

算法分析与设计 复习题

判断题.....	1
选择题:	15

判断题

1. 算法就是一组有穷的规则。

答案：正确

2. 概率算法中蒙特卡罗算法得到的解必是正确的。

答案：错误

3. 程序和算法一样，都是某种程序设计语言的具体实现。

答案：错误

4. 合并排序算法是渐近最优算法。

答案：正确

5. 递归定义必须是有确切含义是指必须一步比一步简单，最后是有终结的，决不能无限循环下去。

答案：正确

6. 二分搜索方法在最坏的情况下用 $O(\log n)$ 时间完成搜索任务。

答案：正确

7. 能否利用分治法完全取决于问题是否具有如下特征：利用该问题分解出的子问题的解可以合并为该问题的解。

答案：正确

8. 分治法的基本思想是将一个规模较大的问题分解成若干个规模较小的子问题，这些子问题之间并不一定相互独立。

答案：错误

9. 递归算法的效率往往很低，费时和费内存空间。

答案：正确

10. 当一个问题具有最优子结构性质时只能用动态规划方法求解。

答案：错误

11. 如果一类活动过程一个阶段的决策确定以后，常影响到下一个阶段的决策，则称它为多阶段决策问题。

答案：正确

12. 反复应用分治手段，不能使子问题与原问题类型一致而其规模却不断缩小。

答案：错误

13. 斐波那契数列的定义： $f(n)=f(n-1)+f(n-2)$, $f(0)=1$, $f(1)=2$ ，其数据的定义形式不是按递归定义。

答案：错误

14. 0-1 背包问题与背包问题这两类问题都可以用贪心算法求解。

答案：错误

15. 证明贪心选择后的问题简化为规模更小的类似子问题的关键在于利用该问题的最优子结构性质。

答案：错误

16. 子问题之间不包含公共的子问题，这个条件涉及到分治法的效率。

答案：正确

17. 概率算法允许在执行过程中随机地选择下一个计算步骤。

答案：正确

18. 二分搜索法的二分查找只适用于顺序存储结构。

答案：正确

19. 要想在电脑上扩大所处理问题的规模，有效的途径是降低算法的计算复杂度。

答案：正确

20. 用回溯法解题一个显著特征是在搜索过程中动态产生问题的解空间。

答案：错误

21. 从分治法的一般设计模式可以看出，用它设计出的程序一般是一个递归过程。因此，分治法的计算效率通常可以用递归方程来进行分析。

答案：正确

22. 多阶段决策问题中，每一个阶段可能有若干个决策可供选择

答案：正确

23. 拉斯维加斯算法不会得到不正确的解，但有时找不到解。

答案：正确

24. 在通往边界条件的递归调用过程中，系统用堆栈保存的每次调用的中间结果是局部变量

和返回地址值。

答案：正确

25. 要想在电脑上扩大所处理问题的规模，有效的途径是提高算法的计算复杂度。

答案：错误

26. 程序必须满足算法具有数据输出的性质。

答案：正确

27. 反复应用分治手段，可以使子问题与原问题类型一致而其规模却不断缩小

答案：正确

28. 一个算法产生一个或多个输出，它们是同输入有某种特定关系的量

答案：正确

29. 最优子结构性性质特征反映了递归思想的应用

答案：正确

30. 递归边界本身并不使用递归的定义

答案：正确

31. 用分治法求解一个问题，所需的时间是由子问题的个数、大小以及把这个问题分解为子问题所需的工作总量来确定的。

答案：正确

32. 应用回溯法解问题时，首先应明确定义问题的解空间。问题的解空间应至少包含问题的一个（最优）解。

答案：正确

33. 好的约束函数能显著地减少所生成的结点数，但这样的约束函数往往计算量较大。因此，在选择约束函数时通常存在生成结点数与约束函数计算量之间的折衷。

答案：正确

34. 一个递归定义必须是有确切含义的，必须一步比一步简单，最后是有终结的，不能无限循环下去。

答案：正确

35. 最优子结构性性质是应用分治法的前提。

答案：正确

36. 操作系统，它是一个在无限循环中执行的程序，因而不是一个算法。

答案：正确

37. 有些数据结构如二叉树等，由于其本身的递归特性、特别适合用递归的形式来描述。

答案：正确

38. 概率算法的一个基本特征是, 对所求问题的同一个实例用同一个算法求解两次一定能得到完全相同的效果。

答案: 错误

39. 问题可以分解为若干个规模较小的相同问题, 即称该问题具有最优子结构性质。

答案: 错误

40. 递推是从内边界条件出发, 通过递推式达到边界条件。

答案: 正确

41. 所有的递归函数都能找到对应的非递归定义。

答案: 正确

42. 定义递归函数时可以没有初始值。

答案: 错误

43. 动态规划算法的基本要素是最优子结构。

答案: 正确

44. 最优子结构性质是指原问题的最优解包含其子问题的最优解。

答案: 正确

45. 动态规划算法求解问题时, 分解出来的子问题相互独立。

答案: 错误

46. 满足贪心选择性质必满足最优子结构性质。

答案: 错误

47. 回溯法中限界函数的目的是剪去得不到最优解的子树。

答案: 正确

48. 分支限界法类似于回溯法, 也是一种在问题的解空间树 T 上搜索问题解的算法, 两者的搜索方式是相同的。

答案: 错误

49. 任何递归算法都有递归出口。

答案: 正确

50. 递归算法的执行效率比功能相同的非递归算法的执行效率高。

答案: 错误

51. 递归算法不能转换成对应的非递归算法。

答案: 错误

52. 数据元素是数据的最小单位

答案：错误

53. 数据对象就是一组数据元素的集合

答案：错误

54. 任何数据结构都具备三个基本运算：插入、删除和查找。

答案：错误

55. 数据对象是由有限个类型相同的数据元素构成的。

答案：正确

56. 数据的逻辑结构与各数据元素在计算机中如何存储有关。

答案：错误

57. 如果数据元素值发生改变，则数据的逻辑结构也随之改变。

答案：错误

58. 逻辑结构相同的数据，可以采用多种不同的存储方法。

答案：正确

59. 逻辑结构不相同的数据，必须采用不同的存储方法来存储。

答案：错误

60. 数据的逻辑结构是指数据元素的各数据项之间的逻辑关系。

答案：错误

61. 顺序存储方式只能用于存储线性结构。

答案：错误

62. 算法可以用不同的语言来描述，如果用 C 语言或 Pascal 语言等高级语言来描述，则算法就等同于程序。

答案：错误

63. 数据的逻辑结构是指各数据元素之间的逻辑关系。

答案：正确

64. 数据结构、数据元素、数据项在计算机中的映像（或表示）分别称为存储结构、节点和数据域。

答案：正确

65. 数据的物理结构是指数据在计算机内的实际存储形式。

答案：正确

66.分配给单链表的内存与地址必须是连续的。

答案：错误

67.从长度为 n 的顺序表中删除任何一个元素，时间复杂度都是 $O(n)$ 。

答案：错误

68. 向顺序表中插入一个元素，平均要移动大约一半的元素。

答案：正确

69.凡是为空的单链表都是不含任何节点的。

答案：错误

70. 如果单链表带有头节点，则插入操作永远不会改变头节点指针的值。

答案：正确

71. 在循环单链表中，任何一个节点的指针域都不可能为空。

答案：正确

72. 顺序存储方式的特点是存储密度大且插入、删除运算效率高。

答案：错误

73. 线性表的顺序存储结构优于链式存储结构。

答案：错误

74. 顺序存储结构属于静态结构而链式存储结构属于动态结构。

答案：正确

75. 由于顺序存储结构要求连续的存储区域，所以在存储管理上不够灵活。

答案：正确

76. 对于单链表来说，只有从头节点开始才能扫描表中全部节点。

答案：正确

77. 对于循环单链表来说，从表中任一节点出发都能扫描表中全部节点。

答案：正确

78. 双链表的特点是很容易找任一节点的前趋和后继。

答案：正确

79. 线性表有两种存储结构：一是顺序表，二是链表。

答案：正确

80. 如果有多个线性表同时共存，并且在处理过程中各表的长度会动态地发生变化，线性表

的总数也会自动改变。在此情况下，应选用链式存储结构。

答案：正确

81.若线性表的总数基本稳定且很少进行插入和删除操作，但要求以最快的速度存取线性表中的元素，那么应该选用顺序存储结构。

答案：正确

82.对于单链表和双链表，均能从当前节点出发访问到任一节点。

答案：错误

83. 循环单链表和循环双链表从尾指针出发可以访问到链表中的任意节点。

答案：正确

84.若频繁地对一个线性表进行插入和删除操作，该线性表宜采用链式存储结构。

答案：正确

85.栈底元素是不能删除的元素。

答案：错误

86.顺序栈中元素值的大小是有序的。

答案：错误

87.栈顶元素和栈底元素有可能是同一个元素。

答案：正确

88. 若用 $s[1...m]$ 表示顺序栈的存储空间，则对栈的进栈、出栈操作最多只能进行 m 次。

答案：错误

89. 栈是一种对进栈、出栈操作总次数作了限制的线性表。

答案：错误

90. 空栈没有栈顶指针

答案：错误

91. 环形队列中有多少元素，可以根据队首指针和队尾指针的值来计算。

答案：正确

92. 无论是顺序队列，还是链式队列，插入、删除运算的时间复杂度都是 $O(1)$ 。

答案：正确

93. 栈和队列都是插入和删除操作受限的线性表。

答案：正确

94. 栈和队列的存储方式既可以是顺序方式，也可以是链式方式

答案：正确

95. 环形队列也存在空间溢出的问题

答案：正确

96. 消除递归不一定需要使用栈。

答案：正确

97. 二分搜索算法是利用贪心法实现的算法。

答案：错误

98. 动态规划算法通常是以自底向上的方式求解最优解。

答案：正确

99. 贪心法不能解决 0/1 背包问题。

答案：正确

100. 深度优先不是分支限界法的搜索方式。

答案：正确

101. 二分搜索算法是利用分治策略实现的算法。

答案：正确

102. 背包问题不能使用贪心法解决。

答案：错误

103. 单源最短路径问题不能使用贪心法解决。

答案：错误

104. 时间复杂度低是衡量一个算法好坏的标准。

答案：正确

105. 归并排序不可以使用分治法求解。

答案：错误

106. 拉斯维加斯算法有时找不到问题的解。

答案：正确

107. 舍伍德算法有时候找不到问题的解。

答案：错误

108. NP 问题都是不可能解决的问题

答案：错误

109. P 类问题包含在 NP 类问题中。

答案：正确

110. NP 类问题包含在 P 类问题中。

答案：错误

111. NP 完全问题是 P 类问题的子集

答案：错误

112. 蒙特卡罗算法是概率算法的一种

答案：正确

113. 蒙特卡罗算法是贪心算法的一种

答案：错误

114. 蒙特卡罗算法是回溯算法的一种

答案：错误

115. 动态规划算法不是随机化算法

答案：正确

116. 最优子结构性质是贪心算法与动态规划算法的共同点

答案：正确

117. 矩阵连乘问题的算法可由动态规划算法来设计实现

答案：正确

118. Strassen 矩阵乘法是利用分治策略实现的算法

答案：正确

119. Strassen 矩阵乘法是利用贪心法实现的算法

答案：错误

120. 贪心选择性质是贪心算法的基本要素

答案：正确

121. 以深度优先方式系统搜索问题解的算法称为回溯算法

答案：正确

122. 算法分析的两个主要方面是时间复杂度和空间复杂度分析

答案：正确

123. 实现最大子段和利用的算法是动态规划法

答案：正确

124. 实现最大子段和利用的算法是贪心法

答案：错误

125. 实现最大子段和利用的算法是回溯法

答案：错误

126. 广度优先是分支限界算法的一种搜索方式

答案：正确

127. 广度优先是回溯算法的一种搜索方式

答案：错误

128. 广度优先是贪心算法的一种搜索方式

答案：错误

129. 舍伍德算法是概率算法的一种

答案：正确

130. 舍伍德算法是贪心算法的一种。

答案：错误

131. 舍伍德算法是回溯算法的一种。

答案：错误

132. 实现最长公共子序列利用的算法是动态规划法。

答案：正确

133. 计算机算法指的是解决问题的方法和过程。

答案：正确

134. 根据排序元素所在位置的不同，排序分内排序和外排序。

答案：正确

135. 根据排序元素所在位置的不同，排序分首排序和尾排序。

答案：错误

136. 算法必须具备输入、输出和有穷性、确定性和可行性等 5 个特性。

答案：正确

137. 算法必须具备输入、输出和易读性、稳定性和安全性等 5 个特性。

答案：错误

138. 与分治法不同的是，适合于用动态规划求解的问题经分解得到的子问题往往不是相互独立的

答案：正确

139. 与分治法不同的是，适合于用动态规划求解的问题往往是相互独立的

答案：错误

140. 二分搜索算法的基本思想是将 n 个元素分成个数大致相同的两半，取 $a[n/2]$ 与 x 进行比较：如果 $x < a[n/2]$ ，则只要在数组 a 的左半部继续搜索 x

答案：正确

141. 二分搜索算法的基本思想是将 n 个元素分成个数大致相同的两半，取 $a[n/2]$ 与 x 进行比较：如果 $x > a[n/2]$ ，则只要在数组 a 的左半部继续搜索 x

答案：错误

142. 算法必须具备输入、输出和可执行性、可移植性和可扩充性等 5 个特性。

答案：错误

143. 适用动态规划的问题必须满足最优化原理和无后效性。

答案：正确

144. 适用动态规划的问题必须满足最优化原理和后效性。

答案：错误

145. 二分查找只适用于顺序存储结构。

答案：正确

146. 二分查找只适用于链式存储结构。

答案：错误

147. 应用分治法的两个前提是问题的可分性和解的可归并性。

答案：正确

148. 应用分治法的两个前提是问题的可分性和解的复杂性。

答案：错误

149. 对于 n 个元素的排序问题。 $n=2$ 时只要作 1 次比较即可排好序。

答案：正确

150. 对于 n 个元素的排序问题。 $n=2$ 时要作 2 次比较即可排好序。

答案：错误

151. 分治法所能解决的问题应具有的关键特征是利用该问题分解出的子问题的解可以合并为该问题的解。

答案：正确

152. 分治法所能解决的问题应具有的关键特征是问题的规模缩小到一定的程度就可以容易地解决。

答案：错误

153. 直接或间接的调用自身的算法称为递归算法。

答案：正确

154. 直接或间接的调用自身的算法称为动态规划算法。

答案：错误

155. 当上下限表示相等时我们使用 Θ 表示法来描述算法代价。

答案：正确

156. 当上下限表示相等时我们使用大 O 表示法来描述算法代价。

答案：错误

157. 递归通常用栈来实现。

答案：正确

158. 递归通常用队列来实现。

答案：错误

159. 分治法的设计思想是将一个难以直接解决的大问题分割成规模较小的子问题分别解决子问题最后将子问题的解组合起来形成原问题的解。这要求原问题和子问题的问题规模不同，问题性质相同。

答案：正确

160. 0/1 背包问题不能用贪心算法求解。

答案：正确

161. 可以由多项式时间算法求解的问题是易处理的。

答案：正确

162. 可以由多项式时间算法求解的问题是难处理的。

答案：错误

163. 需要超过多项式时间算法求解的问题是不能处理的。

答案：错误

164. 递归通常用数组来实现。

答案：错误

165. 哈密尔顿回路问题是典型的 NP 完全问题。

答案：正确

166. 排序问题是典型的 NP 完全问题。

答案：错误

167. 算法分析需要对算法需要多少计算时间和存储空间作定量分析。

答案：正确

168. 用数量级形式表示算法的执行时间称为算法的时间复杂度。

答案：正确

169. 用数量级形式表示算法的执行时间称为算法的空间复杂度。

答案：错误

170. 最坏情况下，顺序查找的时间复杂度为 $O(n)$ 。

答案：正确

171. 最坏情况下，折半查找的时间复杂度为 $O(\log_2 n)$ 。

答案：正确

172. 合并排序的基本运算是把两个或多个有序序列合并成一个有序序列

答案：正确

173. 最优子结构是动态规划算法的基本要素之一。

答案：正确

174. 快速排序算法是基于分治策略的一种排序算法。

答案：正确

175. 快速排序算法是基于回溯的一种排序算法。

答案：错误

176. 快速排序算法是基于贪心法的一种排序算法。

答案：错误

177. 贪心法通常以自顶向下的方式求解最优解。

答案：正确

178. 分治法通常以自顶向下的方式求解最优解。

答案：错误

179. 回溯法通常以自顶向下的方式求解最优解。

答案：错误

180. 不断回头寻找目标的方法称为回溯法。

答案：正确

181. 不断回头寻找目标的方法称为概率算法。

答案：错误

182. 不断回头寻找目标的方法称为贪心法。

答案：错误

183. 拉斯维加斯算法找到的解一定是正确的。

答案：正确

184. 拉斯维加斯算法找到的解正确与否不确定。

答案：错误

185. θ 记号在算法复杂性的表示法中表示紧致界。

答案：正确

186. θ 记号在算法复杂性的表示法中表示上界。

答案：错误

187. θ 记号在算法复杂性的表示法中表示下界。

答案：错误

188. 一个算法是对特定问题求解的一种描述，它是指令的有限序列。

答案：正确

189. 一个递归算法必须包括终止条件和递归部分。

答案：正确

190. 栈和队列的共同点是只允许在端点处插入和删除元素。

答案：正确

191. 排序趟数与原始序列有关的排序方法是冒泡排序法。

答案：正确

192. 栈和队列的共同点都是先进先出。

答案：错误

193. 栈和队列的共同点都是先进后出。

答案：错误

194. 排序趟数与原始序列有关的排序方法是选择排序法。

答案：错误

195. 在算法的三种情况下的复杂性中，可操作性最好且最有实际价值的是最坏情况下的时间复杂度。

答案：正确

196. 在算法的三种情况下的复杂性中，可操作性最好且最有实际价值的是最好情况下的时间复杂度。

答案：错误

197. 若一个算法的时间复杂度用 $T(n)$ 表示，其中 n 的含义是问题规模。

答案：正确

198. 合并排序法的基本思想是：将待排序元素分成大小大致相同的 2 个子集合，分别对每个子集合进行排序，最终将排好序的子集合合并成为所要求的排好序的集合。

答案：正确

选择题：

1. 二分搜索算法是利用（ A ）实现的算法。

选项 A. 分治策略

选项 B. 动态规划法

选项 C. 贪心法

选项 D. 回溯法

答案：

2. 回溯法解旅行售货员问题时的解空间树是（ B ）。

选项 A. 子集树

选项 B. 排列树

选项 C. 深度优先生成树

选项 D. 广度优先生成树

答案：

3. 下列算法中通常以自底向上的方式求解最优解的是（ B ）。

选项 A. 备忘录法

选项 B. 动态规划法

选项 C. 贪心法

选项 D. 回溯法

答案：

4. 下面不是分支界限法搜索方式的是（ D ）。

选项 A. 广度优先

选项 B. 最小耗费优先

选项 C. 最大效益优先

选项 D. 深度优先

答案：

5. 采用贪心算法的最优装载问题的主要计算量在于将集装箱依其重量从小到大排序，故算

法的时间复杂度（ B ）。

A、 $O()$

B、 $O(n \log n)$

C、 $O()$

D、 $O(n)$

6. 分支限界法求解最大团问题时，活结点表的组织形式是（ B ）。

A、最小堆

B、最大堆

C、栈

D、数组

7. 下面问题（ B ）不能使用贪心法解决。

A 单源最短路径问题

B N 皇后问题

C 最小花费生成树问题

D 背包问题

答案：

8. 下列算法中不能解决 0/1 背包问题的是（ A ）。

A 贪心法

B 动态规划

C 回溯法

D 分支限界法

答案：

9. 背包问题的贪心算法所需的计算时间为（ B ）。

A、 $O(n)$

B、 $O(n \log n)$

C、 $O()$

D、 $O(n)$

答案：

10. 二分搜索算法是利用（ A ）实现的算法。

A. 分治策略

B、动态规划法

C、贪心法

D、回溯法

答案：

11. 下列不是动态规划算法基本步骤的是（ B ）。

A、找出最优解的性质

B、构造最优解

- C、算出最优解
- D、定义最优解

答案：

12. 最大效益优先是（ A ）的一种搜索方式。

- A、分支界限法
- B、动态规划法
- C、贪心法
- D、回溯法

答案：

13. 在下列算法中有时找不到问题解的是（ B ）。

- A、蒙特卡罗算法
- B、拉斯维加斯算法
- C、舍伍德算法
- D、数值概率算法

答案：

14. 下列算法中通常以自底向上的方式求解最优解的是（ B ）。

- A、备忘录法
- B、动态规划法
- C、贪心法
- D、回溯法

答案：

14. 下列算法中通常以自底向上的方式求解最优解的是（ B ）。

- A、备忘录法
- B、动态规划法
- C、贪心法
- D、回溯法

答案：

15. 衡量一个算法好坏的标准是（ C ）。

- A 运行速度快
- B 占用空间少
- C 时间复杂度低
- D 代码短

答案：

16. 以下不可以使用分治法求解的是（ D ）。

- A、棋盘覆盖问题
- B、选择问题
- C、归并排序

D、 0/1 背包问题

答案：

17. 实现循环赛日程表利用的算法是（ A ）

A、分治策略

B、动态规划法

C、贪心法

D、回溯法

答案：

18、下列随机算法中运行时有时候成功有时候失败的是（ C ）

A 数值概率算法

B 舍伍德算法

C 拉斯维加斯算法

D 蒙特卡罗算法

答案：

19. 下面不是分支界限法搜索方式的是（ D ）。

A、广度优先

B、最小耗费优先

C、最大效益优先

D、深度优先

答案：

20. 下列算法中通常以深度优先方式系统搜索问题解的是（ D ）。

A、备忘录法

B、动态规划法

C、贪心法

D、回溯法

答案：

21. 备忘录方法是那种算法的变形。（ B ）。

A、分治法

B、动态规划法

C、贪心法

D、回溯法

答案：

22. 哈弗曼编码的贪心算法所需的计算时间为（ B ）。

A、 $O(n)$

B、 $O(n \log n)$

C、 $O()$

D、 $O(n)$

答案：

23. 最长公共子序列算法利用的算法是（ B ）

A、分支界限法

B、动态规划法

C、贪心法

D、回溯法

答案：

24. 实现棋盘覆盖算法利用的算法是（ A ）。

A、分治法

B、动态规划法

C、贪心法

D、回溯法

答案：

25.下面是贪心算法的基本要素的是（ C ）。

- A、重叠子问题
- B、构造最优解
- C、贪心选择性质
- D、定义最优解

26.回溯法的效率不依赖于下列哪些因素（ D ）。

- A. 满足显约束的值的个数
- C. 计算限界函数的时间
- B. 计算约束函数的时间
- D. 确定解空间的时间

27.下面哪种函数是回溯法中为避免无效搜索采取的策略（ B ）。

- A. 递归函数
- B. 剪枝函数
- C. 随机数函数
- D. 搜索函数

28、下面关于 NP 问题说法正确的是（ B ）。

- A. NP 问题都是不可能解决的问题
- B. P 类问题包含在 NP 类问题中
- C. NP 完全问题是 P 类问题的子集
- D. NP 类问题包含在 P 类问题中

29、蒙特卡罗算法是（ B ）的一种。

- A. 分支界限算法
- B. 概率算法
- C. 贪心算法
- D. 回溯算法

答案：

30.下列哪一种算法不是随机化算法（ C ）。

- A. 蒙特卡罗算法
- B. 拉斯维加斯算法
- C. 动态规划算法
- D.舍伍德算法

答案：

31.（ D ）是贪心算法与动态规划算法的共同点。

- A、重叠子问题
- B、构造最优解

- C、贪心选择性质
- D、最优子结构性性质

答案：

32. 矩阵连乘问题的算法可由（ B ）设计实现。

- A、分支界限算法
- B、动态规划算法
- C、贪心算法
- D、回溯算法

答案：

33、Strassen 矩阵乘法是利用（ A ）实现的算法。

- A、分治策略
- B、动态规划法
- C、贪心法
- D、回溯法

答案：

34、使用分治法求解不需要满足的条件是（ A ）。

- A 子问题必须是一样的
- B 子问题不能够重复
- C 子问题的解可以合并
- D 原问题和子问题使用相同的方法求解

答案：

35、回溯法搜索状态空间树是按照（ C ）的顺序。

- A 中序遍历
- B 广度优先遍历
- C 深度优先遍历
- D 层次优先遍历

答案：

36、实现合并排序利用的算法是（ A ）

- A、分治策略
- B、动态规划法
- C、贪心法
- D、回溯法

答案：

37、下列是动态规划算法基本要素的是（ D ）

- A、定义最优解
- B、构造最优解
- C、算出最优解
- D、子空间重叠性质

答案：

38. 采用广度优先策略搜索的算法是（ A ）

- A、分支界限法
- B、动态规划法
- C、贪心法
- D、回溯法

答案：

39. 在下列算法中得到的解未必正确的是（ A ）

- A、蒙特卡罗算法
- B、拉斯维加斯算法
- C、舍伍德算法
- D、数值概率算法

答案：

40. 实现大整数的乘法是利用的算法（ C ）

- A、贪心法
- B、动态规划法
- C、分治策略
- D、回溯法

答案：

41. 0-1 背包问题的回溯算法所需的计算时间为（ A ）

- A、 $O(n)$
- B、 $O(n \log n)$
- C、 $O(1)$
- D、 $O(n^2)$

答案：

42. 贪心算法与动态规划算法的主要区别是（ B ）

- A、最优子结构
- B、贪心选择性质
- C、构造最优解
- D、定义最优解

答案：

43. 实现最大子段和利用的算法是（ B ）。

- A、分治策略
- B、动态规划法
- C、贪心法
- D、回溯法

答案：

44. 优先队列式分支限界法选取扩展结点的原则是（ C ）

- A、先进先出
- B、后进先出
- C、结点的优先级
- D、随机

答案：

45、广度优先是（ A ）的一种搜索方式。

- A、分支界限算法
- B、动态规划法
- C、贪心算法
- D、回溯算法

答案：

46、舍伍德算法是（ B ）的一种

- A、分支界限算法
- B、概率算法
- C、贪心算法
- D、回溯算法

答案：

47、在下列算法中有时找不到问题解的是（ B ）

- A、蒙特卡罗算法
- B、拉斯维加斯算法
- C、舍伍德算法
- D、数值概率算法

答案：

48 下列哪一种算法是随机化算法（ D ）。

- A. 贪心算法
- B. 回溯法
- C. 动态规划算法
- D. 舍伍德算法

答案：

49. 一个问题可用动态规划算法或贪心算法求解的关键特征是问题的（ B ）。

- A、重叠子问题
- B、最优子结构性

C、贪心选择性质

D、定义最优解

答案：

52. 以深度优先方式系统搜索问题解的算法称为（ D ）。

A、分支界限算法

B、概率算法

C、贪心算法

D、回溯算法

答案：

53. 实现最长公共子序列利用的算法是（ B ）。

A、分治策略

B、动态规划法

C、贪心法

D、回溯法

54. 算法分析的两个主要方面是（ A ）。

A. 空间复杂度和时间复杂度

B. 正确性和简单性

C. 可读性和文档性

D. 数据复杂度和程序复杂度

55. 计算机算法指的是（ C ）。

A. 计算方法

B. 排序方法

C. 解决问题的方法和过程

D. 调度方法

56. 多阶段决策问题就是要在可以选择的那些策略中间选取一个（ A ）策略使在预定的标准下达到最好的效果。

A. 最优

B. 最差

C. 平衡

D. 任意

57. 根据排序元素所在位置的不同，排序分（ A ）。

A. 内排序和外排序

B. 首排序和尾排序

C. 顺序排序和逆序排序

D. 堆排序和栈排序

58. 算法必须具备输入、输出和（ B ）等 5 个特性。

A. 可执行性、可移植性和可扩充性

- B. 可行性、确定性和有穷性
- C. 确定性、有穷性和稳定性
- D. 易读性、稳定性和安全性

59. 与分治法不同的是，适合于用动态规划求解的问题（ A ）。

- A. 经分解得到子问题往往不是互相独立的
- B. 经分解得到子问题往往是互相独立的
- C. 经分解得到子问题往往是互相交叉的
- D. 经分解得到子问题往往是任意的

60. 二分搜索算法的基本思想是将 n 个元素分成个数大致相同的两半，取 $a[n/2]$ 与 x 进行比较：如果（ A ），则只要在数组 a 的左半部继续搜索 x 。

- A. $x < a[n/2]$
- B. $x = a[n/2]$
- C. $x > a[n/2]$
- D. $x \geq a[n/2]$

61. 活动安排问题就是在所给的活动集合中，选出（ C ）的相容活子集。

- A. 最小
- B. 任意
- C. 最大
- D. 一个

62. 在对问题的解空间树进行搜索的方法中一个活结点最多有一次机会成为活结点的是（ B ）。

- A. 回溯法
- B. 分支限界法
- C. 回溯法和分支限界法
- D. 回溯法求解子集树问题

63. 适用动态规划的问题必须满足（ D ）。

- A. 最优化原理
- B. 无前效性
- C. 最优化原理和后效性
- D. 最优化原理和无后效性

64. 算法的每种运算必须要有确切的定义不能有二义性，以下符合算法确定性运算的是（ B ）。

- A. $5/0$
- B. 将 6 或 7 与 x 相加
- C. 未赋值变量参与运算
- D. $f(n) = f(n-1) + 2$, $F(1) = 10$, n 为自然数

65. 直接或间接的调用自身的算法称为（ B ）。

- A. 贪心算法
- B. 递归算法
- C. 迭代算法
- D. 动态规划算法

66. 二分查找只适用（ B ）存储结构。

- A. 堆
- B. 顺序
- C. 任意顺序
- D. 栈

67. 应用分治法的两个前提是（ A ）。

- A. 问题的可分性和解的可归并性
- B. 问题的可分性和解的存在性
- C. 问题的复杂性和解的可归并性
- D. 问题的可分性和解的复杂性

68. 优先队列的分支限界法将活结点表组织成一个优先队列，并按优先队列中规定的结点优先级选取优先级最高的下一个结点成为当前扩展结点。优先队列中规定的结点优先级常用一个与该结点相关的数值 p 来表示。结点优先级的高低与 p 值大小相关，根据问题的不同情况，采用（ C ）来描述优先队列。

- A. 先进先出队列
- B. 后进先出的栈
- C. 最大堆或最小堆
- D. 随机序列

69. 阶乘函数用递归定义 `Public static int factorial(int n) {if(n==0)return 1; return(B);}`

- A. `n*factorial(n)`
- B. `n*factorial(n-1)`
- C. `n*factorial(n-2)`
- D. `n*factorial(n+1)`

70. （ B ）能够求得问题的解但却无法有效地判定解的正确性。

- A. 数值概率算法
- B. 蒙特卡罗算法
- C. 拉斯维加斯算法
- D. 舍伍德算法

71. 对于 n 个元素的排序问题。 $n=2$ 时只要作（ C ）次比较即可排好序。

- A. 3
- B. 2
- C. 1
- D. 4

72. 一般地讲, 当一个问题所有子问题都至少要解一次时, 用动态规划算法和备忘录算法相比: (B)。

- A. 效果一样
- B. 动态规划效果好
- C. 备忘录方法效果好
- D. 无法判断哪个效果好

73. 分支限界法与回溯法都是在问题的解空间树 T 上搜索问题的解, 二者 (B)。

- A. 求解目标不同搜索方式相同
- B. 求解目标不同搜索方式也不同
- C. 求解目标相同搜索方式不同
- D. 求解目标相同搜索方式也相同

74. 递归算法不能适用以下场合 (D)。

- A. 数据的定义形式按递归定义
- B. 数据之间的关系即数据结构按递归定义
- C. 问题解法按递归算法实现
- D. 概率问题

75. 若当子问题之间包含公共的子子问题时, 则分治法要做许多不必要的工作, 重复地解公共的子问题, 此时一般用 (A) 法较好。

- A. 动态规划
- B. 分治
- C. 贪心
- D. 概率

76. 分治法所能解决的问题应具有的关键特征是 (C)。

- A. 该问题的规模缩小到一定的程度就可以容易地解决
- B. 该问题可以分解为若干个规模较小的相同问题
- C. 利用该问题分解出的子问题的解可以合并为该问题的解
- D. 该问题所分解出的各个子问题是相互独立的

77. 对于货箱装船问题根据贪心策略首先选择 (A) 的货箱然后选 (A) 的货箱如此下去直到所有货箱均装上船或船上不能再容纳其他任何一个货箱。

- A. 最轻 次轻
- B. 最重 次重
- C. 最轻 次重
- D. 最重 次轻

78. 用回溯法解 n 后问题时, 用完全 n 叉树表示解空间。可行性约束 place 剪去不满足行、列和斜线约束的子树, place 中的 if 判断条件应为 (A)。

- A. $(\text{Math.abs}(k-j) == \text{Math.abs}(x[j]-x[k])) \vee |x[j] == x[k]|$
- B. $(\text{Math.abs}(k-j) == \text{Math.abs}(x[j]-x[k]))$

- C. $(x[j]-x[k])$
- D. 以上都不正确

79. 分支限界法的搜索策略是：在扩展结点处，先生成其（ D ）儿子结点（分支），然后再从当前的活结点表中选择下一个扩展对点。为了有效地选择下一扩展结点，以加速搜索的进程，在每一活结点处，计算一个函数值（限界），并根据这些已计算出的函数值，从当前活结点表中选择一个最有利的结点作为扩展结点，使搜索朝着解空间树上有最优解的分支推进，以便尽快地找出一个最优解。

- A. 一个
- B. 二个
- C. 任意多个
- D. 所有的

80. 能够用动态规划解决的问题还有一个显著特征（ D ）这个性质并不是动态规划适用的必要条件，但是如果该性质无法满足，动态规划算法同其他算法相比就不具备优势。

- A. 子问题的可求解性
- B. 子问题的独立性
- C. 子问题的可合并性
- D. 子问题的重叠性

81. 在任何一个 $2^k * 2^k$ 的棋盘覆盖中，用到的 L 型骨牌个数恰为（ A ）。

- A. $(4^k - 1) / 3$
- B. $(4^k - 1) / 2$
- C. $(2^k - 1) / 3$
- D. $(2^k - 1) / 2$

82. 以 Bitonic 旅行路线问题为例，动态规划的时间复杂度为（ C ）。

- A. $O(n)$
- B. $O(n!)$
- C. $O(n^2)$
- D. $O(n^3)$

83. 一个算法应该包含如下几条性质除了（ A ）。

- A. 二义性
- B. 有限性
- C. 正确性
- D. 可终止性

84. 解决一个问题通常有多种方法。若说一个算法“有效”是指（ D ）。

- A. 这个算法能在一定的时间和空间资源限制内将问题解决
- B. 这个算法能在人的反应时间内将问题解决
- C. 这个算法比其他已知算法都更快地将问题解决
- D. A 和 C

85. 当输入规模为 n 时算法增长率最小的是 (B) 。

- A. $5n$
- B. $20\log_2 n$
- C. $2n^2$
- D. $3n\log_3 n$

86. 渐进算法分析是指 (B) 。

- A. 算法在最佳情况、最差情况和平均情况下的代价
- B. 当规模逐步往极限方向增大时对算法资源开销“增长率”上的简化分析
- C. 数据结构所占用的空间
- D. 在最小输入规模下算法的资源代价

87. 当上下限表达式相等时我们使用下列哪种表示法来描述算法代价? (C)

- A. 大 O 表示法
- B. 大 Ω 表示法
- C. Θ 表示法
- D. 小 o 表示法

88. 采用“顺序搜索法”从一个长度为 N 的随机分布数组中搜寻值为 K 的元素，以下对顺序搜索法分析正确的是 (B)。

- A. 最佳情况、最差情况和平均情况下顺序搜索法的渐进代价都相同
- B. 最佳情况的渐进代价要好于最差情况和平均情况的渐进代价
- C. 最佳情况和平均情况的渐进代价要好于最差情况的渐进代价
- D. 最佳情况的渐进代价要好于平均情况的渐进代价而平均情况的渐进代价要好于最差情况的渐进代价

89. 递归通常用 (C) 来实现。

- A. 有序的线性表
- B. 队列
- C. 栈
- D. 数组

90. 分治法的设计思想是将一个难以直接解决的大问题分割成规模较小的子问题分别解决子问题最后将子问题的解组合起来形成原问题的解。这要求原问题和子问题 (C)。

- A 问题规模相同，问题性质相同
- B 问题规模相同，问题性质不同

- C 问题规模不同，问题性质相同
- D 问题规模不同，问题性质不同

91. 在寻找 n 个元素中第 k 小元素问题中如快速排序算法思想运用分治算法对 n 个元素进行划分如何选择划分基准下面（ D ）答案解释最合理。

- A. 随机选择一个元素作为划分基准
- B. 取子序列的第一个元素作为划分基准
- C. 用中位数的中位数方法寻找划分基准
- D. 以上皆可行。但不同方法算法复杂度上界可能不同

92. 对于 01 背包问题和背包问题的解法下面（ C ）答案解释正确。

- A. 01 背包问题和背包问题都可用贪心算法求解
- B. 01 背包问题可用贪心算法求解但背包问题则不能用贪心算法求解
- C. 01 背包问题不能用贪心算法求解但可以使用动态规划或搜索算法求解而背包问题则可以用贪心算法求解
- D. 因为 01 背包问题不具有最优子结构性质所以不能用贪心算法求解

93. 关于回溯搜索法的介绍下面（ D ）是不正确描述。

- A. 回溯法有“通用解题法”之称它可以系统地搜索一个问题的所有解或任意解
- B. 回溯法是一种既带系统性又带有跳跃性的搜索算法
- C. 回溯算法在生成解空间的任一结点时先判断该结点是否可能包含问题的解如果肯定不包含则跳过对该结点为根的子树的搜索逐层向祖先结点回溯
- D. 回溯算法需要借助队列这种结构来保存从根结点到当前扩展结点的路径

94. 关于回溯算法和分支限界法以下（ A ）是不正确描述。

- A 回溯法中每个活结点只有一次机会成为扩展结点
- B 分支限界法中活结点一旦成为扩展结点就一次性产生其所有儿子结点在这些儿子结点中那些导致不可行解或导致非最优解的儿子结点被舍弃其余儿子加入活结点表中
- C 回溯法采用深度优先的结点生成策略
- D 分支限界法采用广度优先或最小耗费优先最大效益优先的结点生成策略

95. 优先队列通常用以下（ B ）数据结构来实现。

- A. 栈
- B. 堆
- C. 队列
- D. 二叉查找树

96. 在分支限界算法中根据从活结点表中选择下一扩展结点的不同方式可有几种常用分类，以下（ D ）描述最为准确。

- A 采用 FIFO 队列的队列式分支限界法
- B 采用最小值堆的优先队列式分支限界法

- C 采用最大值堆的优先队列式分支限界法
- D 以上都常用针对具体问题可以选择采用其中某种更为合适的方式

97. 对布线问题以下（ C ）是不正确描述。

- A 布线问题的解空间是一个图
- B 可以对方格阵列四周设置围墙，即增设标记的附加方格的预处理，使得算法简化对边界的判定
- C 采用广度优先的标号法找到从起点到终点的布线方案这个方案如果存在的话不一定是最短的
- D 采用先入先出的队列作为活结点表以，终点 b 为扩展结点或活结点队列为空作为算法结束条件

98. 下述表达不正确的是（ D ）。

- A. $n^2 / 2 + 2^n$ 的渐进表达式上界函数是 $O(2^n)$
- B. $n^2 / 2 + 2^n$ 的渐进表达式下界函数是 $\Omega(2^n)$
- C. $\log n^3$ 的渐进表达式上界函数是 $O(\log n)$
- D. $\log n^3$ 的渐进表达式下界函数是 $\Omega(n^3)$

99. 当输入规模为 n 时，算法增长率最大的是（ A ）。

- A. 5^n
- B. $20 \log_2 n$
- C. $2n^2$
- D. $3n \log_3 n$

100. T(n)表示当输入规模为 n 时的算法效率，以下算法效率最优的是（ C ）。

- A. $T(n)=T(n-1)+1$, $T(1)=1$
- B. $T(n)=2n^2$
- C. $T(n)=T(n/2)+1$, $T(1)=1$
- D. $T(n)=3n \log_2 n$

101. 有 9 个村庄，其坐标位置如下表所示：

i	1	2	3	4	5	6	7	8	9
---	---	---	---	---	---	---	---	---	---

x(i)	1	2	3	4	5	6	7	8	9
y(i)	1	2	3	4	5	6	7	8	9

现在要盖一所邮局为这 9 个村庄服务，请问邮局应该盖在（ C ）才能使邮局到 9 个村庄的距离和最短。

- A. (4.5, 0)
- B. (4.5, 4.5)
- C. (5, 5)
- D. (5, 0)

102. n 个人拎着水桶在一个水龙头前面排队打水水桶有大有小水桶必须打满水流恒定。如下（ A ）说法不正确

- A. 让水桶大的人先打水可以使得每个人排队时间之和最小
- B. 让水桶小的人先打水可以使得每个人排队时间之和最小
- C. 让水桶小的人先打水在某个确定的时间 t 内可以让尽可能多的人打上水
- D. 若要在尽可能短的时间内 n 个人都打完水按照什么顺序其实都一样

103. 对于含有 n 个元素的子集树问题，最坏情况下其解空间的叶结点数目为（ B ）。

- A. $n!$
- B. 2^n
- C. $2^{n+1} - 1$
- D. $\sum_{i=1}^n n!/i!$

104. 以下关于判定问题难易处理的叙述中正确的是（ C ）

- A. 可以由多项式时间算法求解的问题是难处理的
- B. 需要超过多项式时间算法求解的问题是易处理的
- C. 可以由多项式时间算法求解的问题是易处理的
- D. 需要超过多项式时间算法求解的问题是不能处理的

105. 回溯法在解空间树 T 上的搜索方式是（ A ）

- A. 深度优先
- B. 广度优先
- C. 最小耗费优先
- D. 活结点优先

106. 设 $f(N)$, $g(N)$ 是定义在正数集上的正函数，如果存在正的常数 C 和自然数 N_0 ，使得当 $N \geq N_0$ 时有 $f(N) \leq Cg(N)$ ，则称函数 $f(N)$ 当 N 充分大时有上界 $g(N)$ ，记作 $f(N) = O(g(N))$ ，即 $f(N)$ 的

阶（ A ） $g(N)$ 的阶。

- A. 不高于
- B. 不低于
- C. 等价于
- D. 逼近

107. 以下（ C ）不能在线性时间完成排序

- A. 计数排序
- B. 基数排序
- C. 堆排序
- D. 桶排序

108. 以下（ A ）不一定得到问题的最优解

- A. 贪心算法
- B. 回溯算法
- C. 分支限界法
- D. 动态规划法

109. 下列不属于一个好的算法应具有的特性的是（ C ）。

- A. 正确性
- B. 简明性
- C. 无限性
- D. 最优性

110. 算法分析是（ C ）。

- A. 将算法用某种程序设计语言恰当地表示出来
- B. 在抽象数据集合上执行程序，以确定是否会产生错误的结果
- C. 对算法需要多少计算时间和存储空间作定量分析
- D. 证明算法对所有可能的合法输入都能算出正确的答案

111. 学校要举行运动会，请你设计一个能够对运动员分数自动排序的软件，如果要设计此软件，以下最好的方法和步骤是（ C ）。

- A. 分析问题，编写程序，设计算法，调试程序
- B. 设计算法，编写程序，提出问题，调试程序
- C. 提出问题，设计算法，编写程序，调试程序
- D. 设计算法，提出问题，编写程序，调试程序

112. 考虑背包问题： $n=6$ ， $M=10$ ， $V(1:6) = (15, 59, 21, 30, 60, 5)$ ， $W(1:6) = (1, 5, 2, 3, 6, 1)$ 。该问题的最大效益值为（ C ）。

- A. 101

- B. 110
- C. 115
- D. 120

113. 考虑背包问题: $n=6$, $M=10$, $V(1:6) = (15, 59, 21, 30, 60, 5)$, $W(1:6) = (1, 5, 2, 3, 6, 1)$ 。若把它看作是 0/1 背包问题, 则该问题的最大效益值为 (B)。

- A. 101
- B. 110
- C. 115
- D. 120

114. 找最小生成树的算法 Kruskal 的时间复杂度为 (D)。

- A. $O(n^2)$
- B. $O(m \log n)$
- C. $O(n \log m)$
- D. $O(m \log m)$

115. 算法与程序的区别在于算法具有 (C)。

- A. 能行性
- B. 确定性
- C. 有穷性
- D. 输入和输出

116. 设 $A[1..60] = \{11, 12, \dots, 70\}$ 。算法折半查找在 A 上搜索 $x=33$ 、7、70、77 时执行的元素比较次数分别为 a、b、c、d, 则 (C)。

- A. $a < b < c < d$
- B. $a > b = c = d$
- C. $a < b = c = d$
- D. $a < c < b = d$

117. 算法直接插入排序在 $A[1..8] = \{45, 33, 24, 45, 12, 12, 24, 12\}$ 上运行时执行的元素比较次数为 (D)。

- A. 14
- B. 28
- C. 7
- D. 22

118. void hanoi(int n, int a, int b, int c)

{

```

    if (n>0)
    {
        hanoi( n-1, a, c, b);
        move(a, b);
        hanoi( n-1, c, b, a);
    }
}

```

上述算法的时间复杂度为（ A ）。

- A. $O(2^n)$
- B. $O(n \log n)$
- C. $\theta(n!)$
- D. $\theta(n^n)$

119. 当一个确定性算法在最坏情况下的计算复杂性与其在平均情况下的计算复杂性有较大差别时，可以使用（ B ）来消除或减少问题的好坏实例间的这种差别。

- A. 数值概率算法
- B. 舍伍德算法
- C. 拉斯维加斯算法
- D. 蒙特卡罗算法

120. 当输入规模为 n 时，算法增长率最快的是（ C ）。

- A. $12n$
- B. $100 \log_2 n$
- C. $2n^2$
- D. $3n \log_3 n$

121. 关于 0-1 背包问题，以下描述正确的是（ D ）。

- A. 可以使用贪心算法找到最优解
- B. 能找到多项式时间的有效算法
- C. 使用教材介绍的动态规划方法可求解任意 0-1 背包问题
- D. 对于同一背包与相同的物品，做背包问题取得的总价值一定大于等于做 0-1 背包问题。

122. 设有 n 项独立的作业 $\{1, 2, \dots, n\}$ ，由 m 台相同的机器加工处理。作业 i 所需要的处理时间为 t_i 。约定：任何一项作业可在任何一台机器上处理，但未完工前不准中断处理；任何作业不能拆分更小的子作业。多机调度问题要求给出一种调度方案，使所给的 n 个作业在尽可能短的时间内由 m 台机器处理完（ $n > m$ ）。对于多级调度问题，使用哪种贪心策略比较合适（ B ）。

- A. 作业从小到大依次分配给空闲的机器
- B. 作业从大到小依次分配给空闲的机器
- C. 每个机器分配一样的作业数

D. 使用以上几种贪心策略都能找到最优解，所以都合适

123. 使用二分搜索算法在 1000 个有序元素表中搜索一个特定元素，在最坏情况下，搜索总共需要比较的次数为（ A ）。

- A. 10
- B. 11
- C. 500
- D. 1000

124. 用数量级形式表示算法的执行时间称为算法的（ A ）。

- A. 时间复杂度
- B. 空间复杂度
- C. 处理器复杂度
- D. 通信复杂度

125. 下面哪个问题不是典型的 NP 完全问题（ D ）。

- A. m-着色问题
- B. 旅行商问题
- C. 哈密尔顿回路问题
- D. 排序问题

126. 顺序查找的时间复杂度为（ A ）。

- A. $O(n)$
- B. $O(\log n)$
- C. $O(n^2)$
- D. $O(n \log n)$

127. 折半查找的时间复杂度为（ B ）。

- A. $O(n)$
- B. $O(\log n)$
- C. $O(n^2)$
- D. $O(n \log n)$

128. 算法的复杂性有时间复杂性和（ C ）复杂性之分。

- A. 处理器复杂性
- B. 通信复杂性
- C. 空间复杂性
- D. 存储复杂性

129. 算法的复杂性有空间复杂性和（ C ）复杂性之分。

- A. 处理器复杂性
- B. 通信复杂性

- C. 时间复杂性
- D. 存储复杂性

130. 算法是由若干条指令组成的有穷序列，且要满足输入、输出、确定性和有限性四条性质。其中算法的“确定性”是指组成算法的每条（ B ）是清晰的，无歧义的。

- A. 程序
- B. 指令
- C. 语句
- D. 语句块

131.（ C ）的基本运算是把两个或多个有序序列合并成一个有序序列。

- A. 快速排序
- B. 希尔排序
- C. 合并排序
- D. 堆排序

132. 解决 0/1 背包问题可以使用动态规划，回溯法，分支限界法。其中不需要排序的是（ A ）。

- A. 动态规划
- B. 回溯法
- C. 分支限界法
- D. 以上 3 种方法都需要排序

133. 解决 0/1 背包问题时需要排序的方法是回溯法和（ B ）。

- A. 动态规划
- B. 分支限界法
- C. 贪心法
- D. 线性规划

134. 下面哪项是动态规划算法基本要素之一（ D ）。

- A、定义最优解
- B、构造最优解
- C、算出最优解
- D、最优子结构

135. 快速排序算法是基于（ A ）的一种排序算法。

- A. 分治策略
- B. 贪心
- C. 回溯
- D. 动态规划

136.（ C ）是贪心算法可行的第一个基本要素，也是贪心算法与动态规划算法的主要区

别。

- A、重叠子问题
- B、最优子结构
- C、贪心选择性质
- D、定义最优解

137. 如果无向连通图 G 中不包含任何关节点，则称该图 G 为（ A ）。

- A. 双连通图
- B. 单连通图
- C. 强连通图
- D. 弱连通图

138. 使用剪枝函数的深度优先生成状态空间树中结点的求解方法称为（ D ）。

- A. 动态规划
- B. 分支限界法
- C. 贪心法
- D. 回溯法

139. 回溯法的算法框架按照问题的解空间一般分为子集树算法框架和（ A ）算法框架。

- A. 排列树
- B. 二叉树
- C. B 树
- D. B^+ 树

140. 下列算法中通常以自顶向下的方式求解最优解的是（ C ）。

- A. 分治法
- B. 动态规划法
- C. 贪心法
- D. 回溯法

141. 哈夫曼编码可利用（ C ）算法实现。

- A、分治策略
- B、动态规划法
- C、贪心法
- D、回溯法

142. 在对问题的解空间树进行搜索的方法中一个活结点有多次机会成为活结点的是（ A ）。

- A. 回溯法
- B. 分支限界法
- C. 回溯法和分支限界法
- D. 动态规划

143、秦始皇吞并六国使用的远交近攻逐个击破的连横策略采用了以下哪种算法思想（ B ）。

- A、递归
- B、分治
- C、迭代
- D、模拟。

144、FIFO 是（ A ）的一搜索方式。

- A、分支界限法
- B、动态规划法
- C、贪心法
- D、回溯法

145、投点法是（ B ）的一种。

- A、分支界限算法
- B、概率算法
- C、贪心算法
- D、回溯算法

146、若线性规划问题存在最优解它一定不在（ C ）。

- A 可行域的某个顶点上
- B 可行域的某条边上
- C 可行域内部
- D 以上都不对

147、在一般输入数据的程序里输入多多少少会影响到算法的计算复杂度，为了消除这种影响可用（ B ）对输入进行预处理。

- A、蒙特卡罗算法
- B、拉斯维加斯算法
- C、舍伍德算法
- D、数值概率算法

148、若 L 是一个 NP 完全问题 L 经过多项式时间变换后得到问题 l 则 l 是（ A ）。

- A、P 类问题
- B、NP 难问题
- C、NP 完全问题
- D、P 类语言

149. 不断回头寻找目标的方法称为（ C ）。

- A. 动态规划
- B. 贪心法
- C. 回溯法
- D. 概率算法

150. 拉斯维加斯算法找到的解一定是（ B ）。

- A. 不确定的
- B. 正确的
- C. 不正确的
- D. 局部最优的

151. θ 记号在算法复杂性的表示法中表示 (C)。

- A. 上界
- B. 下界
- C. 紧致界
- D. 以上说法都不对

152. 一个无向连通图不是双向连通图的充要条件是图中存在 (B)。

- A. 回路
- B. 关节点
- C. 最大团
- D. 最小团

153. 一个算法是对特定问题求解的一种描述, 它是 (A)。

- A. 指令的有限序列
- B. 程序的有限序列
- C. 语句的有限序列
- D. 代码的有限序列

154. 矩阵乘法如下:

```
for (i=0; i<n; i++)  
    for (j=0; j<n; j++) {  
        C[i][j]=0;  
        for (k=0; k<n; k++)  
            C[i][j]+=a[i][k]*b[k][j];  
    }
```

它的渐近时间复杂度为 (B)。

- A. $O(n^2)$
- B. $O(n^3)$
- C. $O(n)$
- D. $O(n^4)$

155. 二分搜索过程的算法行为可以用一棵 (B) 来描述。

- A. 二叉排序树
- B. 二叉判定树
- C. 子集树
- D. 排列树

156. 用贪心法求解背包问题时, 为了使收益最大化要选择 (A) 的物品装入背包。

- A. 单位重量收益最大
- B. 收益最大
- C. 重量最大
- D. 重量最小

157. 一个递归算法必须包括 (B)。

- A. 递归部分
- B. 终止条件和递归部分
- C. 迭代部分
- D. 终止条件和迭代部分

158. 有六个元素 6, 5, 4, 3, 2, 1 的顺序进栈, 则下列哪一个不是合法的出栈序列 (C)。

- A. 5 4 3 6 1 2
- B. 4 5 3 1 2 6
- C. 3 4 6 5 2 1
- D. 2 3 4 1 5 6

159. 栈和队列的共同点是 (C)。

- A. 都是先进先出
- B. 都是先进后出
- C. 只允许在端点处插入和删除元素
- D. 没有共同点

160. 若一棵二叉树具有 10 个度为 2 的结点, 5 个度为 1 的结点, 则度为 0 的结点个数是 ()。

- A. 9
- B. 11
- C. 15
- D. 不确定

161. 排序趟数与原始序列有关的排序方法是 (C) 排序法。

- A. 插入
- B. 选择
- C. 冒泡
- D. 快速

162. 如果给定权值总数有 n 个, 则其哈夫曼树的结点总数为 (D)。

- A. 不确定
- B. $2n$
- C. $2n+1$
- D. $2n-1$

163. 一个栈的输入序列为 $123\dots n$, 若输出序列的第一个元素是 n , 则输出第 i ($1 \leq i \leq n$) 个元素是 (B)。

- A. 不确定
- B. $n-i+1$
- C. i
- D. $n-i$

164. 下列序列中, (C) 是执行第一趟快速排序后所得的序列。

- A. [68, 11, 18, 69][23, 93, 73]
- B. [68, 11, 69, 23][18, 93, 73]
- C. [93, 73] [68, 11, 69, 23, 18]
- D. [68, 11, 69, 23, 18] [93, 73]

165. 下列哪个问题不能用贪心法求解? (C)

- A. 哈夫曼编码问题
- B. 单源最短路径问题
- C. 0-1 背包问题
- D. 最小生成树问题

166. 下列随机算法一定有解但解不一定正确的是 (C)。

- A. Sherwood
- B. LasVegas
- C. MonteCarlo
- D. 三者都不是

167. 在算法的三种情况下的复杂性中, 可操作性最好且最有实际价值的是 (B) 情况下的时间复杂度。

- A. 最好
- B. 最坏
- C. 平均
- D. 以上都不正确

168. 快速排序算法的性能取决于 (A)。

- A. 划分的对称性
- B. 数据的原始序列
- C. 随机选择策略
- D. 以上都不正确

169. 回溯法在问题的解空间树中, 按 (B) 策略, 从根节点出发搜索解空间树。

- A. 广度优先
- B. 深度优先
- C. 随机
- D. 以上说法都不对

170. 大 Ω 符号用来描述增长率的下限, 这个下限的阶越 (A), 结果就越有价值。

- A. 高

- B. 低
- C. 和具体问题有关
- D. 以上都不正确

171. 设计动态规划算法的步骤为：(1) 找出最优解的性质，并刻画其结构特征。(2) (B)。
(3) 以自底向上的方式计算出最优值。(4) 计算最优值得到的信息，构造最优解。

- A. 非递归的定义最优值
- B. 递归的定义最优值
- C. 迭代的定义最优值
- D. 递推的定义最优值

172. 设计动态规划算法的步骤为：(1) 找出最优解的性质，并刻画其结构特征。(2) 递归的定义最优值。(3) (A)。(4) 计算最优值得到的信息，构造最优解。

- A. 以自底向上的方式计算出最优值
- B. 以自顶向下的方式计算出最优值
- C. 迭代的方式计算出最优值
- D. 递推的方式算出最优值

173. 最优二叉搜索树即是 (A) 的二叉搜索树。

- A. 最小平均查找时间
- B. 最小最坏查找时间
- C. 最小平均存储空间
- D. 最小最坏存储空间

174. 当 $(a_1, a_2, a_3, a_4, a_5, a_6) = (-2, 11, -4, 13, -5, -2)$ 时，最大子段和为 (C)。

- A. 24
- B. 22
- C. 20
- D. 15

175. $9n^2 + 10n$ 的渐近表达式是 (A)。

- A. $O(n^2)$
- B. $O(n^3)$
- C. $O(n)$
- D. $O(n^4)$

176. 下面程序段的时间复杂度是 (C)。

```
for (i=0; i<n; i++)  
    for (j=0; j<n; j++)
```

- A. $O(n)$
- B. $O(n^3)$
- C. $O(n^2)$
- D. $O(n^4)$

177. 快速排序算法是基于分治策略的一个算法，其基本思想是，对于输入的子数组 $a[p:r]$ ，按以下三个步骤进行排序：（ A ）。

- A. 分解、递归求解、合并
- B. 递归求解、分解、合并
- C. 合并、递归求解、分解
- D. 分解、合并、递归求解

178. 最优装载问题可用贪心算法求解，采用（ C ）先装的贪心选择策略，可产生最优装载问题的最优解。

- A. 重量最重者
- B. 单位重量收益大
- C. 重量最轻者
- D. 收益最大

179. 写出下列 $f(n)$ 的渐进性态，若 $f(n)=C_0$ ， C_0 为常数，则 $f(n)=$ （ A ）。

- A. $O(1)$
- B. $O(n)$
- C. $O(n^2)$
- D. $O(n^3)$

180. 写出下列 $f(n)$ 的渐进性态，若 $f(n)=3n+2$ ，则 $f(n)=$ （ B ）。

- A. $O(1)$
- B. $O(n)$
- C. $O(n^2)$
- D. $O(n^3)$

181. 写出下列 $f(n)$ 的渐进性态，若 $f(n)=6*2^n+n$ ，则 $f(n)=$ （ B ）。

- A. $O(1)$
- B. $O(2^n)$
- C. $O(n^2)$
- D. $O(n^3)$

182. 若一个算法的时间复杂度用 $T(n)$ 表示，其中 n 的含义是（ A ）。

- A. 问题规模
- B. 语句条数
- C. 循环层数
- D. 函数数量

183. 背包问题可获得最优解的输入是按（ B ）。

- A. 重量密度排序
- B. 价值密度排序
- C. 单位重量收益大小排序

D. 重量大小排序

184. 合并排序法的基本思想是：将待排序元素分成大小大致相同的（ C ）个子集合，分别对每个子集合进行排序，最终将排好序的子集合合并成为所要求的排好序的集合。

- A. 4
- B. 3
- C. 2
- D. 5

185. 二分搜索算法的基本思想是将 n 个元素分成个数大致相同的两半，取 $a[n/2]$ 与 x 进行比较：如果（ C ），则只要在数组 a 的右半部继续搜索 x 。

- A. $x < a[n/2]$
- B. $x = a[n/2]$
- C. $x > a[n/2]$
- D. $x \geq a[n/2]$

186. 当问题规模 n 趋向于无穷大时，（ B ）的数量级（阶）称为算法的渐进时间复杂度。

- A. 空间复杂度
- B. 时间复杂度
- C. 冗余度
- D. 迭代次数

187. 通常用来表示时间算法的有以下六种多项式： $O(1)$, $O(n^3)$, $O(\log_2 n)$, $O(n^2)$, $O(n)$, $O(n \log_2 n)$, 按从小到大的顺序排列是（ A ）。

- A. $O(1) < O(\log_2 n) < O(n) < O(n \log_2 n) < O(n^2) < O(n^3)$
- B. $O(1) < O(n) < O(\log_2 n) < O(n \log_2 n) < O(n^2) < O(n^3)$
- C. $O(1) < O(\log_2 n) < O(n) < O(n^2) < O(n \log_2 n) < O(n^3)$
- D. $O(1) < O(n) < O(\log_2 n) < O(n \log_2 n) < O(n^2) < O(n^3)$

188. 贪心方法是一种求（ C ）的方法。

- A. 最小解
- B. 最大解
- C. 最优解
- D. 局部最优解

189. $O(Pf(N)) = O(f(N))$, 其中 P 是一个（ A ）。

- A. 正的常数
- B. 负的常数
- C. 不确定
- D. 以上说法都不对

190. 当问题规模 n 趋向于无穷大时，时间复杂度的数量级（阶）称为算法的（ C ）。

- A. 平均时间复杂度
- B. 最坏时间复杂度
- C. 渐进时间复杂度
- D. 最优时间复杂度

191. $f(n)=O(g(n))$ 表示当且仅当存在正的常数 C 和 N_0 ，使得对于所有的 $n \geq N_0$ ，有（ A ）。

- A. $f(n) \leq Cg(n)$
- B. $f(n) \geq Cg(n)$
- C. $f(n) > Cg(n)$
- D. $f(n) = Cg(n)$

192. 分支限界法与回溯法的相同点是：都是一种在问题的（ D ）中搜索问题解的算法。

- A. 子集树 T
- B. 排列树 T
- C. 二叉搜索树 T
- D. 解空间树 T

193. 对于分支限界法与回溯法，下面说法错误的是（ B ）。

- A. 求解目标不同
- B. 搜索方式相同
- C. 对扩展结点的扩展方式不同
- D. 存储空间的要求不同

194. 对于分支限界法与回溯法，下面说法正确的是（ D ）。

- A. 求解目标相同
- B. 搜索方式相同
- C. 对扩展结点的扩展方式相同
- D. 都是一种在问题的解空间树中搜索问题解的算法

195. 对于分治法与动态规划法，下面的说法正确的是（ C ）。

- A. 适合于用动态规划法求解的问题，经分解得到的子问题往往是相互独立的
- B. 使用分治法求解的问题，经分解得到的子问题往往不是相互独立的
- C. 适合于用动态规划法求解的问题，经分解得到的子问题往往不是相互独立的
- D. 分治法可以不需要将待求解问题分成若干个子问题

196. 舍伍德算法总能求得问题的（ A ）。

- A. 一个解

- B. 两个解
- C. 三个解
- D.三个以上的解

197. 采用舍伍德算法进行查找的时间复杂度为（ B ）。

- A. $O(1)$
- B. $O(n)$
- C. $O(n^2)$
- D. $O(\log_2 n)$

198. 回溯法搜索解空间树时，常用的两种剪枝函数为（ D ）和限界函数。

- A. 递归函数
- B. 迭代函数
- C. 非递归函数
- D. 约束函数

199. 分支限界法主要有（ A ）分支限界法和优先队列式分支限界法。

- A. 队列式（FIFO）
- B. 栈式
- C. 二叉树式
- D. 链式

200. 从分治法的一般设计模式可以看出，用它设计出的程序一般是（ B ）。

- A. 非递归算法
- B. 递归算法
- C. 数值算法
- D. 非数值算法

201. $n^2+10n-1$ 的渐进表达式为（ C ）。

- A. 1
- B. 2^n
- C. n^2
- D. n^3

202. $14+5/n+1/n^2$ 的渐进表达式为（ A ）。

- A. 14
- B. $1/n$
- C. $1/n^2$
- D. $5/n$

203. 下列不是基本计算模型的是（ B ）。

- A. RAM
- B. ROM
- C. RASP
- D. TM

204. 下列各步骤的先后顺序是（ B ）。①调试程序，②分析问题，③设计算法，④编写程序。

- A. ②③①④
- B. ②③④①
- C. ③②④①
- D. ③②①④

205. 贪心算法是一种只顾眼前的步骤，而难以顾忌全局步骤的算法，对于贪心算法表现出的特点，下面说法错误的是（ C ）。

- A. 不能保证最后求得的解是最佳的，即多半是近似解。（少数问题除外）
- B. 策略容易发现，而且运用简单，被广泛应用。
- C. 策略单一，结果也单一。
- D. 算法实现过程中，通常用到辅助算法：排序。

206. 贪心算法从初始阶段开始，每一个阶段总是作一个使（ D ）的贪心选择。

- A. 全局最优
- B. 局部最大
- C. 局部最小
- D. 局部最优

207. 对于 0-1 背包问题，用动态规划法的计算时间为（ A ）。

- A. $O(\min\{nc, 2^n\})$
- B. $O(\min\{nc\})$
- C. $O(\min\{2^n\})$
- D. $O(\min\{nc, 2n\})$

208. 动态规划算法的基本思想是将待求解问题分成若干（ B ）。

- A. 递归问题
- B. 子问题
- C. 小问题
- D. 非递归问题

209. 算法就是一组有穷的（ D ），它们规定了解决某一特定类型问题的一系列运算。

- A. 伪代码
- B. 指令
- C. 代码
- D. 规则

210. 算法的复杂性是（ A ）的度量，是评价算法优劣的重要依据。

- A. 算法效率
- B. 计算时间
- C. 存储空间
- D. 运行时间

211. 在算法的时间和空间关系上，（ A ）是决定性因素。

- A. 时间
- B. 空间
- C. 两者都是
- D. 以上说法都不对

212. 我们通常说的有效算法或实际可行算法是指（ B ）。

- A. 时间复杂度可以达到常数阶的算法
- B. 时间复杂度可以达到多项式时间的算法
- C. 时间复杂度可以达到对数阶的算法
- D. 时间复杂度可以达到指数阶的算法

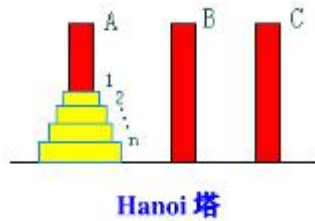
213. 对于 P 问题和 NP 问题，下面的关系正确的是（ A ）。

- A.
- B.
- C. $P=NP$
- D.

214. 应用 Johnson 法则的流水作业调度采用的算法是（ D ）。

- A. 贪心算法
- B. 分支限界法
- C. 分治法
- D. 动态规划算法

215. Hanoi 塔问题如下图所示。现要求将塔座 A 上的所有圆盘移到塔座 B 上，并按同样顺序叠置。移动圆盘时遵守 Hanoi 塔问题的移动规则。由此设计出 Hanoi 塔问题的递归算法正确的是（ B ）。



A. `void hanoi(int n, int A, int C, int B)`
`{ if (n>0)`
`{ hanoi(n-1, A, C, B);`
`move(n, a, b);`
`hanoi(n-1, C, B, A);`
`}`
`}`

B. `void hanoi(int n, int A, int B, int C)`
`{ if (n>0)`
`{ hanoi(n-1, A, C, B);`
`move(n, a, b);`
`hanoi(n-1, C, B, A);`
`}`
`}`

C. `void hanoi(int n, int C, int B, int A)`
`{ if (n>0)`
`{ hanoi(n-1, A, C, B);`
`move(n, a, b);`
`hanoi(n-1, C, B, A);`
`}`
`}`

C. `void hanoi(int n, int C, int A, int B)`
`{ if (n>0)`
`{ hanoi(n-1, A, C, B);`
`move(n, a, b);`
`hanoi(n-1, C, B, A);`
`}`
`}`

216. 动态规划算法的基本要素是（ C ）。

- A. 最优子结构性质与贪心选择性质
- B. 重叠子问题性质与贪心选择性质
- C. 最优子结构性质与重叠子问题性质

D. 预排序与递归调用

217. 算法分析中，记号 O 表示（ B ）。

- A. 渐进下界
- B. 渐进上界
- C. 非紧上界
- D. 紧渐近界

218. 算法分析中，记号 Ω 表示（ A ）。

- A. 渐进下界
- B. 渐进上界
- C. 非紧上界
- D. 紧渐近界