

Министерство науки и высшего образования Российской Федерации
НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ (НИ ТГУ)

Механико-математический факультет

УТВЕРЖДЕНО:

Декан

Л. В. Гензе

Оценочные материалы по дисциплине

Программирование

по направлению подготовки / специальности

01.03.01 Математика

02.03.01 Математика и компьютерные науки

01.03.03 Механика и математическое моделирование

Направленность (профиль) подготовки/ специализация:

Современная математика и математическое моделирование

Вычислительная математика и компьютерное моделирование

Теоретическая, вычислительная и экспериментальная механика

Форма обучения

Очная

Квалификация

Математик. Преподаватель / Математик. Аналитик / Математик. Исследователь

Математик. Преподаватель / Математик. Вычислитель /

Исследователь в области математики и компьютерных наук

Механик. / Механик. Исследователь

Год приема

2024, 2025

СОГЛАСОВАНО:

Руководитель ОП

Л.В.Гензе

Председатель УМК

Е.А.Тарасов

Томск – 2024

1. Компетенции и индикаторы их достижения, проверяемые данными оценочными материалами

Целью освоения дисциплины является формирование следующих компетенций:

ОПК-2 Способен разрабатывать, анализировать и внедрять математические модели в современной науке и технике, экономике и управлении.

ОПК-3 Способен разрабатывать и реализовывать алгоритмы и компьютерные программы, пригодные для практического применения.

Результатами освоения дисциплины являются следующие индикаторы достижения компетенций:

РООПК-2.1 Умеет обоснованно выбрать тип математической модели для формализации решаемой задачи

РООПК-2.2 Применяет стандартные и типовые действия при построении математической модели определенного типа

РООПК-2.3 Применяет подходы визуализации и представления результатов математического моделирования для апробации и демонстрации в виде отчетов, презентаций и научных текстов

РООПК-3.1 Владеет теоретическими основами программирования и алгоритмизации

РООПК-3.2 Способен реализовывать алгоритмы на языках программирования высокого уровня

РООПК-3.3 Способен разрабатывать программные алгоритмы при решении задач методами математики и механики

2. Оценочные материалы текущего контроля и критерии оценивания

Элементы текущего контроля:

- тесты;
- контрольная работа.

Примерный перечень тестовых заданий (РООПК-3.1, 3.2, 3.3)

1. Алгоритм - это...
 - a. последовательность действий над данными
 - b. процесс решения задачи
 - c. порядок действий над допустимым набором исходных данных, приводящий к верному результату за конечное время
 - d. любой набор инструкций
2. Язык программирования это...
 - a. набор правил записи программ
 - b. нет верного ответа
 - c. набор знаков для описания действий
 - d. формальный язык, предназначенный для записи компьютерных программ
3. Константа - это...
 - a. последовательность цифр
 - b. неизменяемый набор символов
 - c. значение, которое не может быть изменено
 - d. служебное слово языка C++
4. Глобальная переменная видна...
 - a. всем функциям программы
 - b. всем функциям, описанным в одном с ней файле
 - c. только функции main независимо от локализации описания
 - d. только функциям, описанным в одном файле с main

5. Значение глобальной переменной можно изменить...
 - a. только в функции main независимо от локализации описания
 - b. в любой функции программы
 - c. в любой функции, описанной в одном с ней файле
 - d. только в функциях, описанных в одном файле с main
6. Определение функции в C++ имеет вид:
 - a. тип имя(список параметров);
 - b. тип имя(){тело}
 - c. тип имя(список параметров){тело}
 - d. имя(список параметров)
7. В языке C++ не допускается определение функции...
 - a. в отдельном файле
 - b. в теле другой функции
 - c. в одном файле с функцией main
 - d. в одном файле с другой функцией
8. Унарная операция «*»...
 - a. не определена в C++
 - b. применима к операнду любого типа данных, извлекает значение операнда
 - c. применима только к указателю, извлекает значение объекта, на который указывает операнд
 - d. применима к операнду любого типа данных, получает адрес операнда
9. Бинарная операция «==»...
 - a. не определена в C++
 - b. выполняет присваивание левому операнду значения правого
 - c. выполняет сравнение операндов
 - d. производит 1, если значения операндов совпадают, и 0 в противном случае
10. Какой оператор возвращает значение из функции в языке программирования Си++?
 - a. return
 - b. end
 - c. out
 - d. break

Ключи: 1 c), 2 d), 3 c), 4 a), 5 b), 6 c), 7 b), 8 c), 9 d), 10 a).

Примерный перечень вариантов (заданий) контрольных работ (РООПК-2.1, 2.2, 2.3)

1. Определить пользовательские функции min() и max() и найти значение выражения: $y = \min(a, b, c) / \max(a, b, c)$.
2. Дан файл f, элементы которого являются действительными числами. Найти модуль суммы и квадрат произведения элементов файла f.
3. Написать программу для реализации трех алгоритмов сортировки одномерного массива (сортировка выбором, сортировка «пузырек», сортировка вставками). Каждый алгоритм должен быть оформлен в виде функции.
4. Сформировать массив a[n], элементы которого выбираются случайным образом из интервала [10,90]. Определить методом последовательного и бинарного поиска, содержит ли он заданное число. Если элемент найден, то удалить его из массива.
5. Организовать линейный односвязный список. Для решения задачи реализовать все необходимые функции (создание списка, добавление элемента в список, печать элементов списка, удаление элементов и т.д.).

Критерии оценивания:

Результаты контрольной работы определяются оценками «зачтено», «незачтено».

Оценка «зачтено» выставляется, если задание решено без ошибок.

Оценка «незачтено» выставляется, если при решении задания допущены ошибки.

3. Оценочные материалы итогового контроля (промежуточной аттестации) и критерии оценивания

Экзамен в первом семестре проводится в устной форме по билетам. Билет содержит два теоретических вопроса и задачу. Продолжительность экзамена 1,5 часа. Билет проверяет РООПК-3.1, РООПК-3.2, РООПК-3.3. Ответы на вопросы билета даются путем выбора из списка предложенных.

Примерный перечень теоретических вопросов на экзамене:

1. История вычислительной техники (от абака до ЭВМ). Закон Мура. Классы современных компьютеров. Суперкомпьютеры. Кластер ТГУ Cyberia.
2. Архитектура ЭВМ фон Неймана. Основные устройства. ЭВМ с общей и распределенной памятью.
3. Основная память ЭВМ, ее структура. Представление информации в памяти и единицы измерения.
4. Системы счисления (позиционная и непозиционная). Перевод числа из одной системы счисления в другую.
5. Система элементарных команд процессора (операции, параметры операций).
6. Архитектура персонального компьютера. Системный блок и периферийные устройства.
7. Определение алгоритма и его свойства. Пример (алгоритм Евклида).
8. Способы записи алгоритмов.
9. Простейшие алгоритмы (блок-схемы): сложение 2-х чисел, поиск максимального из 2-х чисел, суммирование n чисел.
10. Блок-схемы алгоритмов с рекуррентными последовательностями: отрезки рядов, числа Фибоначчи.
11. Алгоритмический язык и его составляющие: алфавит, синтаксис, семантика. Компиляция и выполнение.
12. Способы описания синтаксиса языка программирования (БНФ и синтаксические диаграммы).
13. Основные понятия языка программирования: оператор, идентификатор, описание, функция.
14. Основные понятия языка программирования: константа, переменная. Обмен значениями.
15. Язык C/C++. История появления, достоинства и недостатки. Структура программы на C. Идентификаторы.
16. Базовые скалярные типы и операции над ними в ЯПВУ и в C. Примеры значений, операции, хранение в памяти.
17. Форматированный ввод/вывод на C.
18. Простой оператор присваивания на C. Арифметические операции. Математические функции. Преобразование типов.
19. Операции инкремента и декремента в C. Составные операторы присваивания в C.
20. Базовые управляющие конструкции (альтернатива) в C. Логические операции. Логический тип. Примеры.
21. Базовые управляющие конструкции (многозначное ветвление и цикл с предусловием) в C. Примеры. Проверка числа на простоту.
22. Базовые управляющие конструкции (цикл с постусловием и с параметром) в C. Примеры. Рекомендации по выбору цикла.
23. Тестирование и отладка программ. Тестирование как метод демонстрации ошибок в программе. Типы программных ошибок.
24. Методы отладки программ.

25. Операциональный подход программирования. Операторы условного и безусловного перехода. История появления структурного программирования.
26. Назначение структурного программирования. Требования к “хорошим” программам.
27. Принципы структурного программирования.
28. Структурная теорема Бома-Джакопини, ее следствия. Основное свойство “хороших” структур управления.
29. Методы структурного программирования. Пример.
30. Правила оформления текста программ в соответствии с принципами структурного программирования.
31. Теоретическое доказательство правильности программ.
32. Конструирование типов. Одномерные массивы. Описание и особенности работы с ними на С. Примеры.
33. Работа с массивами. Вычисление максимального и суммы элементов массива. Запись элементов массива в обратном порядке.
34. Представление массивов в памяти ЭВМ. Функция sizeof. Примеры. Эквивалентность массивов и указателей.
35. Многомерные массивы на С. Примеры. Вложенные циклы.
36. Поточковый ввод/вывод на С++. Структура программы на С++ с потоковым вводом/выводом.
37. Представление чисел в памяти ЭВМ с фиксированной и плавающей точкой. Арифметические операции для чисел с фиксированной и плавающей точкой. Отличие «компьютерных» чисел от классических математических.
38. Порядок. Сортировка массивов без использования дополнительных. Оценка сложности алгоритмов.
39. Поиск. Линейный и двоичный поиск элемента в массиве. Оценка сложности алгоритмов.
40. Статические и динамические структуры данных. Ссылочная реализация.
41. Указатели, ссылки, разыменование указателей на С. Арифметика указателей.
42. Динамические переменные на С. Создание, использование и удаление динамических переменных.
43. Динамические массивы их создание, использование и удаление. Пример.
44. Строки на С. Объявление строк. Работа со строками в С. Примеры функций из stdlib.h.
45. Структуры. Объявление структурного типа и экземпляра структуры. Работа с полями экземпляра структуры. Указатель на экземпляр структуры. Операция «->».
46. Файлы на С. Объявление файловой переменной, открытие и закрытие файла, чтение и запись из файла. Функция feof.
47. Файлы на С++. Объявление файловой переменной, открытие и закрытие файла, чтение и запись из файла.

Примеры задач на экзамене:

1. Написать программу на языке С «обращения» целочисленного массива: изменения порядка следования компонентов на противоположный. {1,2,5,6,4} -> {4,6,5,2,1}
2. Дано вещественное число X и целое N . Составить программу на языке С для вычисления X^N без использования математических функций. Значения X и N ввести с клавиатуры.
3. В одномерном целочисленном массиве определить минимальный элемент и его положения (номера индекса). Составить программу на языке С. Массив вводится с клавиатуры.
4. Написать программу на языке С, подсчитывающую количество и сумму цифр натурального числа. Число вводится с клавиатуры.

5. Написать программу на языке C, определяющую количество смен знаков в целочисленном массиве, не содержащем «0»-х значений. Ввод значений массива осуществить с клавиатуры.
6. Дан одномерный целочисленный массив. Найти количество различных значений элементов этого массива. Например, для {0,0,1,0,2,1,0} должно быть напечатано 0 – 4, 1 – 2, 2 – 1.

Результаты экзамена определяются оценками «отлично», «хорошо», «удовлетворительно», «неудовлетворительно». Экзамен по курсу проводится устно. Экзаменационный билет состоит из двух теоретических вопросов и одной задачи. Б Максимальная возможная оценка 5 баллов. При ответе на вопрос оценивается полнота и точность ответа, логичность и аргументированность изложения материала, умения использовать в ответе фактический материал (Таблица 2).

Таблица 2

Оценка	Критерии соответствия
5	Правильно и развернуто изложен материал каждого вопроса и отчетов по индивидуальным заданиям соответствующего раздела. Студент полно, четко и логично излагает материал вопроса и защищаемый материал задания.
4	Правильно изложен материал каждого вопроса и отчетов по индивидуальным заданиям, но не весь материал изложен развернуто и логически структурировано.
3	В целом правильно изложен материал каждого вопроса и защищаемого отчетов по заданию, но изложение носит поверхностный характер и с нарушением логики изложения.
2	Материал ответа на каждый вопрос и защищаемых отчетов по заданиям представлен очень поверхностно и с нарушением логики изложения. Студент очень плохо владеет основными концепциями дисциплины. Допущены существенные терминологические и фактические ошибки.
1	Неверно изложен материал на вопросы билета, каждый отчет по индивидуальным заданиям написан с грубыми ошибками или отчеты вообще не подготовлены к защите.

Во втором семестре проводится зачет. Билет проверяет РООПК-2.1, РООПК-2.2, РООПК-2.3. На зачете студент выбирает билет втемную из предложенных и дает ответы на вопросы билета.

Вопросы к зачету за второй семестр:

1. Эффективность и сложность алгоритмов.
2. Рекурсивные функции. Примеры. Правила, которые нужно соблюдать при написании рекурсивных функций.
3. Алгоритм поиска с возвратом на примере поиска пути в дорожной сети.
4. Задачи поиска, вставки и удаления информации в неупорядоченном массиве, вставка и удаление. Оценка сложности.
5. Задачи поиска, вставки и удаления информации в упорядоченном массиве. Оценка сложности.
6. Постановка задачи сортировки массива.
7. Сортировка массива методом «пузырька». Оценка сложности.
8. Сортировка массива методом простого включения. Оценка сложности.
9. Сортировка массива методом прямого выбора. Оценка сложности.
10. Быстрая сортировка массива. Оценка сложности.
11. Хэширование как метод размещения данных в массиве.

Зачет по курсу за второй семестр проводится устно. Экзаменационный билет состоит из двух теоретических вопросов. При ответе на вопрос оценивается полнота и

точность ответа, логичность и аргументированность изложения материала, умения использовать в ответе фактический материал (Таблица 3).

Таблица 3. Система критериев при оценивании ответов на зачете

Критерии соответствия	Оценка
Содержание ответа являются полными. Студент правильно понимает терминологию. Демонстрирует умение понимать, доказательно и логически связно отвечать на вопросы.	зачтено
Неполное, логически противоречивое изложение ответов. Студент не понимает и неправильно использует терминологию. Не может доказательно и логически связно отвечать на вопросы.	не зачтено

Вопросы к экзамену за третий семестр

1. Оператор описания типа. Примеры использования.
2. Понятие динамической структуры данных. Логический и физический уровни описания динамических структур.
3. Стек как динамическая структура. Логическое описание стека.
4. Физическая реализация стека с помощью массива.
5. Физическая реализация стека с помощью динамических переменных.
6. Очередь как динамическая структура. Логическое описание очереди.
7. Физическая реализация очереди с помощью динамических переменных.
8. Линейный однонаправленный список (ЛОС). Логическое описание ЛОС.
9. Физическая реализация ЛОС с помощью динамических переменных.
10. Деревья как динамическая структура данных. Основные определения, связанные с понятием дерева: корень, потомки, лист, степень дерева, глубина элемента и дерева в целом. Связь между количеством элементов в дереве, его степенью и глубиной.
11. Обходы деревьев и их разновидности.
12. Двоичные деревья. Обходы двоичных деревьев. Двоичные деревья поиска – определение.
13. Физическая реализация двоичных деревьев с помощью динамических переменных. Функции, с помощью которых осуществляются обходы двоичных деревьев.
14. Двоичные деревья поиска - алгоритмы поиска и вставки элементов; функции для реализации этих алгоритмов.
15. Двоичные деревья поиска – алгоритм удаления элемента (словесное описание или описание на уровне блок-схемы).
16. Основные принципы объектно-ориентированного программирования. Структура описания класса в языке C++. Уровни доступа к полям и методам объектов.
17. Конструкторы. Конструктор копирования.
18. Методы класса, их описание и использование.
19. Определение операций над объектами класса.
20. Шаблоны классов.
21. Понятие дружественных для класса функций и операций; их описание.
22. Принципы наследования в объектно-ориентированном программировании.
23. Понятие виртуальных методов класса.
24. Стандартный класс **string** в языке C++
25. Поточные классы в C++. Стандартные потоки **cin** и **cout**. Использование манипуляторов для форматирования данных при вводе-выводе.
26. Файловые потоки. Основные операции с файловыми потоками: открытие, закрытие, ввод-вывод.
27. Методы неформатированного чтения и записи в поток.

Экзамен по курсу за третий семестр проводится устно. Экзаменационный билет состоит из двух теоретических вопросов и одной задачи. При ответе на вопрос оценивается полнота и точность ответа, логичность и аргументированность изложения материала, умения использовать в ответе фактический материал (Таблица 3). Билет проверяет РООПК-3.1, РООПК-3.2, РООПК-3.3. На экзамене студент выбирает билет втемную из предложенных и дает ответы на вопросы билета.

4. Оценочные материалы для проверки остаточных знаний (сформированности компетенций)

Для проверки остаточных знаний предлагается билет из трех вопросов, которые проверяют остаточные знания по программированию. Всего предлагается 24 варианта билетов.

Вариант 1

1. Архитектура вычислительной машины фон Неймана. Ее составные элементы. (ИОПК 2.1)
 2. Перечислите функции доступа из функциональной спецификации стека. (ИОПК 2.1, ИОПК 2.2, ИОПК 2.3)
 3. Перевести число из одной системы счисления в другую: $(123.71)_8 = (\dots)_2$. (ИОПК 7.1, ИОПК 7.2)
-

Вариант 2

1. Алгоритм и его свойства. Что значит такое свойство алгоритма как "массовость"? (ИОПК 2.1)
 2. Ссылочная реализация: указатели, ссылочный тип, переменные с указателем. (ИОПК 2.1, ИОПК 2.2, ИОПК 2.3)
 3. Написать блок-схему решения задачи: По введенному натуральному числу определить правильный вариант представления возраста человека «год», «года», «лет». (ИОПК 7.1, ИОПК 7.2)
-

Вариант 3

1. Способы описания алгоритмов. (ИОПК 2.1)
 2. Очередь. Функциональная спецификация. Дек. (ИОПК 2.1, ИОПК 2.2, ИОПК 2.3)
 3. Архитектура персонального компьютера. Внешние устройства. (ИОПК 7.1, ИОПК 7.2)
-

Вариант 4

1. Основные понятия языка программирования: оператор, идентификатор, описание, переменная. (ИОПК 2.1)
 2. Почему появились языки программирования высокого уровня? Как выполняются программы, написанные на ЯП высокого уровня? (ИОПК 2.1, ИОПК 2.2, ИОПК 2.3)
 3. Написать программу нахождения всех трехзначных чисел, сумма цифр которых равна 17. (ИОПК 7.1, ИОПК 7.2)
-

Вариант 5

1. Какие способы описания синтаксиса языков программирования вы знаете? В чем заключается преимущество синтаксических диаграмм? (ИОПК 2.1)

2. Описать правило перевода целых чисел из одной системы счисления в другую. (ИОПК 2.1, ИОПК 2.2, ИОПК 2.3)
 3. На языке С-псевдокода описать алгоритм решения задачи: Пусть x и y – действительные корни квадратного уравнения. Вычислить его коэффициенты. (ИОПК 7.1, ИОПК 7.2)
-

Вариант 6

1. Структурная теорема Бема-Якопини, ее следствия. Основное свойство “хороших” структур управления. (ИОПК 2.1)
 2. Можно ли вещественные числа представлять в памяти ЭВМ с фиксированной точкой? (ИОПК 2.1, ИОПК 2.2, ИОПК 2.3)
 3. Перевести число из одной системы счисления в другую: $(ffa.2)_{16} = (\dots)_4$ (ИОПК 7.1, ИОПК 7.2)
-

Вариант 7

1. Правила оформления текста программ в соответствии с принципами структурного программирования. (ИОПК 2.1, ИОПК 2.2, ИОПК 2.3)
 2. Чем отличаются функции доступа в функциональной спецификации структуры данных? (ИОПК 2.1)
 3. Написать программу решения задачи: Присвоить целой переменной первую цифру из дробной части положительного вещественного числа x . (ИОПК 7.1, ИОПК 7.2)
-

Вариант 8

1. Алгоритмический язык и его составляющие: алфавит, синтаксис, семантика. (ИОПК 2.1)
 2. Линейный однонаправленный связанный список. Определение. (ИОПК 2.1, ИОПК 2.2, ИОПК 2.3)
 3. Единицы измерения информации в компьютере. Что такое бит? Сколько байт в килобайте? (ИОПК 7.1, ИОПК 7.2)
-

Вариант 9

1. Понятие исполнителя, объекты его действия. Программа. Программирование. Пример. (ИОПК 2.1)
 2. Какие виды памяти используются в ЭВМ. Их назначение. (ИОПК 2.1, ИОПК 2.2, ИОПК 2.3)
 3. Перевести число из одной системы счисления в другую: $(a91.f)_{16} = (\dots)_2$. (ИОПК 7.1, ИОПК 7.2)
-

Вариант 10

1. Основные свойства “хороших” программ. Назначение, принципы и методы структурного программирования. (ИОПК 2.1)
2. Перечислить единицы представления информации в памяти ЭВМ. (ИОПК 2.1, ИОПК 2.2, ИОПК 2.3)
3. Написать блок-схему решения задачи: Дано четыре числа. Найти наибольшее. (ИОПК 7.1, ИОПК 7.2)

Вариант 11

1. Способы изображения алгоритмов. В чем заключается преимущество программного способа представления алгоритма? (ИОПК 2.1)
 2. Каковы характерные особенности таких структур данных как массивы? Как осуществляется их описание и работа с ними? (ИОПК 2.1, ИОПК 2.2, ИОПК 2.3)
 3. Выберите, на ваш взгляд, правильное выражение: $p^*:=NULL$; $*p:=NULL$; $p=NULL$; (ИОПК 7.1, ИОПК 7.2)
-

Вариант 12

1. Конструирование типов. Структуры. Оператор «стрелка»: $->$. (ИОПК 2.3)
 2. В чем заключаются назначение и основные принципы структурного программирования? (ИОПК 2.1, ИОПК 2.2)
 3. Сортировка массива «на том же месте» прямым включением. Описание алгоритма. (ИОПК 7.1, ИОПК 7.2)
-

Вариант 13

1. Конструирование типов. Массивы. (ИОПК 2.1, ИОПК 2.2, ИОПК 2.3)
 2. Представление чисел в памяти ЭВМ с фиксированной и плавающей точкой. Отличие «компьютерных» чисел от классических математических. (ИОПК 2.1, ИОПК 2.2)
 3. Выражение цикла со счетчиком через цикл с предусловием. (ИОПК 7.1, ИОПК 7.2)
-

Вариант 14

1. Сортировка массивов «на том же месте» прямым выделением и прямым обменом. Описание алгоритма. (ИОПК 7.1, ИОПК 7.2)
 2. Как следует организовывать в программировании вычисление суммы элементов последовательности, определение максимального или минимального значения в числовом массиве данных? (ИОПК 7.1, ИОПК 7.2)
 3. Динамические структуры данных – деревья. Определение и использование. (ИОПК 2.1, ИОПК 2.2, ИОПК 2.3)
-

Вариант 15

1. Метки и оператор безусловного перехода. Дисциплина использования. (ИОПК 2.1, ИОПК 2.2, ИОПК 2.3)
 2. Варианты циклов. Их особенности, сходство и различия. (ИОПК 7.1, ИОПК 7.2)
 3. Понятие динамических структур данных. Файлы: реализация и методы доступа. Файлы – это динамические или статические структуры данных? (ИОПК 2.1, ИОПК 2.2, ИОПК 2.3)
-

Вариант 16

1. Определение и вызов подпрограммы, передача информации через аппарат формальных и фактических параметров. В чем отличие глобальных и локальных параметров? (ИОПК 2.1, ИОПК 2.2, ИОПК 2.3)
2. Двоичное дерево – определение. (ИОПК 2.1, ИОПК 2.2, ИОПК 2.3)

3. Дана строка из n символов. Предполагается, что в строке есть хотя бы одна запятая. Найти положение (номер) в строке первой и последней по порядку запятой. Написать блок-схему и составить программу. (ИОПК 7.1, ИОПК 7.2)
-

Вариант 17

1. Функции на языке С. (ИОПК 2.1, ИОПК 2.2, ИОПК 2.3)
 2. Зачем нужна функция delete() на С? (ИОПК 2.1, ИОПК 2.2, ИОПК 2.3)
 4. Выражение цикла с предусловием через цикл с постусловием. (ИОПК 7.1, ИОПК 7.2)
-

Вариант 18

1. Файловая система как часть операционной системы. Зачем файлу имя и расширение? (ИОПК 2.1, ИОПК 2.2, ИОПК 2.3)
 5. Описать правило перевода правильных дробей из одной системы счисления в другую. Дать примеры. (ИОПК 7.1, ИОПК 7.2)
 2. Какой способ передачи информации между программными модулями реализуют глобальные переменные? (ИОПК 2.1, ИОПК 2.2, ИОПК 2.3)
-

Вариант 19

1. Стек. Определение и функциональная спецификация. (ИОПК 2.1, ИОПК 2.2, ИОПК 2.3)
 6. Для чего используется операция “->” на языке С? (ИОПК 7.1, ИОПК 7.2)
 2. Что же все-таки эффективнее использовать: рекурсию или итерацию? Почему? (ИОПК 2.1, ИОПК 2.2, ИОПК 2.3)
-

Вариант 20

1. Структуры. Назначение и особенности использования. (ИОПК 2.1, ИОПК 2.2, ИОПК 2.3)
 7. Для алгоритма решения квадратного уравнения дать характеристику свойств алгоритма. (ИОПК 7.1, ИОПК 7.2)
 2. Ссылочная реализация: порождение и уничтожение динамических объектов. (ИОПК 2.1, ИОПК 2.2, ИОПК 2.3)
-

Вариант 21

8. Какие варианты циклов вам известны? Их особенности, сходство и различия. (ИОПК 7.1, ИОПК 7.2)
 1. Какие операции разрешены для переменных ссылочного типа? (ИОПК 2.1, ИОПК 2.2, ИОПК 2.3)
 2. Что такое пиксел? (ИОПК 2.1, ИОПК 2.2, ИОПК 2.3)
-

Вариант 22

3. 1. Операционная система, назначение и основные функции. Виды ОС и их состав. (ИОПК 2.1, ИОПК 2.2, ИОПК 2.3)
4. 2. Перечислить основные правила оформления хорошо структурированных текстов программ. (ИОПК 2.1, ИОПК 2.2, ИОПК 2.3)

9. 3. В каком случае переменные ссылочного типа равны? (ИОПК 7.1, ИОПК 7.2)
-

Вариант 23

5. Способы передачи информации между программными модулями. Как они реализуются на языке С? (ИОПК 2.1, ИОПК 2.2, ИОПК 2.3)
6. Что более абстрактно и почему: функциональная спецификация структуры данных или ее физическое представление? (ИОПК 2.1, ИОПК 2.2, ИОПК 2.3)
10. Выражение цикла с предусловием через цикл со счетчиком. (ИОПК 7.1, ИОПК 7.2)
-

Вариант 24

11. Правила использования файлов на языке С. Что такое текстовый файл? (ИОПК 7.1, ИОПК 7.2)
7. Может ли переменная с указателем иметь имя «А»? Почему? (ИОПК 2.1, ИОПК 2.2, ИОПК 2.3)
8. Представление в памяти ЭВМ чисел с фиксированной точкой. Достоинства. (ИОПК 2.1, ИОПК 2.2, ИОПК 2.3)
-

Информация о разработчиках

Старченко Александр Васильевич, профессор, доктор физико-математических наук.
Гольдин Виктор Данилович, доцент, кандидат физико-математических наук
Зюзьков Валентин Михайлович, доцент, кандидат физико-математических наук
Лаева Валентина Ивановна, старший преподаватель
Данилкин Евгений Александрович, доцент, кандидат физико-математических наук
Шельмина Елена Александровна, доцент, кандидат физико-математических наук
Стребкова Екатерина Александровна, старший преподаватель
Михайлов Михаил Дмитриевич, старший преподаватель