

Экзаменационные вопросы по дисциплине
«Алгоритмизация и программирование»
(1 семестр 2021/2022 уч.г.)

1. История развития ЭВМ.
2. Понятие архитектуры компьютера. Принципы фон Неймана.
3. Состав и назначение программного обеспечения (ПО).
4. Технология создания программ.
5. Классификация языков программирования. Схема развития алгоритмических языков.
6. Алгоритм и его свойства.
7. Способы описания алгоритмов. Схемы алгоритмов.
8. Разновидности структур алгоритмов. Теорема Бёма-Якопини. Линейные структуры алгоритмов. Пример программы линейной структуры.
9. Разветвляющиеся структуры алгоритмов. Пример программы разветвляющейся структуры.
10. Циклические структуры алгоритмов. Циклы, управляемые условием "пока".
11. Циклические структуры алгоритмов. Циклы, управляемые условием "до".
12. Циклические структуры алгоритмов. Цикл с параметром.
13. Состав языка. Способы описания синтаксиса языка.
14. История создания C и C++.
15. Структура и конструкция программы на Си/C++. Идентификатор.
16. Ключевые слова. Константы.
17. Типы данных. Концепция типов данных. Целый тип (**int**).
18. Типы данных. Концепция типов данных. Символьный тип (**char**).
19. Типы данных. Концепция типов данных. Логический тип (**bool**).
20. Типы данных. Концепция типов данных. Типы с плавающей точкой.
21. Типы данных. Концепция типов данных. Соотношение размеров основных типов данных. Тип **void**.
22. Структура программы. Пример. Переменная.
23. Выражения. Классы памяти.
24. Операторы. Операции инкремента и декремента. Операция **sizeof**.
25. Операторы – относительные приоритеты. Побитовые операторы. Операторы присваивания.
26. Преобразования типов данных.
27. Приоритеты операций и порядок вычислений.
28. Базовые конструкции структурного программирования. Операторы ветвления **if-else** и **?:**. Пример программы.
29. Пример вычисления функции, определяемой многими условиями.
30. Базовые конструкции структурного программирования. Оператор ветвления **switch**. Пример программы.
31. Операторы цикла **while**. Пример программы.
32. Операторы цикла **for**. Оператор «.». Пример программы.
33. Операторы цикла **do-while**. Пример программы.
34. Прерывание циклов. Пример программы.
35. Операторы передачи управления **goto**. Пример программы.
36. Прерывание циклов. Оператор **return**.
37. Указатели. Виды указателей. Операции над указателями. Пример программы.
38. Указатели и аргументы функций.
39. Ссылки. Правила работы над ссылками.
40. Функции, объявление и определение. Пример.
41. Параметры функции. Формальные и фактические параметры.
42. Параметры функции. Механизм передачи параметров и возврата значения.
43. Глобальные и локальные переменные. Пример на область действия имен (операция **::**).

44. Параметры функции. Виды параметров. Механизм возврата и пример.
45. Параметры со значениями по умолчанию. Функции с переменным числом параметров. Примеры.
46. Параметры функции. Аргументы в командной строке. Пример.
47. Одномерные статические массивы. Объявление, инициализация, печать.
48. Одномерные статические массивы. Поиск минимального элемента.
49. Одномерные статические массивы. Поиск индекса минимального элемента.
50. Одномерные статические массивы. Поиск элемента (линейный поиск и поиск с барьером).
51. Одномерные статические массивы. Поиск элемента **binsearch**.
52. Динамические массивы. Управление памятью.
53. Указатели и массивы (функция **strlen**).
54. Адресная арифметика.
55. Символьные указатели (функция **strcpy**).
56. Символьные указатели (функция **strcmp**).
57. Основные методы сортировки. Требования к алгоритму сортировки.
58. Сортировка методом вставки. Пример.
59. Сортировка методом прямого выбора. Пример.
60. Сортировка методом прямым обменом. Пример.
61. Двумерные динамические массивы. Выделение и освобождение памяти и обращение к элементу массива.
62. Многомерные статические массивы. Инициализация и обращение к элементу массива.
63. Массивы указателей. Пример описания. Разница между двумерным массивом и массивом указателей.
64. Стандартный ввод-вывод (**getchar**).
65. Стандартный ввод-вывод (**putchar**).
66. Форматный вывод (**printf**).
67. Форматный ввод (**scanf**).
68. Ввод-вывод (**gets**, **puts**).
69. Типы данных, определяемые пользователем **typedef**.
70. Типы данных, определяемые пользователем **enum**.
71. Типы данных, определяемые пользователем **struct**.
72. Типы данных, определяемые пользователем **struct**. Функция сравнения двух дробей.
73. Типы данных, определяемые пользователем **struct**. Функция сложения двух дробей.
74. Типы данных, определяемые пользователем **union**, битовые поля.
75. Массивы структур. Функция, определяющая средний рост мужчин.
76. Массивы структур. Функция, определяющая имя самого низкого мужчины из группы.
77. Массивы структур. Логическая функция поиска людей одного роста.
78. Вложенные циклы, пример.
79. Позиционные системы счисления. Расширенная форма записи числа.
80. Перевод целых чисел из десятичной системы счисления в иные системы.
81. Перевод дробных чисел из десятичной системы счисления в иные системы.
82. Выполнение арифметических операций в различных системах счисления.
83. Перевод чисел, представленных в системах счисления с основаниями 2, 8, 16.
84. Выполнение арифметических операций в различных системах счисления.
85. Дополнительный код числа.
86. Представление целых чисел в формате с фиксированной точкой.
87. Представление вещественных чисел в формате IBM-360 с плавающей точкой
88. Представление вещественных чисел в формате IEEE-754 с плавающей точкой
89. Кодирование текстовой информации: 8-битовые таблицы.
90. Кодирование текстовой информации: юникод.