1. 二分搜索算法是利用（ ）实现的算法。

A.分治策略

B.动态规划法

C.贪心法

D.回溯法

2. 回溯法解旅行售货员问题时的解空间树是（ ）。

A.子集树

B.排列树

C.深度优先生成树

D.广度优先生成树

3. 下列算法中通常以自底向上的方式求解最优解的是（ ）。

A.备忘录法

B.动态规划法

C.贪心法

D.回溯法

4. 下面不是分支界限法搜索方式的是（ ）。

A.广度优先

B.最小耗费优先

C.最大效益优先

D.深度优先

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为（ ）。

A.O（n2n）

B.O（）

C.O（2n）

D.O（）

6. 分支限界法求解最大团问题时，活结点表的组织形式是（ ）。

A.最小堆

B.最大堆

C.栈

D.数组

7. 下面问题（ ）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

8. 下列算法中不能解决0/1 背包问题的是（ ）。

A.贪心法

B.动态规划

C.回溯法

D.分支限界法

9. 背包问题的贪心算法所需的计算时间为（ ）。

A.O（n2n）

B.O（nlogn）

C.O（2n）

D.O（n）

10. 二分查找是利用（ ）实现的算法。

A. 分治策略

B. 动态规划法

C. 分支限界法

D. 概率算法

11. 下列不是动态规划算法基本步骤的是（ ）。

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

12. 最大效益优先是（ ）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

13. 在下列算法中有时找不到问题解的是（ ）。

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

14. 对于动态规划，下面说法错误的是（ ）。

A.动态规划（dynamic programming）是运筹学的一个分支

B.动态规划是求解决策过程（decision process）最优化的数学方法

C.虽然动态规划主要用于求解以时间划分阶段的动态过程的优化问题，但是一些与时间无关的静态规划(如线性规划、非线性规划)，只要人为地引进时间因素，把它视为多阶段决策过程，也可以用动态规划方法方便地求解。

D. 动态规划类似搜索或数值计算那样，具有一个标准的数学表达式和明确清晰的解题方法。

15. 衡量一个算法好坏的标准是（ ）。

A. 运行速度快

B. 占用空间少

C. 时间复杂度低

D. 代码短

16. 以下不可以使用分治法求解的是（ ）。

A.棋盘覆盖问题

B.选择问题

C.归并排序

D.0/1背包问题

17. 实现循环赛日程表利用的算法是（ ）。

A.分治策略

B.动态规划法

C.贪心法

D.回溯法

18. 下列随机算法中运行时有时候成功有时候失败的是（ ）。

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

19. 对于分支限界法，下面不属于分支限界法搜索方式的是（ ）。

A. 广度优先

B. 最小耗费优先

C. 最大效益优先

D. 层次优先

20. 下列算法中通常以深度优先方式系统搜索问题解的是（ ）。

A. 备忘录法

B. 动态规划法

C. 贪心法

D. 回溯法

21. 备忘录方法是那种算法的变形（ ）。

A.分治法

B.动态规划法

C.贪心法

D.回溯法

22. 哈夫曼编码的贪心算法所需的计算时间为（ ）。

A. O(n2n)

B. O(nlogn)

C. O(2n)

D. O(n)

23. 最长公共子序列算法利用的算法是（ ）。

A.分支界限法

B.动态规划法

C.贪心法

D.回溯法

24. 实现棋盘覆盖算法利用的算法是（ ）。

A.分治法

B.动态规划法

C.贪心法

D.回溯法

25. 下面是贪心算法的基本要素的是（ ）。

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（ ）。

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（ ）。

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（ ）。

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（ ）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

30. 下列哪一种算法不是随机化算法（ ）。

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

31. （ ）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.最优子结构性质

32. 矩阵连乘问题的算法可由（ ）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

33. Strassen矩阵乘法是利用（ ）实现的算法。

A.分治策略

B.动态规划法

C.贪心法

D.回溯法

34. 使用分治法求解不需要满足的条件是（ ）。

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

35. 回溯法搜索状态空间树是按（ ）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

36. 实现合并排序利用的算法是（ ）。

A.分治策略

B.动态规划法

C.贪心法

D.回溯法

37. 下列是动态规划算法基本要素的是（ ）。

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

38. 采用广度优先策略搜索的算法是（ ）。

A.分支限界法

B.动态规划法

C.贪心法

D.回溯法

39. 在下列算法中得到的解未必正确的是（ ）。

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

40. 实现大整数的乘法是利用的算法（ ）。

A.贪心法

B.动态规划法

C.分治策略

D.回溯法

41. 0/1 背包问题的回溯算法所需的计算时间为（ ）。

A. O（n2n）

B. O（nlogn）

C. O（2n）

D. O（n）

42. 动态规划算法与贪心法的主要区别是（ ）。

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

43. 实现最大子段和利用的算法是（ ）。

A.分治策略

B.动态规划法

C.贪心法

D.回溯法

44. 优先队列式分支限界法选取扩展结点的原则是（ ）。

A.先进先出

B.后进先出

C.结点的优先级

D.随机

45. 广度优先是（ ）的一种搜索方式。

A.分支限界算法

B.动态规划法

C.贪心算法

D.回溯算法

46. 舍伍德算法是（ ）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

47. 对于下列算法，有时找不到问题解的是（ ）。

A.蒙特卡罗算法

B.拉斯维加斯算法

C.快速排序算法

D.数值概率算法

48. 下列哪一种算法是随机化算法（ ）。

A.贪心算法

B.回溯法

C.动态规划算法

D.舍伍德算法

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（ ）。

A.重叠子问题

B.最优子结构性质

C.贪心选择性质

D.定义最优解

50. 以深度优先方式系统搜索问题解的算法称为（ ）。

A.分支界限算法

B.概率算法

C.贪心算法

D.回溯算法

51. 对于最长公共子序列，下面说法错误的是（ ）。

A. 最长公共子序列，英文缩写为LCS（Longest Common Subsequence）。其定义是，一个序列 S ，如果分别是两个或多个已知序列的子序列，且是所有符合此条件序列中最长的，则 S 称为已知序列的最长公共子序列。

B.最长公共子序列是一个十分实用的问题，它可以描述两段文字之间的“相似度”。

C.最长公共子串和最长公共子序列是不同的

D.最长公共子串和最长公共子序列是相同的

52. 当一个算法的空间复杂度与问题的规模n成正比时，则表示为（ ）。

A.O(1)

B.O(n)

C.O(n\*n)

D.O1

53. 当一个算法的时间复杂性与问题的规模n大小无关时，则表示为（ ）。

A.1

B.O(1)

C.O(n)

D.O(0)

54. 算法分析的两个主要方面是（ ）。

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（ ）。

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（ ）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.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（ ）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.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（ ），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>=a[n/2]

61. 活动安排问题就是在所给的活动集合中，选出（ ）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（ ）。

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（ ）。

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（ ）。

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（ ）。

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（ ）存储结构。

A.堆

B.顺序

C.任意次序

D.栈

67. 应用分治法的两个前提是（ ）。

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（ ）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（ ）：}

A. n\*factorial(n)

B. n\*factorial(n-1)

C. n\*factorial(n-2)

D. n\*factorial(n+1)

70. （ ）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.舍伍得算法

71. 对于n个元素的排序问题。n＝2时只要作（ ）次比较即可排好序。

A.3

B.2

C.1

D.4

72. 一般地讲，当一个问题的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：（ ）。

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（ ）。

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. 递归算法不适用以下场合（ ）。

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（ ）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最关键特征是（ ）。

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（ ）的货箱然后选（ ）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. (Math.abs(k-j)==Math.abs(x[j]-x[k]))||x[j]==x[k])

B. (Math.abs(k-j)==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（ ）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（ ），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. 在任何一个的棋盘覆盖中，用到的L型骨牌个数恰为（ ）。

A.

B.

C.

D.

82. 以 Bitonic旅行路线问题为例，动态规划的时间复杂度为（ ）。

A. O(n)

B. O(n!)

C. O(n2)

D. O(n3)

83. 一个算法应该包含如下几条性质除了（ ）。

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（ ）。

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（ ）。

A.5n

B.20log2n

C.2n

D.3nlog3n

86. 渐进算法分析是指（ ）。

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（ ）。**

**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（ ）。

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（ ）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（ ）。

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（ ）

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0/1背包问题和背包问题的解法下面（ ）

A．0/1背包问题和背包问题都可用贪心算法求解

B．0/1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．0/1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0/1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（ ）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（ ）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（ ）数据结构来实现。

A．栈

B．堆

C．队列

D．二叉查找树

96. 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（ ）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（ ）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（ ）。

A. 的渐进表达式上界函数是

B．的渐进表达式下界函数是

C．的渐进表达式上界函数是

D．的渐进表达式下界函数是

99. 当输入规模为n时，算法增长率最大的是（ ）。

A．

B．

C．

D．

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（ ）。

A．T(n)=T(n-1)+1，T(1)=1

B．T(n)=2n2

C. T(n)=T(n/2)+1，T(1)=1

D. T(n)=3nlog2n

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（ ）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水流恒定。如下（ ）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（ ）。

A. 2n-1

B. 2n

C. 2n+1-1

D. 2n+1

104.以下关于判定问题难易处理的叙述中正确的是（ ）。

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.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N0，使得当N≥N0时有f(N)≤ 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（ ）g(N)的阶。

A.不高于

B.不低于

C.等价于

D.逼近

107.以下（ ）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（ ）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（ ）。

A.正确性

B.简明性

C.无限性

D.最优性

110.算法分析是（ ）。

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111.学校要举行运动会，请你设计111. 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（ ）。

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（ ）。

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（ ）。

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（ ）。

A. O(n2)

B. O(mlogn)

C. O(nlogm)

D. O(mlogm)

115.算法与程序的区别在于算法具有（ ）。

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,…,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（ ）。

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（ ）。

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. O(2n)

B. O(nlogn)

C. 

D. 

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（ ）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（ ）。

A. 12n

B.

C. 2n2

D.

121.关于0/1背包问题，以下描述正确的是（ ）。

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,…, 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ti。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（ ）。

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.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.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（ ）。

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. O(n)

B. O(logn)

C. O(n2)

D. O(nlogn)

127.折半查找的时间复杂度为（ ）。

A. O(n)

B. O(logn)

C. O(n2)

D. O(nlogn)

128.算法的复杂性有时间复杂性和（ ）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（ ）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（ ）是清晰的，无歧义的。

A.程序

B.指令

C.语句

D.语句块

131.（ ）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. 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（ ）。

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（ ）。

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（ ）的一种排序算法。

A.分治策略

B.贪心

C.回溯

D.动态规划

136.（ ）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.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（ ）。

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（ ）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（ ）。

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（ ）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.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（ ）。

A.递归

B.分治

C.迭代

D.模拟。

144.FIFO是（ ）的一搜索方式。

A.分支限界法

B.动态规划法

C.贪心法

D.回溯法

145.投点法是（ ）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（ ）。

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（ ）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. 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（ ）。

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（ ）。

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（ ）。

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（ ）。

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.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（ ）。

A. O(n2)

B. O(n3)

C. O(n)

D. O(n4)

155.二分搜索过程的算法行为可以用一棵（ ）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（ ）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（ ）。

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（ ）。

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（ ）。

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，那么度为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（ ）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（ ）。

A.不确定

B.2n

C.2n+1

D.2n-1

163.一个栈的输入序列为123…n，若输出序列的第一个元素是n, 那么输出第i（）个元素是（ ）。

A. 不确定

B. n-i+1

C. i

D. n-i

164.下列序列中，（ ）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（ ）。

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（ ）。

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（ ）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.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（ ）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.大符号用来描述增长率的下限，这个下限的阶越（ ），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）（ ）。（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）（ ）。（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（ ）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(a1, a2, a3, a4, a5,a6)=(-2, 11, -4, 13, -5, -2)时，最大子段和为（ ）。

A.24

B.22

C.20

D.15

175. 9n2+10n的渐近表达式是（ ）。

A. O(n2)

B. O(n3)

C. O(n)

D. O(n4)

176.下面程序段的时间复杂度是（ ）。

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

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

A. O(n)

B. O(n3)

C. O(n2)

D. O(n4)

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.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（ ）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)= C0，C0为常数，则f(n)=（ ）。

A.O(1)

B.O(n)

C.O(n2)

D.O(n3)

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

A. O(1)

B. O(n)

C. O(n2)

D. O(n3)

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

A. O(1)

B. O(2n)

C. O(n2)

D. O(n3)

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.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（ ）。

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（ ）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（ ），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>=a[n/2]

186.当问题规模n趋向于无穷大时，（ ）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(n3), O(log2n), O(n2), O(n), O(nlog2n),按从小到大的顺序排列是（ ）。

A. O(1)< O(log2n)< O(n)< O(nlog2n)<O(n2)<O(n3)

B. O(1)< O(n) <O(log2n)< O(nlog2n)<O(n2)<O(n3)

C. O(1)< O(log2n)< O(n)< O(n2) <O(nlog2n)<O(n3)

D. O(1)< O(n) <O(log2n)<O(nlog2n)<O(n2)<O(n3)

188.贪心方法是一种求（ ）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.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（ ）。

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N0，使得对于所有的n>=N0, 有（ ）。

A. f(n)<=Cg(n)

B. f(n)>=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（ ）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（ ）。

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（ ）。

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（ ）。

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.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（ ）。

A.O(1)

B.O(n)

C.O(n2)

D.O(log2n)

198.回溯法搜索解空间树时，常用的两种剪枝函数为（ ）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（ ）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（ ）。

A.非递归算法

B.递归算法

C.数值算法

D.非数值算法

201.n2+10n-1的渐进表达式为（ ）。

A.1

B.2n

C.n2

D.n3

202.14+5/n+1/n2的渐进表达式为（ ）。

A.14

B.1/n

C.1/n2

D.5/n

203.下列不是基本计算模型的是（ ）。

A.RAM

B.ROM

C.RASP

D.TM

204.下列各步骤的先后顺序是（ ）。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（ ）。

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（ ）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. O(min{nc,2n})

B. O(min{nc})

C. O(min{2n})

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（ ）。

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（ ），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.语言

C.代码

D.规则

210.算法的复杂性是（ ）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，（ ）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（ ）。

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. P类问题包含在NP类问题中

B. NP类问题包含在P类问题中

C.P=NP

D. NP完全问题是P类问题的子集

214.应用Johnson法则的流水作业调度采用的算法是（ ）。

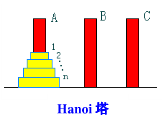
A.贪心算法

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（ ）。



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);

}

}

D. 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（ ）。

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（ ）。

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.渐进下界

B.渐进上界

C.非紧上界

D.紧渐近界

219.分支限界法在问题的解空间树中，按（ ）策略，从根结点出发搜索解空间树。

A.广度优先

B.深度优先

C.活结点优先

D.扩展结点优先

220.回溯法的效率不依赖于下面的哪一个因素（ ）。

A.产生x[k]的时间

B.满足显约束的x[k]值的个数

C.问题的解空间的形式

D.计算上界函数bound的时间

221.常见的两种分支限界法为（ ）。

A.广度优先分支限界法与深度优先分支限界法

B.队列式（FIFO）分支限界法与堆栈式分支限界法

C.排列树法与子集树法

D.队列式（FIFO）分支限界法与优先队列式分支限界法

222.用回溯法解0/1背包问题时，该问题的解空间结构为（ ）结构。

A.子集树

B.排列树

C.深度优先生成树

D.广度优先生成树

223.用回溯法解批处理作业调度问题时，该问题的解空间结构为（ ）结构。

A.子集树

B.排列树

C.深度优先生成树

D.广度优先生成树

224.用回溯法解题的一个显著特征是在搜索过程中动态产生问题的解空间。在任何时刻，算法只保存从根结点到当前扩展结点的路径。如果解空间树中从根结点到叶结点的最长路径的长度为h(n), 则回溯法所需的计算空间通常为（ ）。

A. O(n)

B. O(n2)

C. O(h(n))

D. O(h(n)+n)

225.O(f(n))+O(g(n))=（ ）。

A. O(f(n))

B. O(g(n))

C. O(max{f(n), g(n)})

D. O(min{f(n), g(n)})

226.记号O的定义正确的是（ ）。

A.O(g(n))={f(n)|存在正常数c和n0使得所有n大于等于n0有：f(n)大于等于0，并且f(n)小于等于cg(n)}

B.O(g(n))={f(n)|存在正常数c和n0使得所有n大于等于n0有：cg(n)大于等于0，并且cg(n)小于等于f(n)}

C.O(g(n))={f(n)|对于任何正常数c>0，存在正数和n0大于0，使得对所有n大于等于n0有，f(n)大于等于0，并且f(n)小于等于cg(n)}

D.O(g(n))={f(n)|对于任何正常数c>0，存在正数和n0大于0，使得对所有n大于等于n0有：}

227.记号Ω的定义正确的是（ ）。

A.Ω(g(n))={f(n)|存在正常数c和n0使得所有n大于等于n0有：f(n)大于等于0，并且f(n)小于等于cg(n)}

B.Ω(g(n))={f(n)|存在正常数c和n0使得所有n大于等于n0有：cg(n)大于等于0，并且cg(n)小于等于f(n)}

C.Ω(g(n))={f(n)|对于任何正常数c>0，存在正数和n0大于0，使得对所有n大于等于n0有，f(n)大于等于0，并且f(n)小于等于cg(n)}

D.Ω(g(n))={f(n)|对于任何正常数c>0，存在正数和n0大于0，使得对所有n大于等于n0有：}

228.下面的程序段所需要的计算时间为（ ）。

int Maxsum(int n, int \*a, int &besti, int &bestj)

{

int sum=0;

for (int i=1; i<=n; i++) {

int thissum=0;

for (int j=1; j<=n; j++) {

thissum+=a[j];

if (thissum>sum) {

sum=thissum;

besti=i;

bestj=j;

}

}

}

return sum;

}

A.O(1)

B.O(n)

C.O(n2)

D.O(n3)

229.用回溯法解图的m着色问题时，使用下面的函数OK检查当前扩展节点的每一个儿子所相应的颜色的可用性，则需耗时（渐近时间上限）为（ ）。

Bool Color::OK(int k)

{//

for (int j=1; j<=n; j++)

if ((a[k][j]==1)&&(x[j]==x[k])) return false;

return true;

}

A. O(m)

B. O(n)

C. O(mn)

D. O(m2n2)

230.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与（ ）算法框架。

A.排列树

B.二叉树

C.深度优先生成树

D.广度优先生成树

231.回溯法的算法框架按照问题的解空间一般分为排列树算法框架与（ ）算法框架。

A.二叉树

B.子集树

C.深度优先生成树

D.广度优先生成树

232.现有Hanoi塔问题的递归方程为：h(n)=2h(n-1)+1，h(1)=1则h(n)的非递归表达式为（ ）。

A.2n-1

B.2n-1

C.2n+1

D.2n

233.3n2+10n的渐进表达式为（ ）。

A.n2

B.10n

C.3n

D.3n2

234.21+1/n的渐进表达式为（ ）。

A. 21

B. 1/n

C. n

D. n2

235.对于下面的两个函数：f(n)=logn2 ; g(n)=logn+5, 下列关系正确的是（ ）。

A. *f*（*n*）＝O（*g*（*n*））

B. *f*（*n*）＝Ω（*g*（*n*））

C. *f*（*n*）＝Θ（*g*（*n*））

D.以上的关系都不对

236.对于下面的两个函数：f(n)=logn2，g(n)=n1/2下列关系正确的是（ ）。

A. *f*（*n*）＝O（*g*（*n*））

B. *f*（*n*）＝Ω（*g*（*n*））

C. *f*（*n*）＝Θ（*g*（*n*））

D.以上的关系都不对

237.对于下面的两个函数：f(n)=n ; g(n)=log2n, 下列关系正确的是（ ）。

A. *f*（*n*）＝O（*g*（*n*））

B. *f*（*n*）＝Ω（*g*（*n*））

C. *f*（*n*）＝Θ（*g*（*n*））

D.以上的关系都不对

238.对于下面的两个函数：f(n)=nlogn ; g(n)=logn, 下列关系正确的是（ ）。

A. *f*（*n*）＝O（*g*（*n*））

B. *f*（*n*）＝Ω（*g*（*n*））

C. *f*（*n*）＝Θ（*g*（*n*））

D.以上的关系都不对

239.对于蒙特卡罗算法，下面的说法不正确的是（ ）。

A.蒙特卡罗算法用于求解问题的准确解，且该解一定是正确的

B.求得正确解的概率依赖于算法的计算时间

C.多次执行蒙特卡罗算法，可以提高获得正确解的概率

D.无法有效判定所得到的解是否肯定正确

240.对于拉斯维加斯算法，下面的说法不正确的是（ ）。

A.不会得到不正确的解

B.有时找不到问题的解

C.找到正确解的概率随算法计算时间的增加而提高

D.用同一拉斯维加斯算法对同一问题求解多次，对求解失败的概率没有影响

241.对于舍伍德算法，下面的说法不正确的是（ ）。

A.总能求得问题的一个解

B.不一定能求得问题的解

C.所求得的解总是正确的

D.将确定性算法引入随机性改造成舍伍德算法，可消除或减少问题对于好坏实例间的差别

242.对于数值概率算法，下面的说法不正确的是（ ）。

A.常用于数值问题的求解，得到的往往是近似解

B.解的精度随计算时间的增加而提高

C.解的精度和计算时间之间没有关系

D.在很多情况下，计算出问题的精确解是不可能或没必要

243.下面哪个不属于算法的三要素（ ）。

A.操作

B.控制结构

C.数据结构

D.程序

244.下面哪个不属于算法设计的质量指标（ ）。

A.正确性

B.可读性

C.健壮性

D.有穷性

245.对于迭代法，下面的说法不正确的是（ ）。

A.需要确定迭代模型

B.需要建立迭代关系式

C.需要对迭代过程进行控制，要考虑什么时候结束迭代过程

D.不需要对迭代过程进行控制

246.对于高级语言，下面的说法不正确的是（ ）。

A. 高级语言更接近算法语言，易学，易掌握

B. 高级语言为程序员提供了结构化程序设计的环境和工具

C. 高级语言依赖于机器语言

D. 高级语言不依赖于机器语言

247.算法的时间复杂度的性能为当n=1时，f(n)=1，当n大于等于2时，f(n)=8f(3n/7),则算法的时间复杂度的阶为（ ）。

A. n2

B. log2n

C. n

D.nlog7/38

248.当一个算法的运行时间为n2+n+1时，由于n2+n+1与n2的数量级相等，则称n2为这个算法的（ ）。

A.渐进时间复杂度或时间复杂度

B.空间复杂度

C.问题规模

D.输入规模

249.多项式A(n)=amnm+…+a2n2+a1n+a0的上界为（ ）。

A.O(nm)

B.O(nm-1)

C.O(nm-2)

D.O(a0)

250.折半查找、合并排序、二叉树遍历等算法中均采用了（ ）策略。

A.动态规划

B.回溯

C.分治

D.贪心选择

251.递归算法设计的关键在于找出递归关系和（ ）。

A.初始值

B.递归终止（边界）条件

C.递归方程

D.递归函数入口

252.（ ）是问题能用贪婪算法或动态规划算法求解的前提。

A.无后效性

B.问题规模不能太大

C.时间复杂度不能太高

D.空间复杂度不能太高

253.回溯算法是尝试搜索算法中最为基本的一种算法，其采用了一种（ ）的思想作为其控制结构。

A.深度优先搜索

B.广度优先搜索

C.不能走就掉头

D.分治

254.算法分析的目的是（ ）。

A.找出数据结构的合理

B.研究算法中的输入输出关系

C.分析算法效率以求改进

D.分析算法的易懂性和文档性

255.一个算法必须保证在执行有限步之后结束，这是算法的（ ）。

A.有穷性

B.确定性

C.可行性

D.输出

256.算法原则上能够精确地执行，而且我们用纸和笔做有限次的运算后即可完成，这是算法的（ ）特性。

A.有穷性

B.确定性

C.可行性

D.输出

257.通常，最适合描述算法的语言是（ ）。

A.自然语言

B.数学公式

C.计算机程序设计语言

D.介于自然语言和程序设计语言之间的伪语言

258.对于反复多次使用的程序，应尽量选用（ ）算法。

A.节约空间

B.节约时间

C.简明易懂

D.容易调试

259.评价一个算法时间性能的主要指标是（ ）。

A.算法易于调试

B.算法易于理解

C.算法的稳定性和正确性

D.算法的时间复杂度

260.A算法的时间复杂度为O(n3)，B算法的时间复杂度为O(2n)，则说明（ ）。

A.对于任何数据量，A算法的时间开销都比B算法小

B.随着问题规模n的增大，A算法比B算法有效

C.随着问题规模n的增大，B算法比A算法有效

D.对于任何数据量，B算法的时间开销都比A算法小

261.对于算法的时间复杂度来说，可操作性最好、最有实用价值的是（ ）。

A.最好情况下的时间复杂度

B.最坏情况下的时间复杂度

C.平均时间复杂度

D.问题规模较大时的平均时间复杂度

262.能否利用分治法完全取决于（ ）。

A.该问题的规模缩小到一定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，即该问题具有最优子结构性质

C.该问题分解出的子问题的解可以合并为该问题的解

D.该问题分解出的各个子问题是相互独立的，即子问题之间不包含公共的子问题

263.两个2×2的矩阵的乘积，（ ）次乘法是必要的。

A.6

B.7

C.8

D.9

264.递归方程T(n)=3T(n/2)+2n1.5，T(n)=1，则T(n)=（ ）。

A.O(n)

B.O(n2)

C.O(nlog3)

D.O(nlog5)

265.贪心算法能获得（ ）。

A.一定是最优解

B.不一定是最优解

C.次最优解

D.次最差解

266.启发式算法能获得的是（ ）。

A.最差解

B.最优解

C.次最优解

D.次最差解

267.如果有一个问题，它的过程可以分为若干阶段，而且对于任一阶段i，过程在i阶段以后的行为仅仅依赖于i阶段的状态，而与过程如何达到此种状态（即达到的方式）无关，则称之为（ ）。

A.状态转换

B.动态规划法

C.最佳性原理

D.一个多阶段的决策过程

268.当需要找出它的解集，或者要求回答什么解是满足某些条件的最佳解时，往往要使用（ ）。

A.分治法

B.贪心法

C.动态规划法

D.回溯法

269.下面的（ ）方法是基于穷举法的。

A.分治法

B.贪心法

C.动态规划法

D.回溯法

270.将问题分支为子问题，采用广度优先产生状态空间树的结点，并使用剪枝函数对这些子问题限界而求解问题的方法称为（ ）。

A.分治法

B.回溯法

C.动态规划法

D.分支限界法

271.分支限界法只能应用于解决（ ）。

A.一般背包问题

B.最优化问题

C.可行解

D.非最优化问题

272.（ ）的一个基本特征是用同一概率算法求解问题的同一实例两次，得到的结果可能完全不同。

A.贪心算法

B.回溯算法

C.概率算法

D.近似算法

273.使用（ ）求解问题，通常是求解问题的准确解。

A.数值概率算法

B.蒙特卡罗算法

C.舍伍德算法

D.拉斯维加斯算法

274.在最坏情况下，对于一个具有n个项的有序表，二分检索算法用了（ ）次比较。

A.O(n)

B.O(n2)

C.O(log2n)

D.O(nlog2n)

275.程序是算法用某种程序设计语言的具体实现，它可以不满足（ ）。

A.有穷性

B.确定性

C.可行性

D.输入，输出

276.算法一般分为两类：精确算法和（ ）。

A.概率算法

B.非数值算法

C.近似算法（启发式算法）

D.数值算法

277.下面程序段的时间复杂度为（ ）。

for (i=0; i<m; i++)

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

A.O(m2)

B.O(n2)

C.O(m\*n)

D.O(m+n)

278.设两个算法在同一台机器上运行，其执行时间分别是n2和2n，要使后者快于前者，当n取多少时才行（ ）。

A. 2

B. 3

C. 4

D. 5

279.执行下面的程序段时，执行S语句的次数为（ ）。

for (i=1; i<=n; i++)

for (j=1; j<=i; j++) S；

A.n2

B.n2/2

C.n(n+1)

D.n(n+1)/2

280.对于下面的时间复杂度关系排序，正确的是（ ）。

A.O(2n)<O(n!)<O(nn)

B.O(n!)<O(2n)<O(nn)

C.O(2n)<O(nn)<O(n!)

D.O(n!) <O(nn) <O(2n)

281.求下面式子的渐进表达式：logn5，其渐进表达式为（ ）。

A.O(n)

B.O(logn)

C.O(n2)

D.O(n!)

282.求下面式子的渐进表达式：6log4n，其渐进表达式为（ ）。

A.O(n)

B.O(logn)

C.O(n2)

D.O(n!)

283.假设某算法在输入规模为n时的计算时间为T(n)=3×2n。在某台计算机上实现并完成该算法的时间为t秒。现有另一台计算机，其运行速度为第一台的64倍，那么在这台新机器上用同一算法在t秒内能解输入规模为多大的问题（ ）。

A.n+1

B.n+2

C.n+4

D.n+6

284.假设某算法在输入规模为n时的计算时间为T(n)=n2。在某台计算机上实现并完成该算法的时间为t秒。现有另一台计算机，其运行速度为第一台的64倍，那么在这台新机器上用同一算法在t秒内能解输入规模为多大的问题（ ）。

A.2n

B.6n

C.8n

D.10n

285.硬件厂商XYZ公司宣称他们最新研制的微处理器运行速度为其竞争对手 ABC公司同类产品的100倍。对于计算复杂性为n的算法，若用ABC公司的计算机在1小时内能解输入规模为n的问题，那么用XYZ公司的计算机在1小时能解输入规模为多大的问题（ ）。

A.100n

B.10n

C.n

D.n2

286.硬件厂商XYZ公司宣称他们最新研制的微处理器运行速度为其竞争对手 ABC公司同类产品的100倍。对于计算复杂性为n2的算法，若用ABC公司的计算机在1小时内能解输入规模为n的问题，那么用XYZ公司的计算机在1小时能解输入规模为多大的问题（ ）。

A.100n

B.10n

C.n

D.n2

287.硬件厂商XYZ公司宣称他们最新研制的微处理器运行速度为其竞争对手 ABC公司同类产品的100倍。对于计算复杂性为n3的算法，若用ABC公司的计算机在1小时内能解输入规模为n的问题，那么用XYZ公司的计算机在1小时能解输入规模为多大的问题（ ）。

A.n

B.2n

C.4.64n

D.n2

288.对于分治法，下面的描述错误的是（ ）。

A.分治法将一个难以直接解决的大问题，分割成一些规模较小的类型相同的问题，这些子问题相互独立，以便各个击破，分而治之。

B.如果原问题可以分割成m个子问题，并且这些子问题都可解，然后求解这些子问题，那么就可以用这些子问题的解求出原问题的解。

C.如果子问题还比较复杂而不能直接求解，还可以继续细分，直到子问题足够小，能够直接求解为止。

D.子问题只能细分一次，不能继续细分。

289.对于已按升序排好序的元素，采用二分检索，下面的描述错误的是（ ）。

A.将待查找的数据与升序数组中的中间元素进行比较，若二者相等则表示找到

B.若待查找的数据小于中间元素的值，则下次在数组的前半部分中继续找

C.若待查找的数据大于中间元素的值，则下次在数组的后半部分中继续找

D.若待查找的数据大于中间元素的值，则下次在数组的前半部分中继续找

290.所谓贪心法的（ ），是指可以根据该度量标准，实行多步地决策进行求解，虽然在该度量意义下所做的这些选择都是局部最优的，最终得到的解却不一定是全局最优的。

A.最优度量标准

B.平均度量标准

C.最差度量标准

D.随机度量标准

291.下面哪个问题不适合用贪心法解决（ ）。

A.背包问题

B.磁带的最优存储

C.有限期的作业调度问题

D.汉诺塔问题

292.最坏情况下，合并两个大小为n的已排序数组所需要的比较次数为（ ）。

A.2n-1

B.2n

C.2n+1

D.2n-2

293.回溯法中常见的两类典型的解空间树是子集树和排列树。当所给的问题是从n个元素的集合S中找出满足某种性质的子集时，相应的解空间树称为子集树。这类子集树通常有（ ）个叶结点。

A. 2n

B. 2n-1

C. 2n+1

D. 2n-2

294.回溯法中常见的两类典型的解空间树是子集树和排列树。当所给的问题是从n个元素的集合S中找出满足某种性质的子集时，相应的解空间树称为子集树。这类子集树通常有2n个叶结点,遍历子集树通常需要（ ）计算时间。

A.O(n)

B.O(logn)

C.O(n2)

D.O(2n)

295.回溯法中常见的两类典型的解空间树是子集树和排列树。当所给的问题是确定n个元素满足某种性质的排列时，相应的解空间树称为排列树。这类排列树通常有（ ）个叶结点。

A.n!

B.2n

C.2n+1

D.2n-1

296.回溯法中常见的两类典型的解空间树是子集树和排列树。当所给的问题是确定n个元素满足某种性质的排列时，相应的解空间树称为排列树。这类排列树通常有n!个叶结点。遍历排列树需要（ ）的计算时间。

A.O(n!)

B.O(logn)

C.O(n2)

D.O(2n)

297.给定已按升序排好序的n个元素a[0:n-1]，现要在这n个元素中找出一特定元素x，返回其在数组中的位置，如果未找到返回-1。若采用二分搜索的算法，其时间复杂度为（ ）。

A.O(n!)

B.O(logn)

C.O(n2)

D.O(2n)

298.采用分治法实现合并排序，其时间复杂度为（ ）。

A.O(n!)

B.O(logn)

C.O(nlogn)

D.O(2n)

299.对于字符串的子串（substring）和子序列（subsequence），下面的描述错误的是（ ）。

A.子串：字符串中任何一串连续的字符都是这个字符串的子串。

B.子序列：字符串中一串字符的组合。

C.子序列：字符串中一串字符的组合，这些字符不一定是连续的，但字符的顺序与原字符串相同。

D.子序列：字符串中一串字符的组合，这些字符不一定是连续的，字符的顺序可以与原字符串不同。

300.对于算法复杂性的描述，下面的说法错误的是（ ）。

A.导致算法难解性的原因1：算法过于复杂，只存在指数级时间复杂度的算法。

B.导致算法难解性的原因2：问题的答案本身的长度大于任何由输入数目组成的多项式。

C.每个NP问题都是难解的。

D.并不是每个NP问题都是难解的。

301.当问题的规模递增时，将复杂度的极限称为算法的（ ）。

A.渐进复杂度

B.最坏复杂度

C.平均复杂度

D.最优复杂度

302.对于5n2+8n，其渐进表达式为（ ）。

A.O(1)

B.O(n)

C.O(n2)

D.O(logn)

303.对于3n2/11+3n，其渐进表达式为（ ）。

A.O(1)

B.O(n)

C.O(n2)

D.O(3n)

304.对于56+3/n，其渐进表达式为（ ）。

A.O(1)

B.O(n)

C.O(n2)

D.O(3n)

305.对于logn5，其渐进表达式为（ ）。

A.O(1)

B.O(n)

C.O(n2)

D.O(logn)

306.对于6log4n，其渐进表达式为（ ）。

A.O(1)

B.O(n)

C.O(n2)

D.O(logn)

307.按照渐进阶从低到高的顺序排列下列表达式：5n2，logn，3n，45n，6，3n3/2，n!，下面的关系正确的是（ ）。

A. 6<logn<45n<3n3/2<5n2<3n<n!

B. 6<logn<45n<3n3/2<5n2<n!<3n

C. 6<45n<logn<3n3/2<5n2<3n<n!

D. 6<logn<3n3/2<45n<5n2<3n<n!

308.确定关系：对于下列各组函数*f*（*n*）和*g*（*n*），确定*f*（*n*）＝O（*g*（*n*））或*f*（*n*）＝Ω（*g*（*n*））或*f*（*n*）＝（*g*（*n*））。对于*f*（*n*）=logn2, *g*（*n*）=logn+7，下列关系正确的是（ ）。

A. *f*（*n*）＝O（*g*（*n*））

B. *f*（*n*）＝Ω（*g*（*n*））

C. *f*（*n*）＝Θ（*g*（*n*））

309.确定关系：对于下列各组函数*f*（*n*）和*g*（*n*），确定*f*（*n*）＝O（*g*（*n*））或*f*（*n*）＝Ω（*g*（*n*））或*f*（*n*）＝（*g*（*n*））。对于*f*（*n*）=logn2, *g*（*n*）=n1/2，下列关系正确的是（ ）。

A. *f*（*n*）＝O（*g*（*n*））

B. *f*（*n*）＝Ω（*g*（*n*））

C. *f*（*n*）＝Θ（*g*（*n*））

310.确定关系：对于下列各组函数*f*（*n*）和*g*（*n*），确定*f*（*n*）＝O（*g*（*n*））或*f*（*n*）＝Ω（*g*（*n*））或*f*（*n*）＝（*g*（*n*））。对于*f*（*n*）=n, *g*（*n*）=log2n，下列关系正确的是（ ）。

A. *f*（*n*）＝O（*g*（*n*））

B. *f*（*n*）＝Ω（*g*（*n*））

C. *f*（*n*）＝Θ（*g*（*n*））

311.确定关系：对于下列各组函数*f*（*n*）和*g*（*n*），确定*f*（*n*）＝O（*g*（*n*））或*f*（*n*）＝Ω（*g*（*n*））或*f*（*n*）＝（*g*（*n*））。对于*f*（*n*）=nlogn+n, *g*（*n*）=logn，下列关系正确的是（ ）。

A. *f*（*n*）＝O（*g*（*n*））

B. *f*（*n*）＝Ω（*g*（*n*））

C. *f*（*n*）＝Θ（*g*（*n*））

312.确定关系：对于下列各组函数*f*（*n*）和*g*（*n*），确定*f*（*n*）＝O（*g*（*n*））或*f*（*n*）＝Ω（*g*（*n*））或*f*（*n*）＝（*g*（*n*））。对于*f*（*n*）=11, *g*（*n*）=log11，下列关系正确的是（ ）。

A. *f*（*n*）＝O（*g*（*n*））

B. *f*（*n*）＝Ω（*g*（*n*））

C. *f*（*n*）＝Θ（*g*（*n*））

313.确定关系：对于下列各组函数*f*（*n*）和*g*（*n*），确定*f*（*n*）＝O（*g*（*n*））或*f*（*n*）＝Ω（*g*（*n*））或*f*（*n*）＝（*g*（*n*））。对于*f*（*n*）=log2n, *g*（*n*）=logn，下列关系正确的是（ ）。

A. *f*（*n*）＝O（*g*（*n*））

B. *f*（*n*）＝Ω（*g*（*n*））

C. *f*（*n*）＝Θ（*g*（*n*））

314.确定关系：对于下列各组函数*f*（*n*）和*g*（*n*），确定*f*（*n*）＝O（*g*（*n*））或*f*（*n*）＝Ω（*g*（*n*））或*f*（*n*）＝（*g*（*n*））。对于*f*（*n*）=2n, *g*（*n*）=100n2，下列关系正确的是（ ）。

A. *f*（*n*）＝O（*g*（*n*））

B. *f*（*n*）＝Ω（*g*（*n*））

C. *f*（*n*）＝Θ（*g*（*n*））

315.确定关系：对于下列各组函数*f*（*n*）和*g*（*n*），确定*f*（*n*）＝O（*g*（*n*））或*f*（*n*）＝Ω（*g*（*n*））或*f*（*n*）＝（*g*（*n*））。对于*f*（*n*）=2n, *g*（*n*）=3n，下列关系正确的是（ ）。

A. *f*（*n*）＝O（*g*（*n*））

B. *f*（*n*）＝Ω（*g*（*n*））

C. *f*（*n*）＝Θ（*g*（*n*））

316.对于n!和nn，下面的关系正确的是（ ）。

A. n!=O(nn)

B. n!= Ω(nn)

C. n!= Θ(nn)

D.以上关系都不对

317.如果一个算法在平均情况下的计算时间复杂度为Θ(f(n))，则该算法在最坏情况下所需的计算时间为（ ）。

A. O(f(n))

B. Ω(f(n))

C. Θ(f(n))

D. 以上都不对

318.以下是NP问题的是（ ）。

A.0/1背包问题

B.哈密顿回路

C.求最大公因子

D.顶点覆盖

319.带图灵机可形式化地描述为一个7元组(Q, T, I, δ, b, q0, qf)，其中Q表示（ ）。

A.转移函数

B.状态集合

C.输入字母表

D.拒绝状态

320.根据转移函数是单值的还是（ ）的，图灵机可以分为确定型和不确定型两类。

A.多值

B.二值

C.递归

D.非递归

321.根据转移函数是单值的还是多值的，图灵机可以分为确定型和（ ）两类。

A.不确定型

B.随机型

C.动态型

D.变化型

322.P中的一切问题都是（ ）可解的。

A.指数时间

B.多项式时间

C.常数阶时间

D.对数阶时间

323.（ ）就是指在解决优化问题中，最后得到的结果能保证在一定的误差之类的算法。

A.概率算法

B.近似算法

C.并行算法

D.加密算法

324.针对许多不能在多项式时间内求解的NP完全问题，提出求解接近（ ）的相似解来代替最优解。

A.相似解

B.精确解

C.最优解

D.次最优解

325.针对许多不能在多项式时间内求解的NP完全问题，提出求解接近精确解的（ ）来代替最优解。

A.相似解

B.精确解

C.最优解

D.次最优解

326.针对许多不能在多项式时间内求解的NP完全问题，提出求解接近精确解的相似解来代替（ ）。

A.相似解

B.精确解

C.最优解

D.次最优解

327.通常，对那些无法在多项式时间内求解的NP完全问题采用（ ）。

A.近似算法

B.概率算法

C.递归算法

D.非递归算法

328.回溯法主要有递归回溯法和（ ）回溯法两种编程实现方法。

A.迭代

B.非递归

C.并行

D.深度优先

329.回溯法主要有迭代回溯法和（ ）回溯法两种编程实现方法。

A.非递归

B.递归

C.并行

D.深度优先

330.对于0/1背包问题和普通的背包问题，下列说法错误的是（ ）。

A. 0/1背包问题装入的物品是不可拆分的

B.背包问题装入的物品是可拆分的

C. 0/1背包问题装入的物品是可拆分的

D. 0/1背包问题可用动态规划的思想求解

331.算法由（ ）三要素组成。

A.操作，控制结构，数据结构

B.程序，控制结构，数据结构

C.代码，控制结构，数据结构

D.问题规模，控制结构，数据结构

332.按照时间复杂度从低到高排列：O(4n2)、O(logn)、O(3n)、O(20n)、O(2)、O(n2/3)、O(n!),

则O(n!)应该排在哪一位（ ）。

A.4

B.5

C.6

D.7

333.算法是指解决问题的（ ）。

A.方法或过程

B.思想

C.流程或步骤

D.程序

334.计算下面程序段的时间复杂度为（ ）。

for (i=1; i<n; i++)

{ y=y+1;

for (j=0; j<=2n; j++)

x++;

}

A.O(n)

B.O(n2)

C.O(n3)

D.O(1)

335.直接或间接地调用自身的算法称为递归算法，用函数自身给出定义的函数称为（ ）。

A.递归函数

B.非递归函数

C.迭代函数

D.递推函数

336.楼梯上有n个台阶，上楼时可以上1步，也可以上2步，设计一递归算法求出共有多少种上楼方法F(n)，则F(n)的递归表达式为（ ）。

A.当n=1时，F(n)=1;当n=2时，F(n)=2;当n>2时，F(n)=F(n-1)+F(n-2)

B.当n=1时，F(n)=1;当n=2时，F(n)=2;当n>2时，F(n)=F(n-1)

C.当n=1时，F(n)=1;当n=2时，F(n)=2;当n>2时，F(n)=F(n-2)

D.当n=1时，F(n)=0;当n=2时，F(n)=2;当n>2时，F(n)=F(n-1)+F(n-2)

337.在快速排序，插入排序和合并排序算法中，（ ）算法不是分治算法。

A.快速排序

B.插入排序

C.合并排序

338.合并排序算法使用的是（ ）算法设计的思想

A.分治

B.动态规划法

C.贪心法

D.回溯法

339.回溯算法的效率不依赖于（ ）。

A.问题的规模

B.可行解的个数

C.最优解的个数

D.输入的数据

340.下列关于算法的叙述中，错误的是（ ）。

A.一个算法至少有一个输入和一个输出

B.算法的每一个步骤必须确切地定义

C.一个算法在执行有穷步后必须结束

D.算法中有待执行的运算和操作必须是相当基本的

341.在使用流程图描述算法时，表示变量的计算与赋值应使用的符号框为（ ）。

A.矩形框

B.菱形框

C.平行四边形框

D.椭圆形框

342.一个爱好程序设计的同学，想通过程序设计解决“韩信点兵”的问题，他制定的如下工作过程中，最恰当的是（ ）。

A.设计算法，编写程序，提出问题，运行程序，得到答案

B.分析问题，编写程序，设计算法，运行程序，得到答案

C.分析问题，设计算法，编写程序，运行程序，得到答案

D.设计算法，提出问题，编写程序，运行程序，得到答案

343.回溯算法和分支限界法的问题的解空间不会是（ ）。

A.线性表

B.子集树

C.排列树

344.回溯算法和分支限界法的问题的解空间不会是（ ）。

A.排列树

B.子集树

C.图

345.n2/10+2n的渐进表达式为（ ）。

A. O(n2)

B. O(1)

C. O(n)

D. O(2n)

346.10log3n的渐进表达式为（ ）。

A.O(n)

B.O(logn)

C.O(1)

D.O(3n)

347.当c=50，n=3，w={20，10，50}时，最优装载问题的解为（ ）。

A.(1, 1, 0)

B.(0, 0, 1)

C.(0, 1, 1)

D.(1, 0, 0)

348.当n=3，c=50，w={10，15，30}，v={60，100，120}，0/1背包问题的最优解为（ ）。

A.(1, 1, 0)

B.(0, 0, 1)

C.(0, 1, 1)

D.(1, 0, 0)

349.当n=3，c=50，w={10，15，30}，v={60，100，120}，背包问题的最优解为（ ）。

A.(1, 1, 0)

B.(1, 1, 5/6)

C.(0, 1, 1)

D.(1, 0, 0)

350.算法复杂性的高低体现在运行该算法所需要的计算机资源上。对于计算机资源，最重要的是时间和（ ）资源。

A.内存

B.硬盘

C.空间

D.文件

351.分治法的思想是将原问题划分成（ ）的小问题，然后逐个解决，最终合并得到元问题的解。

A.2个

B.3个

C.4个

D.多个（n个）

352.下面程序段的时间复杂度为（ ）。

i=1;

k=0;

while (i<n) {k=k+10\*i; i++}

A.O(1)

B.O(n)

C.O(n2)

D.O(n3)

353.程序可以不满足以下（ ）特征。

A.输入

B.输出

C.确定性

D.有限性

354.以下（ ）不包括在图灵机结构中

A.控制器

B.读写磁头

C.计算器

D.磁带

355.设有n=2k个运动员要进行循环赛，现设计一个满足以下要求的比赛日程表：

（1）：每个选手必须与其他n-1名选手比赛各一次；

（2）：每个选手一天至多只能赛一次；

（3）：循环赛要在最短时间内完成。

如果n=2k，循环赛最少需要进行（ ）天。

A.2k-1

B.2k

C.2k+1

D.2k-2

356.某个问题的最优解包含着其子问题的最优解，这种性质称为（ ）。

A.贪心选择性质

B.最优子结构性质

C.递归性质

D.动态规划性质

357.回溯法的基本思想是在一棵含有问题全部可能解的（ ）上进行深度优先搜索，解为叶子结点。

A.子集树

B.排列树

C.二叉树

D.状态空间树

358.P问题，也即是（ ）复杂程度的问题。

A.多项式

B.指数

C.对数

D.常数

359.NP就是Non-deterministic Polynomial的问题，也即是多项式复杂程度的（ ）问题。

A.非确定性

B.近似

C.概率

D.确定性

360.下面哪个问题不适合用回溯法（ ）。

A.装载问题

B.0/1背包问题

C.图的m着色问题

D.快速排序问题

361.给定n种物品和一个背包，物品i的重量是wi，其价值为vi，背包的容量为 C。选择装入背包的物品，对于每种物品i只有两种选择，即装入背包或不装入背包，不能将物品i装入背包多次，也不能只装入部分的物品i，最终要使得装入背包中物品的总价值最大。该问题被称为（ ）。

A.装载问题

B.最优装载问题

C.0/1背包问题

D.背包问题

362.对于哈夫曼编码，下面的说法不正确的是（ ）。

A.哈夫曼编码是由哈夫曼提出的构造最优前缀码的贪心算法，由此产生的编码方案称为哈夫曼编码。

B.哈夫曼编码是广泛地用于数据文件压缩的十分有效的编码方法。其压缩率通常在20%～90%之间。

C.哈夫曼编码给出现频率高的字符较短的编码，出现频率较低的字符以较长的编码，可以大大缩短总码长。

D.哈夫曼编码不依赖于信源的统计特性，编码前不用先统计得到信源的概率特性。

363.给定一个无向连通图G和m种不同的颜色。用这些颜色为图G的各顶点着色，每个顶点着一种颜色。是否有一种着色法使G中每条边的2个顶点着有不同颜色。这个问题是（ ）问题。

A.图的m可着色判定

B.四色猜想

C.最大团

D.最小生成树

364.给定一个带权有向图G=(V, E)，其中每条边的权是非负实数。然后选定V中的一个顶点，现在要计算从该顶点到其它各顶点的最短路的长度。这里路的长度是指路上各边权之和。这个问题通常称为（ ）。

A.图的m着色问题

B.单源最短路径问题

C.最小生成树问题

D.最大团问题

365.阶乘函数可递归地定义为（ ）。

A.当n=0时，n!=1;当n>0时，n!=n(n-1)!

B.当n=0时，n!=0;当n>0时，n!=n(n-1)!

C.当n=0时，n!=1;当n>0时，n!=(n-1)(n-2)!

D.当n=0时，n!=1;当n>0时，n!=n(n+1)!

366.无穷数列1, 1, 2, 3, 5, 8, 13, 21, 34, 55,……,称为Fibonacci数列，它可以递归地定义为（ ）。

A.当n=0时，F(n)=0;当n=1时，F(n)=1;当n大于1时，F(n)=F(n-1)+F(n-2)

B.当n=0时，F(n)=1;当n=1时，F(n)=1;当n大于1时，F(n)=F(n-1)

C.当n=0时，F(n)=1;当n=1时，F(n)=1;当n大于1时，F(n)=F(n-2)

D.当n=0时，F(n)=1;当n=1时，F(n)=1;当n大于1时，F(n)=F(n-1)+F(n-2)

367.下面哪个问题不太适合用递归算法（ ）。

A. 阶乘函数

B. Hanoi塔问题

C. Fibonacci数列

D. 简单的四则运算

368.设G=(V, E)是连通带权图，V={1,2,…,n}。采用Prim算法，能得到G的一棵（ ）。

A.哈夫曼树

B.子集树

C.最小生成树

D.最优二叉树

369.“比较”排序算法时间复杂度的下界为（ ）。

A.O(logn)

B.O(n)

C.O(n2)

D.O(nlogn)

370.对于归并排序，下面的说法错误的是（ ）。

A.归并排序是建立在归并操作上的一种有效的排序算法。

B.归并排序算法是采用分治法（Divide and Conquer）的一个非常典型的应用。

C.归并排序平均时间和最坏时间都较长，速度慢。

D.归并排序是一种稳定的排序方法。

371.回溯法在问题的解空间树中，按深度优先策略，从（ ）出发搜索解空间树。

A.叶子结点

B.中间结点

C.根结点

D.任意一个结点

372.对于分治法与动态规划，下面说法错误的是（ ）。

A.分治法是将一个问题划分成一系列独立的子问题，分别处理后将结果组合以得到原问题的答案。

B.动态规划同样将一个问题划分成一系列子问题进行处理。

C.当子问题间不是互相独立而是互有联系时，动态规划不会重复计算子问题间联系的问题。

D.动态规划的效率比分治法低。

373.对贪心算法的描述，下面说法错误的是（ ）。

A.贪心算法的思想是通过选择局部最优以求得最优解

B.某些最优问题无法有局部最优推出

C.贪心算法可以解决0/1背包问题

D.贪心算法对有的问题是有效的，对有的问题是无效的

374.对于Hanoi塔问题，下面的说法错误的是（ ）。

A. Hanoi塔问题源于印度一个古老的传说

B. Hanoi塔问题是用递归求解的一个经典问题

C. Hanoi塔问题只能用递归方法求解，不能用非递归方法求解

D. 假设有n个圆盘，总共需要移动2n-1次

375.设一个栈的输入序列为A,B,C,D，则借助一个栈所得到的输出序列不可能是（ ）。

A. A, B, C, D

B. D, C, B, A

C. A, C, D, B

D. D, A, B, C

376.以下程序段的时间复杂度为（ ）。

int fun(int n)

{ int i,j,k,s;

s=0;

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

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

for (k=0;k<=j;k++)

s++;

return(s);

}

A. O(n3)

B. O(n2)

C. O(n)

D. O(n4)

377.以下程序段的时间复杂度为（ ）。

void func(int n)

{ int i=0,s=0;

while (s<n)

{ i++;

s=s+i;

}

}

A.O(n3)

B.O(n2)

C.O(n1/2)

D.O(n4)

378.有如下算法：调用下述算法的语句为fun(a,n,0)，其时间复杂度为（ ）。

void fun(int a[],int n,int k) //数组a共有n个元素

{ int i;

if (k==n-1)

for (i=0;i<n;i++) 　　//n次

printf("%d\n",a[i]);

else

{ for (i=k;i<n;i++)　　//n-k次

a[i]=a[i]+i\*i;

fun(a,n,k+1);

}

}

A.O(n3)

B.O(n2)

C.O(n1/2)

D.O(n4)

379.以下程序段的时间复杂度为（ ）。

int fun(int a[ ], int n, int k)

{ int i;

i=0;

while (i<n &&a[i]!=k)

i++;

return(i);

}

A.O(n3)

B.O(n2)

C.O(n)

D.O(n4)

380.以下程序段的时间复杂度为（ ）。

int fun(int a[ ], int n, int i)

{ int j, max=a[0];

for (j=1; j<=i-1; j++)

if (a[j]>max) max=a[j];

return(max);

}

A.O(n3)

B.O(n2)

C.O(n)

D.O(n4)

381.数据运算的执行（ ）。

A.效率与采用何种存储结构有关

B.是根据存储结构来定义的

C.有算术运算和关系运算两大类

D.必须用程序设计语言来描述

382.以下（ ）不是算法的基本特性。

A.可行性

B.长度有限

C.在确定的时间完成

D.确定性

383.下面关于算法的说法正确的是（ ）。

A.算法最终必须由计算机程序实现

B.一个算法所花时间等于该算法中每条语句的执行时间之和

C.算法的可行性是指指令不能有二义性

D.以上说法都是错误的

384.算法的时间复杂度与（ ）有关。

A.问题规模

B.计算机硬件性能

C.编译程序质量

D.程序设计语言

385.算法分析的主要任务之一是分析（ ）。

A.算法是否具有较好的可读性

B.算法中是否存在语法错误

C.算法的功能是否符合设计要求

D.算法的执行时间和问题规模之间的关系

386.某算法的时间复杂度为O(n2)，表明该算法的（ ）。

A.问题规模是n2

B.执行时间等于n2

C.执行时间与n2成正比

D.问题规模与n2成正比

387.递归函数f(1)=1，f(n)=f(n-1)+n（n>1）的递归出口是（ ）。

A. f(1)=1

B. f(1)=0

C. f(0)=0

D. f(n)=n

388.递归函数f(1)=1，f(n)=f(n-1)+n（n>1）的递归体是（ ）。

A. f(1)=1

B. f(0)=0

C. f(n)=f(n-1)+n

D. f(n)=n

389.将递归算法转换成对应的非递归算法时，通常需要使用（ ）保存中间结果。

A.队列

B.栈

C.链表

D.树

390.将f=1+1/2+1/3+…+1/n转化成递归函数，若其递归体是f(n)=f(n-1)+1/n(n大于等于2)，则其递归出口是（ ）。

A. f(1)=1

B. f(0)=1

C. f(1)=1/2

D. f(0)=1/2

391.将f=1+1/2+1/3+…+1/n转化成递归函数，若其递归出口是f(1)=1，则其递归体为（ ）。

A. f(n)=f(n-1)+1/n (n大于等于1)

B. f(n)=f(n-1)+1/n (n大于等于2)

C. f(n)=f(n)+1/n (n大于等于2)

D. f(n)=f(n-1)-1/n (n大于等于2)

392.有如下递归过程：

void reverse(int m)

{ printf(“%d”,n%10);

if (n/10!=0)

reverse(n/10);

}，则调用语句reverse(582)的结果是（ ）。

A.581

B.583

C.258

D.285

393.以下算法的时间复杂度为（ ）。

void fun(int n)

{ int y=0;

while (y\*y<=n)

y++;

}

A.O(n)

B.O(1)

C.O(n2)

D.O(n1/2)

394.以下算法的时间复杂度为（ ）。

void fun(int n)

{ int i, j, x=0;

for (i=1;i<n;i++)

for (j=i+1;j<=n;j++)

x++;

}

A. O(n)

B. O(n2)

C. O(n3)

D. O(1)

395.以下为各算法所有语句频度之和的表达式，其中时间复杂度相同的是（ ）。

(1). TA(n)=2n3+3n2+1000 (2). TB(n)=n3-n2log2n-1000

(3). TC(n)=n2log2n+n2 (4). TD(n)=n2+1000

A.(1)和(2)

B.(1)和(4)

C.(3)和(4)

D.(2)和(3)

396.设n是偶数，下列算法的时间复杂度为（ ）。

void fun(int n)

{ int m=0, i, j;

for (i=1; i<=n; i++)

for (j=2\*i; j<=n; j++)

m++;

}

A.O(n)

B.O(n2)

C.O(n3)

D.O(1)

397.下列算法的时间复杂度为（ ）。

void fun(int n)

{ int s=0, i, j, k;

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

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

for(k=0; k<j;k++)

s++;

}

A.O(n)

B.O(n2)

C.O(n3)

D.O(1)

398.线性表是（ ）。

A.一个有限序列，可以为空

B.一个有限序列，不可以为空

C.一个无限序列，可以为空

D.一个无限序列，不可以为空

399.在一个长度为n的顺序表中于第i个元素之前插入一个新元素，需要向后移动（ ）个元素。

A. n-i

B. n-i+1

C. n-i-1

D. i

400.链表不具有的特点是（ ）。

A.可随机访问任一元素

B.插入删除不需要移动元素

C.不必事先估计存储空间

D.所需空间与线性表长度成正比

401.线性表采用链式存储结构时，各节点之间的地址（ ）。

A.必须是连续的

B.一定是不连续的

C.部分地址必须是连续的

D.连续与否均可以

402.在线性表的下列存储结构中，读取指定序号的元素花费时间最少的是（ ）。

A.单链表

B.双链表

C.循环链表

D.顺序表

403.在线性表的如下链式存储结构中，若未知链表头节点的地址，仅已知p指针指向的节点，哪种结构不能删除该节点（ ）。

A.单链表

B.双链表

C.循环单链表

404.对于栈和队列，下面说法错误的是（ ）。

A.栈是后进先出

B.队列是先进先出

C.只允许在端点处插入和删除元素

D.没有共同点

405.元素a、b、c、d依次进顺序栈后，栈顶元素是（ ）。

A. a

B. b

C. c

D. d

406.元素a、b、c、d依次进顺序栈后，栈底元素是（ ）。

A. a

B. b

C. c

D. d

407.经过以下运算后，x的值是（ ）。

InitStack(s); Push(s,a); Push(s,b);Pop(s,x);GetTop(s,x)

A. a

B. b

C. 1

D. 0

408.经过以下栈运算后，StackEmpty(s)的值是（ ）。

InitStack(s); Push(s,a); Push(s,b); Pop(s,x); Pop(s,y)

A. a

B. b

C. 1

D. 0

409.设一个栈的输入序列为A，B，C，D，则借助一个栈所得到的输出序列不可能是（ ）。

A. A，B，C，D

B. D，C，B，A

C. A，C，D，B

D. D，A，B，C

410.已知一个栈的进栈序列是1，2，3，…，n，其输出序列是p1，p2，…，pn，若p1=3，则p2的值（ ）。

A.一定是2

B.一定是1

C.不可能是1

D.以上都不对

411.已知一个栈的进栈序列是1，2，3，…，n，其输出序列是p1，p2，…，pn，若p1=n，则pi的值为（ ）。

A. i

B. n-i

C. n-i+1

D. 不确定

412.设n个元素进栈序列是p1，p2，p3，…，pn，其输出序列是1，2，3，…，n，若pn=1，则pi(1≤i≤n−1)的值是（ ）。

A. n-i+1

B. n-i

C. i

D.有多种可能

413.判定一个顺序栈ST（元素个数最多为StackSize）为空的条件为（ ）。

A. ST.top== -1

B. ST.top!= -1

C. ST.top!=StackSize

D. ST.top==StackSize

414.经过以下队列运算后，队头的值是

InitQueue(qu); enQueue(qu, a); enQueue(qu, b); enQueue(qu, c); deQueue(qu)

A. a

B. b

C. 1

D. 0

415.经过以下队列运算后，QueueEmpty(q)值是（ ）。

InitQueue(qu); enQueue(qu, a); enQueue(qu, b); deQueue(qu, x); deQueue(qu, y)

A. a

B. b

C. 1

D. 0

416.元素a、b、c、d依次进入队列qu后，队头元素是（ ）。

A. a

B. b

C. c

D. d

417.元素a、b、c、d依次进入队列qu后，队尾元素是（ ）。

A. a

B. b

C. c

D. d

418.设环形队列中数组的下标是0~N-1，其头、尾指针分别为f（指向队头元素的前一个位置）和r（指向队尾元素），则其元素个数为（ ）。

A. r-f

B. r-f-1

C. (r-f)%N+1

D. (r-f+N)%N

419.设a是含有n个元素的整数数组，假定数组a的下标从0开始，则n个整数之和的递归定义为（ ）。

A. f(a, 1)=a[0]

f(a, i)=f(a, i-1)+a[i-1] i>1

B. f(a, 1)=a[1]

f(a, i)=f(a, i-1)+a[i-1] i>1

C. f(a, 1)=a[0]

f(a, i)=f(a, i1)+a[i-1] i>1

D. f(a, 1)=a[0]

f(a, i)=f(a, i-1)+a[i-2] i>1

420.设a是含有n个元素的整数数组，假定数组a的下标从0开始，则n个整数之积的递归定义为（ ）。

A. f(a, 1)=a[0]

f(a, i)=f(a, i-1)\*a[i-1] i>1

B. f(a, 1)=a[1]

f(a, i)=f(a, i-1)\*a[i-1] i>1

C. f(a, 1)=a[0]

f(a, i)=f(a, i1)\*a[i-1] i>1

D. f(a, 1)=a[0]

f(a, i)=f(a, i-1)\*a[i-2] i>1

421.以下算法是计算两个正整数u和v最大公因数的递归函数，其递归模型为（ ）。

int gcd(int u, int v)

{ int r;

if ((r=u%v)==0)

return(v);

else

return(gcd(u,r));

}

A. gcd(u,v)=v 当u%v=0时

gcd(u,v)=gcd(u,u/v) 其他情况

B. gcd(u,v)=v 当u%v=0时

gcd(u,v)=gcd(u,u%v) 其他情况

C. gcd(u,v)=v 当u%v!=0时

gcd(u,v)=gcd(u,u%v) 其他情况

D. gcd(u,v)=u 当u%v=0时

gcd(u,v)=gcd(u,u%v) 其他情况

422.编写一个递归算法，读入一个字符串（以“.”作为结束），要求打印出他们的倒序字符串。下面的算法正确的是（ ）。

A. void reverse( )

{ char ch;

scanf(“%c”,&ch);

{ reverse();

printf(“%c”, ch);

}

}

B. void reverse( )

{ char ch;

scanf(“%d”,&ch);

if (ch!=’.’)

{ reverse();

printf(“%d”, ch);

}

}

C. void reverse( )

{ char ch;

scanf(“%c”,&ch);

if (ch!=’\0’)

{ reverse();

printf(“%c”, ch);

}

}

D. void reverse( )

{ char ch;

scanf(“%c”,&ch);

if (ch!=’.’)

{ reverse();

printf(“%c”, ch);

}

}

423.在一个无向图中，所有顶点的度之和等于边数的（ ）倍。

A.1/2

B.1

C.2

D.4

424.一个有n个顶点的无向图最多有（ ）条边。

A. n

B. n(n-1)

C. n(n-1)/2

D. 2n

425.一个有n个顶点的有向图最多有（ ）条边。

A.n

B.n(n-1)

C.n(n-1)/2

D.2n

426.具有6个顶点的无向图至少应有（ ）条边才可能是一个连通图。

A.5

B.6

C.7

D.8

427.在一个具有n个顶点的无向图中，要连通全部顶点至少需要（ ）条边。

A.n

B.n+1

C.n-1

D.n/2

428.只有在顺序存储结构上才能实现的查找方法是（ ）法。

A.顺序查找

B.二分查找

C.树形查找

D.哈希查找

429.在数据元素有序、元素个数较多而且固定不变的情况下，宜采用（ ）法。

A.二分查找

B.分块查找

C.二叉排序树查找

D.顺序查找

430.有一个长度为12的有序表，按二分查找法对该表进行查找，在表内各元素等概率的情况下，查找成功时所需的平均比较次数为（ ）。

A.35/12

B.37/12

C.39/12

D.43/12

431.有一个有序表R[1..13]={1,3,9,12,32,41,45,62,75,77,82,95,100}，当用二分查找法查找值为82的节点时，经过（ ）次比较后查找成功。

A.1

B.2

C.4

D.8

432.在一棵3次树中度为3的节点数为2，度为2的节点数为1，度为1的节点数为2，则度为0的节点数为（ ）。

A.4

B.5

C.6

D.7

433.一棵度为4的树T中，若有20个度为4的节点，10个度为3的节点，1个度为2的节点，10个度为1的节点，则树T的叶子节点个数是（ ）。

A.41

B.82

C.113

D.122

434.采用分块查找时，若线性表中共有625个元素，查找每个元素的概率相同，假设采用顺序查找来确定节点所在的快，则每块分为（ ）个节点最佳。

A.9

B.25

C.6

D.625

435.对一组序列{48，36，68，99，75，24，28，52}进行快速排序，要求结果从小到大排序，则进行一次划分之后的结果为（ ）。

A. (24 28 36)48(52 68 75 99)

B. (28 36 24)48(75 99 68 52)

C. (36 68 99)48(75 24 28 52)

D. (28 36 24)48(99 75 68 52)

436.若一组记录的关键字为{46,79,56,38,40,84}，则利用快速排序的方法，以第1个记录为基准得到的一次划分结果为（ ）。

A. 38，40，46，56，79，84

B. 40，38，46，79，56，84

C. 40，38，46，56，79，84

D. 40，38，46，84，56，79

437.依次将待排序序列中的元素插入到有序子序列中并扩大有序子序列的排序方法是（ ）。

A.快速排序

B.直接插入排序

C.冒泡排序

D.堆排序

438.下列排序方法中，在待排序的数据已经为有序时，花费时间反而最多的是（ ）。

A.快速排序

B.希尔排序

C.冒泡排序

D.堆排序

439.快速排序算法在（ ）情况下最不利于发挥其长处。

A.要排序的数据量太大

B.要排序的数据中含有多个相同值

C.要排序的数据已基本有序

D.要排序的数据个数为奇数

440.一组记录的关键字为{25,48,16,35,79,82,23,40,36,72}，其中，含有5个长度为2的有序表，按归并排序的方法对该序列进行第二趟归并后的结果为（ ）。

A. 16, 25, 35, 48, 23, 40, 79, 82, 36, 72

B. 16, 25, 35, 48, 79, 82, 23, 36, 40, 72

C. 16, 25, 48, 35, 79, 82, 23, 36, 40, 72

D. 16, 25, 35, 48, 79, 23, 36, 40, 72, 82

441.用某种排序方法对线性表{25, 84, 21, 47, 15, 27, 68, 35, 20}进行排序时，元素序列的变化如下：

（1）25, 84, 21, 47, 15, 27, 35, 68, 20

（2）21, 25, 47, 84, 15, 27, 35, 68, 20

（3）15, 21, 25, 27, 35, 47, 68, 84, 20

（4）15, 20, 21, 25, 27, 35, 47, 68, 84

其采用的排序方法是（ ）。

A.简单选择排序

B.希尔排序

C.归并排序

D.快速排序

442.在归并排序中，若待排序记录的个数为20，则共需要进行（ ）趟归并。

A.5

B.4

C.3

D.2

443.在排序过程中，任何情况下都不比较关键字大小的排序方法是（ ）。

A.直接插入排序

B.冒泡排序

C.快速排序

D.基数排序

444.出自于“平衡子问题”的思想，通常分治法在分割原问题，形成若干子问题时，这些子问题的规模都（ ）。

A.完全不同

B.大致不同

C.必须完全相同

D.大致相同

445.使用二分搜索算法在n个有序元素表中搜索一个特定元素，在最佳情况下，搜索的时间复杂性为（ ）。

A.O(1)

B.O(log2n)

C.O(n)

D.O(n2)

446.使用二分搜索算法在n个有序元素表中搜索一个特定元素，在最坏情况下，搜索的时间复杂性为（ ）。

A.O(1)

B.O(log2n)

C.O(n)

D.O(n2)

447.若线性表最常用的算法是存取第i个元素及其前驱的值，则采用（ ）存储方式比较节省时间。

A.单链表

B.双链表

C.循环单链表

D.顺序表

448.动态规划方法有一个变形方法（ ），这种方法不同于动态规划算法“自底向上”的填充方向，而是“自顶向下”的递归方向，为每个解过的子问题建立了备忘录，以备需要时查看，同样也可避免相同子问题的重复求解。

A.备忘录方法

B.贪心法

C.线性规划方法

D.分治法

449.在一个具有n个节点的有序单链表中插入一个新节点使得仍然有序，其算法的时间复杂度为（ ）。

A. O(log2n)

B. O(1)

C. O(n2)

D. O(n)

450.数据的运算（ ）。

A.效率与采用何种存储结构有关

B.是根据存储结构来定义的

C.有算术运算和关系运算两大类

D.必须用程序设计语言来描述

451.链表不具备的特点是（ ）。

A.可随机访问任一节点

B.插入删除不需要移动元素

C.不必事先估计存储空间

D.所需空间与其长度成正比

452.在顺序表中删除一个元素的时间复杂度为（ ）。

A.O(1)

B.O(log2n)

C.O(n)

D.O(n2)

453.以下线性表的存储结构中具有随机存取功能的是（ ）。

A.不带头节点的单链表

B.带头节点的单链表

C.循环双链表

D.顺序表

454.在存储数据时，通常不仅要存储各数据元素的值，而且还要存储（ ）。

A.数据的处理方法

B.数据元素的类型

C.数据元素之间的关系

D.数据的存储方法

455.下述函数中对应的时间复杂度（n为问题规模）最小的是（ ）。

A.T1(n)=nlog2n+5000n

B.T2(n)=n2-8000n

C.T3(n)=nlog2n-6000n

D.T4(n)=100000log2n

456.设线性表有n个元素，在以下操作中，（ ）在顺序表上实现比在链表上实现效率更高。

A.输出第i(1≤i≤n)个元素值

B.交换第1个元素与第2个元素的值

C.顺序输出这n个元素的值

D.输出与给定值x相等的元素在线性表中的序号

457.设二维数组 A[6][10]，每个数组元素占用4个存储单元，若按行优先顺序存放数组元素，a[0][0]的存储地址为860，则a[3][5]的存储地址是（ ）。

A.1000

B.860

C.1140

D.1200

458.在一棵度为3的树中，度为3的节点个数为2，度为2的节点个数为1，则度为0的节点个数为（ ）。

A.4

B.5

C.6

D.7

459.下列关于无向连通图特征的叙述中，正确的是（ ）。

(1). 所有顶点的度之和为偶数

(2). 边数大于顶点个数减1

(3). 至少有一个顶点的度为1

A. 只有（1）

B. 只有（2）

C.（1）和（2）

D.（1）和（3）

458.已知一棵完全二叉树的第6层（设根为第1层）有8个叶子节点，则该完全二叉树的节点个数最多是（ ）。

A.39

B.52

C.111

D.119

461.一棵度为4的树T中，若有20个度为4的节点，10个度为3的节点，1个度为2的节点，10个度为1的节点，则树T的叶子节点个数为（ ）。

A.41

B.82

C.113

D.122

462.若无向图G(V, E)中含7个顶点，则保证图G在任何情况下都是连通的，则需要的边数最少是（ ）。

A.6

B.15

C.16

D.21

463.已知一个长度为16的顺序表，其元素按关键字有序排列，若采用折半查找法查找一个不存在的元素，则比较的次数最多是（ ）。

A.4

B.5

C.6

D.7

464.采用递归方式对顺序表进行快速排序，下列关于递归次数的叙述中，正确的是（ ）。

A.递归次数与初始数据的排列次序无关

B.每次划分后，先处理较长的分区可以减少递归次数

C.每次划分后，先处理较短的分区可以减少递归次数

D.递归次数与每次划分后得到的分区处理顺序无关

465.对一组数据（2, 12, 16, 88, 5, 10）进行排序，若前三趟的结果如下：

第一趟：2, 12, 16, 5, 10, 88

第二趟：2, 12, 5, 10, 16, 88

第三趟：2, 5, 10, 12, 16, 88

则采用的排序方法是（ ）。

A.冒泡排序

B.希尔排序

C.归并排序

D.基数排序

466.设n是描述问题规模的非负整数，下面程序片段的时间复杂度为（ ）。

x=2;

while (x<n/2)

x=2\*x;

A.O(log2n)

B.O(n)

C.O(nlog2n)

D.O(n2)

467.一棵完全二叉树上有768个节点，则该二叉树中叶节点的个数是（ ）。

A.257

B.258

C.384

D.385

468.为实现快速排序算法，待排序序列宜采用的存储方式是（ ）。

A.顺序存储

B.散列存储

C.链式存储

D.索引存储

469.所谓简单路径是指（ ）。

A.任何一条边在这条路径上不重复出现

B.任何一个顶点在这条路径上不重复出现

C.这条路径由一个顶点序列构成，不包含边

D.这条路径由边序列构成，不包含顶点

470.无向图的邻接矩阵是一个（ ）。

A.对称矩阵

B.零矩阵

C.上三角矩阵

D.对角矩阵

471.在一个无向图中，所有顶点的度之和等于边数的（ ）倍。

A.1/2

B.1

C.2

D.4

472.具有5个顶点的无向图至少应有（ ）条边才可能是一个连通图。

A.4

B.3

C.5

D.7

473.下列（ ）的邻接矩阵是对称矩阵。

A. 有向图

B. 无向图

C. AOV网

D. AOE网

474.如果从无向图的任一顶点出发进行一次深度优先搜索即可访问所有顶点，则该图一定是（ ）。

A.完全图

B.连通图

C.有回路

D.一棵树

475.任何一个带权无向连通图（ ）最小生成树。

A.只有一棵

B.有一棵或多棵

C.一定有多棵

D.可能不存在

476.若图的邻接矩阵中主对角线上的元素全是0，其余元素全是1，则可以断定该图一定是（ ）。

A.无向图

B.不是带权图

C.有向图

D.完全图

477.一个无向连通图的生成树是含有该连通图的全部顶点的（ ）。

A.极小连通子图

B.极小子图

C.极大连通子图

D.极大子图

478.判定一个有向图是否存在回路除了可以利用拓扑排序方法外，还可用（ ）。

A.求关键路径的方法

B.求最短路径的Dijkstra方法

C.广度优先遍历算法

D.深度优先遍历算法

479.关键路径是AOE网中（ ）。

A.从源点到汇点的最长路径

B.从源点到汇点的最短路径

C.最长的回路

D.最短的回路

480.求最短路径的Dijstra算法的时间复杂度为（ ）。

A.O(n)

B.O(n+e)

C.O(n2)

D.O(ne)

481.对于含有n个顶点的带权连通图，它的最小生成树是指图中任意一个（ ）。

A.由n-1条权值最小的边构成的子图

B.由n-1条权值之和最小的边构成的子图

C.由n-1条权值之和最小的边构成的连通子图

D.由n个顶点构成的边的权值之和的最小连通子图

482.若无向图G的顶点度数最小值大于等于（ ）时，G至少有一条回路。

A.1

B.2

C.3

D.4

483.对于一个具有n个顶点和e条边的无向图，若采用邻接表表示，则表头数组的大小为（ ）。

A. n

B. n+1

C. n+2

D. 2n

484.对于一个具有n个顶点和e条边的无向图，若采用邻接表表示，则边节点总数是（ ）。

A. e

B. 2e

C. 3e

D. 4e

485.对n个顶点的连通图来说，它的生成树一定有（ ）条边。

A. n-1

B. n

C. n+1

D. n+2

486.已知一个有向图的邻接矩阵表示，删除所有从第i个顶点出发的边的方法是（ ）。

A.将邻接矩阵的第i行全部置为0

B.将邻接矩阵的第i列全部置为0

C.将邻接矩阵的第i-1行全部置为0

D.将邻接矩阵的第i-1列全部置为0

487.设n个元素进栈序列是p1，p2，p3，…，pn，其输出序列是1，2，3，…，n，若p3=3，则p1的值（ ）。

A.可能是２

B.一定是２

C.不可能是１

D.一定是１

488.一棵完全二叉树上有1001个节点，其中叶子节点的个数是（ ）个。

A.499

B.500

C.501

D.502

489.对于链串（长度为n，每个节点存储一个字符），查找元素值为ch的算法的时间复杂度为（ ）。

A.O(1)

B.O(n)

C.O(n2)

D.以上都不对

490.以下各种存储结构中，最适合用做链队的链表是（ ）。

A.带队首指针和队尾指针的循环单链表

B.带队首指针和队尾指针的非循环单链表

C.只带队首指针的非循环单链表

D.只带队尾指针的循环单链表

491.裴波那契数列的定义：f(n)=f(n-1)+f(n-2),f(0)=1,f(1)=2，其数据的定义形式是按（ ）定义的。

A.递推

B.迭代

C.非递归

D.递归

492.反复应用（ ）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。

A.分治

B.迭代

C.递推

D.贪心选择

493.最优子结构性质特征反映了（ ）思想的应用。

A.递推

B.迭代

C.非递归

D.递归

494.概率算法中蒙特卡罗算法得到的解（ ）。

A.一定是正确的

B.一定是最优的

C.不一定是正确的

D.一定是次优的

495.（ ）本身并不使用递归的定义。

A.递归函数

B.递归表达式

C.递归边界

D.递归思想

496.对于一个具有n个顶点的无向图，若采用邻接矩阵表示，则该矩阵的大小是（ ）。

A. n

B. (n-1)2

C. n-1

D. n2

497.若含有n个顶点的无向图恰好形成一个环，则它有（ ）棵生成树。

A. n

B. n-1

C. n-2

D. n-3

498.多阶段决策问题中，每一个阶段可能有（ ）个决策可供选择。

A.1

B.2

C.3

D.若干

499.可以进行拓扑排序的有向图一定是（ ）。

A.无环图

B.有环图

C.完全图

D.带权图

500.若无向图中有m条边，则表示该无向图的邻接表中有（ ）个边节点。

A. m

B. 2m

C. m+1

D. m-1