**Вопросы для самоподготовки к экзамену**

**по дисциплине**

**«Основы алгоритмизации и программирования»**

*Специальность 09.02.07 «Информационные системы и программирование»*

1. Определение алгоритма.
2. Как классифицируются языки программирования?
3. Что характерно для программы, написанной с применением императивной парадигмы программирования?
4. Что характерно для программы, написанной с применением декларативной парадигмы программирования?
5. Что характерно для программы, написанной с применением процедурной парадигмы программирования?
6. Чем отличаются языки низкого уровня от языков высокого уровня?
7. Что характерно для программы, написанной с применением объектно-ориентированной парадигмы программирования?
8. В чем отличие языка программирования С++ от C# и Java?
9. Определение IDE. Известные IDE.
10. Определение стэйтмента. Виды стэйтиментов.
11. Структура программы на языке программирования С++.
12. Что представляет собой пространство имен?
13. Комментарии в коде программы.
14. Операторы консольного ввода-вывода данных.
15. Структура программы на языке программирования С#.
16. Определение переменной. Как объявляются переменные в С++ и С#?
17. Что такое константа в языках программирования?
18. Что такое литерал? Как классифицируются литералы?
19. Какая операция применяется для изменения значения переменной?
20. Перечислите правила именования переменных.
21. Что такое область видимости переменной? В чем отличие глобальной области видимости от локальной?
22. В чем отличие l-value от r-value?
23. Что такое тип данных?
24. Как классифицируются типы данных?
25. Дайте характеристику типу данных bool.
26. Дайте характеристику типу данных int.
27. Дайте характеристику типу данных char.
28. Дайте характеристику типу данных double/float.
29. Как определяется размер переменной?
30. Какой инструмент языков программирования позволяет не указывать явно тип данных переменной? Приведите пример.
31. Какие арифметические операции реализованы в языках программирования?
32. В чем отличие постфиксного декремента от префиксного декремента?
33. Какие операторы сравнения и присваивания реализованы в языках программирования? Опишите их.
34. Какие логические операторы реализованы в языках программирования? Опишите их.
35. Какие методы используются для определения типа данных переменной?
36. Какие существуют способы для неявного преобразования типов?
37. Какие существуют варианты числового расширения?
38. Чем числовая конверсия отличается от числового расширения?
39. Как осуществляется явное преобразование типов в С++?
40. Как осуществляется явное преобразование типов в С#?
41. Какие операторы используются для реализации ветвления процесса выполнения программы?
42. Опишите синтаксис условного оператора. Приведите пример его использования.
43. Опишите синтаксис условного тернарного оператора. Приведите пример его использования.
44. Опишите синтаксис оператора множественного выбора. Приведите пример его использования.
45. Опишите синтаксис оператора управления потоком выполнения программы. Приведите пример его использования.
46. Какие существуют ограничения при использовании оператора goto?
47. Какие операторы используются для реализации многократного исполнения последовательности инструкций в коде программы?
48. В чем отличие циклов с предусловием от цикла с постусловием?
49. Опишите синтаксис цикла с предусловием. Приведите пример его использования.
50. Опишите синтаксис цикла с постусловием. Приведите пример его использования.
51. Опишите синтаксис цикла со счетчиком. Приведите пример его использования.
52. Как объявляется бесконечный цикл?
53. Для чего используется оператор break и continue?
54. Что такое массив?
55. Как происходит создание и инициализация массива на языке программирования С++?
56. Как происходит создание и инициализация массива на языке программирования С#?
57. Как происходит доступ к данным, хранимым в массиве?
58. Что такое указатель?
59. Как объявляется указатель в коде программы?
60. Какой оператор используется для записи или чтения данных, содержащихся в области памяти, адрес которой хранит указатель?
61. В каких арифметических операциях могут принимать участие указатели?
62. Что происходит с указателем при увеличении его значения с помощью операции инкремента?
63. Как вычисляется «расстояние» между указателями?
64. Как определяется приоритет операцией при одновременном использовании операции разыменования и постфиксного инкремента? Приведите пример.
65. Перечислите случаи использования указателей в программировании.
66. Что такое динамическое распределение памяти?
67. В каких случаях возникает утечка памяти?
68. Перечислите комбинации использования ключевого слова const с указателями.
69. Для чего используются указатели на указатели?
70. Как происходит создание двумерного динамического массива?
71. Что такое ссылочная переменная? В чем ее отличие от указателей?
72. Определение функции.
73. Опишите общую форму определения функции на языке С++ и С#.
74. Как происходит вызов функции?
75. Чем отличается передача параметров по значению от передачи параметров по адресу?
76. В С++ функцию нельзя вызвать до объявления самой функции. Какими способами устраняется данная проблема?
77. С помощью какого оператора реализуется возврат функцией определенного значения?
78. Как создаются функции с параметрами по умолчанию? Как происходит вызов функций с параметрами по умолчанию?
79. Как происходит перегрузка функций? Приведите пример перегрузки функций.
80. Перечислите преимущества и недостатки передачи аргументов по значению.
81. Перечислите преимущества и недостатки передачи аргументов по адресу.
82. Когда следует и когда не следует использовать возврат по значению в функции?
83. Когда следует и когда не следует использовать возврат по адресу в функции?
84. Когда следует и когда не следует использовать возврат по ссылке в функции?
85. Что такое указатель на функцию? Как объявляется указатель на функцию?
86. Как происходит передача указателя на функцию в другую функцию и возврат функцией указателя на другую функцию? Приведите пример.
87. Что такое рекурсия? Что такое базовый случай? Приведите пример рекурсивной функции.
88. Что представляет собой делегат? Укажите синтаксис объявления делегата.
89. Как происходит добавление методов в делегат и удаление методов из делегата?
90. В чем заключается принцип объектно-ориентированного программирования?
91. Перечислите основные принципы объектно-ориентированного программирования.
92. Что такое класс?
93. Можете ли вы вызвать метод базового класса, не создавая экземпляр?
94. В чем разница между классом и объектом?
95. Что такое модификатор доступа? Перечислите модификаторы доступа, указав их функциональность.
96. В чем преимущество использования объектно-ориентированной парадигмы программирования?
97. Какой принцип ООП позволяет рассматривать объект в виде самостоятельной программной единицы, полностью отвечающей за хранение и обработку своих данных и предоставляющей пользователям четко определенный набор функций?
98. Как называется элемент данных объекта?
99. Какие типы могут иметь свойства класса?
100. Как называется логически связанный блок кода, связанный с обработкой данных внутри класса?
101. Дайте определение метода-конструктора.
102. Дайте определение метода-деструктора.
103. Какие методы организуют контролируемый доступ к закрытым элементам данных объекта?
104. Перечислите этапы жизненного цикла объекта.
105. Что включает описание класса?
106. Перечислите общие правила для описания свойств класса.
107. Перечислите общие правила для описания методов класса.
108. В каких случаях допускается перегрузка конструкторов класса?
109. В виде какой функции оформляется Set-метод?
110. В виде какой функции оформляется Get-метод?
111. Укажите синтаксические правила для определения класса на языке С++. Приведите пример.
112. Укажите синтаксические правила для определения класса на языке С#. Приведите пример.
113. Как создается экземпляр класса?
114. Как происходит обращение к свойствам и методам экземпляра класса?
115. Какой уровень доступа к элементу данных класса определяет модификатор public?
116. Какой уровень доступа к элементу данных класса определяет модификатор private?
117. Какой уровень доступа к элементу данных класса определяет модификатор protected?
118. Что определяет интерфейс класса?
119. Какой модификатор доступа имеет член класса, если он явно не указан?
120. Чем поле отличается от свойства в языке С#?
121. Укажите стандартный синтаксис для определения свойства и автосвойства в языке С#?
122. При каких условиях можно установить модификатор доступа для блока set или get?
123. Какой может иметь модификатор доступа для блока set или get, если свойство имеет модификатор public?
124. Для чего используются конструкторы класса?
125. Укажите правила описания методов-конструкторов.
126. Что такое конструктор по умолчанию?
127. В каких ситуациях происходит удаление объекта с использованием деструктора?
128. Укажите правила описания методов-деструкторов.
129. В чем отличие значимых типов и ссылочных типов в С#?
130. Как работает копирование значение при присвоении данных объекту значимого типа?
131. Как работает копирование значение при присвоении данных объекту ссылочного типа?
132. Для чего используется ключевое слово ref?
133. В чем заключается назначение ключевого слова this?
134. Какой параметр неявно добавляется в метод класса на этапе компиляции?
135. Что представляют собой статические члены класса? Как они определяются в коде программы?
136. Какие существуют ограничения при определении статических функций (методов)?
137. Как можно осуществить обращение к статическим членам класса?
138. Что представляет собой статический конструктор и статический класс?
139. Какие операции используются для обращения к открытым свойствам и методам классов в языке программирования С++?
140. Чем отличается функциональность ключевого слова this в С++ и С#?
141. Какие переменные и методы следует определять как статические?
142. Какие операторы разрешено перегружать в унарном и бинарном представлении?
143. Какая операции не имеет фиксированного числа операндов при перегрузке?
144. Опишите общую форму метода для перегрузки бинарной операции в языке программирования С#?
145. Перечислите ссылочные типы в языке программирования С#?
146. Перечислите значимые типы языке программирования С#?
147. В чем заключается принцип абстрагирования, используемый при построении классов?