `strcat` и `strcpy`?
46. Каков формат и назначение функций `strcmp` и `strlen`?
47. Каков формат и назначение функций `strset` и `sprintf`?
48. Каков формат и назначение функций `atoi`, `atof`, `isalpha`, `isdigit`?
49. Каков формат и назначение функций `strchr` и `toupper`/`tolower`?
50. Что такое параметры запуска и как их можно задавать программе?
51. Как в программе получить доступ к параметрам запуска и использовать их? Приведите пример.
52. Что такое этапы препроцессинга и компиляции программы, зачем они нужны?
53. Что такое этапы ассемблирования и компоновки программы, зачем они нужны?
54. Что такое объектный файл и для чего он нужен? Что такое таблица символов и для чего она нужна?
55. Что такое препроцессорные директивы? Каков формат и назначение директивы `#define`?
56. Что такое препроцессорные директивы? Каков формат и назначение директив `#include` и `#undef`?

57. Что такое препроцессорные директивы? Каков формат и назначение директив `#ifdef/#ifndef` и `#if/#else/#endif` ?
58. Что такое модульное программирование? Что такое модуль и как он представлен в языке C?
59. Что такое глобальные переменные? Что такое статические переменные? Приведите примеры объявления.
60. Каков формат и назначение команды `typedef`? Приведите примеры использования.
61. Что такое структура и для чего она нужна? Формат объявления структуры?
62. Как происходит копирование структуры? Как использовать указатель на структуру и как обращаться к полям указателя на структуру?
63. Какими способами можно передавать структуру в качестве аргумента функции и в чём отличие между ними? Приведите примеры.
64. Что такое перечисление и для чего оно нужно? Как можно объявить перечисление?
65. Что такое побитовая операция? Какие побитовые операции существуют в языке C? Приведите примеры.
66. Что такое битовые флаги и в каких случаях они используются? Приведите примеры.
67. Что такое битовые маски и в каких случаях они используются? Приведите примеры.
68. Что такое битовые поля, в чём их преимущество и в каких случаях они нужны?
69. Что такое объединения и для чего они нужны? Формат объявления объединения?

Основная учебная литература

1. Керниган Б., Ритчи Д. Язык программирования Си. — 2-е изд. — М.: Вильямс, 2007. — С. 304. — ISBN 0-13-110362-8.

2. Столяров А. В. Язык Си и начальное обучение программированию. — Издатель-ский отдел факультета ВМК МГУ, 2010. — № 7. — С. 78—90.
3. Подбельский В. В., Фомин С. С. Курс программирования на языке Си: учебник. — М.: ДМК Пресс, 2012. — 318 с. — ISBN 978-5-94074-449-8.
4. Прата С. Язык программирования С: Лекции и упражнения. — М.: Вильямс, 2006. — С. 960. — ISBN 5-8459-0986-4.
5. Прата С. Язык программирования С (С11). Лекции и упражнения, 6-е издание. — М.: Вильямс, 2015. — 928 с. — ISBN 978-5-8459-1950-2.

Дополнительная литература

6. Шилдт Г. С: полное руководство, классическое издание. — М.: Вильямс, 2010. — С. 704. — ISBN 978-5-8459-1709-6.
7. Гукин Д. Язык программирования Си. — М.: Диалектика, 2006. — С. 352. — ISBN 0-7645-7068-4.
8. Языки программирования Ада, Си, Паскаль / А. Фьюэр, Н. Джахани. — М.: Радио и Связь, 1989. — 368 с. — 50 000 экз. — ISBN 5-256-00309-7.

Перечень ресурсов информационно–телекоммуникационной сети Интернет для самостоятельной работы

1. Электронно-библиотечная система «Лань». – Режим доступа: с любого компьютера РГПТУ без пароля. – URL: <https://e.lanbook.com/>
2. Электронно-библиотечная система «IPRbooks». – Режим доступа: с любого компьютера РГПТУ без пароля, из сети Интернет по паролю. – URL: <https://iprbookshop.ru/>.
3. Электронная библиотека РГПТУ. – URL: <http://weblib.rrtu/ebs>.
4. Научная электронная библиотека eLibrary. – URL: <http://e.lib/vlsu.ru/www.uisrussia.msu.ru/elibrary.ru>
5. Библиотека и форум по программированию. – URL: <http://www.cyberforum.ru>

6. Национальный открытый университет ИНТУИТ. –
URL:
<http://www.intuit.ru/>

7. Информационно-справочная система. – URL: <http://window.edu.ru>
Оценочные материалы составлены в соответствии с рабочей
программой дисциплины Б1.В «Низко-уровневое
программирование» по направлению подготовки 09.03.04
«Программная инженерия» (уровень бакалавриата).

Составил: к.т.н., доцент кафедры
«Вычислительная и прикладная математика»
Антипов О.В.

Заведующий кафедрой вычислительной и
прикладной математики, д-р техн. наук, профессор
Овечкин Г.В.