随机生成试卷，判断题30个30分，单选题30个30分，多选题20个20分， 填空题4个20分。

 倒计时：  01:29:05

1

单选(1分)

算法复杂度分析的两种方法是（）

* **A.**

最坏情况的复杂度和最好情况的复杂度

* **B.**

最坏情况的复杂度和平均复杂度

* **C.**

事前分析和事后统计

* **D.**

最好情况的复杂度和平均复杂度

2

单选(1分)

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.**

逼近

* **D.**

等价于

3

单选(1分)

从长度为n的数组中多次查找数据，使用(  )方法好？

* **A.**
* **排序后折半查找**
* **B.**

顺序查找

* **C.**

随机查找

* **D.**

无序查找

4

单选(1分)

f(n)= 6×2^n+n^2. f(n)的渐进性态f(n)= O( )

* **A.**

n^2

* **B.**

2^n

* **C.**

nlogn

* **D.**

n

5

单选(1分)

logn^2=(  )(n^1/2)

* **A.**

O

* **B.**

o

* **C.**

θ

* **D.**



6

单选(1分)

( logn!)=(  ) (n^3/2)

* **A.**

o

* **B.**

O

* **C.**

 θ

* **D.**



7

单选(1分)

给定n个元素的数组A，n=10^6, 使用折半查找比使用顺序查找快（  ）倍。

* **A.**

5000

* **B.**

500

* **C.**

50000

* **D.**

10000

8

单选(1分)

折半查找长度为n的线性表，平均查找长度为（）

* **A.**

(n+1)/2

* **B.**

log n

* **C.**

nlogn

* **D.**

n

9

单选(1分)

最大独立集问题，如果在10亿次每秒的计算机上运行，100年可以计算的图的规模估计是？

* **A.**

60

* **B.**

50

* **C.**

40

* **D.**

80

10

单选(1分)

最大独立集问题，如果在10亿次每秒的计算机上运行，当n=50时，需要计算的时间估计是？

* **A.**

100年

* **B.**

 1年

* **C.**

1小时

* **D.**

24小时

11

单选(1分)

下面描述错误的是（）

* **A.**

时间复杂度是输入规模n的函数。

* **B.**

求解问题的输入量,称为问题的规模。

* **C.**

空间复杂度是算法执行所需所有空间的资源量。

* **D.**

时间复杂度衡量算法的效率。

12

单选(1分)

下面不是以空间换时间的方法有（）

* **A.**

数据压缩

* **B.**

预处理

* **C.**

预构造

* **D.**

动态规划

13

单选(1分)

A公司处理器速度是B公司的1000倍。对于复杂度为n^3的算法，B公司的计算机可以在1小时内处理规模为n的问题，A公司的计算机在1小时能处理的问题规模是\_\_\_

* **A.**

n1000^1/2

* **B.**

n

* **C.**

100n

* **D.**

10n

14

单选(1分)

下面关于贪心算法错误的是（）

* **A.**

贪心选择中每一步的局部最优解都构成全局最优解的一部分

* **B.**

贪心算法一般预处理后再进行最优化选择。

* **C.**

问题的最优子结构性质是该问题可用贪心算法或动态规划算法求解的关键特征。

* **D.**

贪心算法总能找到可行解，并且是最优解。

15

单选(1分)

未来与过去无关，指的是无（）性质。

* **A.**

 无后效性

* **B.**

最优子结构

* **C.**

重叠子问题

* **D.**

贪心选择

16

单选(1分)

下面关于贪心算法的说法错误的是（）

* **A.**

贪心算法的思想是依据贪婪准则作出决策，逐步构造解值。

* **B.**

贪心算法总能找到可行解，但未必是最优解。

* **C.**

贪心算法的思想是寻求局部最优解，逐步达到全局最优

* **D.**

未来不影响过去指的是无后效性的性质。

17

单选(1分)

下面有关递归与递推的说法错误的是（）

* **A.**

递归表现为自己调用自己，递推则没有这样的形式。

* **B.**

递归是逆向的，从大规模的问题逐步到小规模间题。

* **C.**

递推是正向的，从小规模的问题推解出大规模间题。

* **D.**

一般来说，递归的效率高于递推

18

单选(1分)

设有5000个无序的元素，希望用最快的速度挑选出其中前500个最大的元素，最好选用(   )法。

* **A.**

冒泡排序

* **B.**

堆排序

* **C.**

快速排序

* **D.**

基数排序

19

单选(1分)

中印战争西山口战役刘伯承提出的“打头、截尾、剖腹、击背” 是（）思想。

* **A.**

递推

* **B.**

分治

* **C.**

枚举

* **D.**

贪心

20

单选(1分)

减少子问题合并的时间，就是减少时间复杂度函数T(n)=aT(n/b)+f(n)中的（）值。

* **A.**

n

* **B.**

a

* **C.**

b

* **D.**

f（n）

21

单选(1分)

（ ）是贪心算法与动态规划算法的共同点。

* **A.**

构造最优解

* **B.**

贪心选择性质

* **C.**

最优子结构性质

* **D.**

重叠子问题

22

单选(1分)

动态规划方程M[i，j]= min(M[i,k] + M[k,j] +wij), 1≤i≤k≤j≤n, 则算法的则算法的时间复杂度为（）。

* **A.**

n^3

* **B.**

(n^2)logn

* **C.**

n^4

* **D.**

n^2

23

单选(1分)

下面有关图上动归的说法错误的是（）

* **A.**

Dijkstra可以计算负边权的最短路问题。

* **B.**

如果图中存在负环，那么从s到t没有最短路。

* **C.**

SPFA算法计算时，如果一个顶点入队列的次数超过n，则存在负权回路。

* **D.**

动态规划计算树上的最大独立集时，从叶子开始，先计算子树，逐步计算到根节点。

24

单选(1分)

下面哪种函数不是回溯法中为避免无效搜索采取的策略（）

* **A.**

限界函数

* **B.**

约束函数

* **C.**

递归函数

* **D.**

剪枝函数

25

单选(1分)

分支限界法解0-1背包问题时的解空间树是（       ）

* **A.**

广度优先生成树

* **B.**

深度优先生成树

* **C.**

子集树

* **D.**

排列树

26

单选(1分)

下面说法错误的是（）

* **A.**

给定连通图G, BFS遍历得到层次图，如果同一层中的结点无边相连，则G是二分图。

* **B.**

设 f 任意流, (A, B) 是任意s-t 割, 则流值不小于割的容量。

* **C.**

设G是n阶无孤立点的图，则V\*是G的顶点覆盖，当且仅当V-V\*是G的独立集。

* **D.**

给定G = <V, E>, G的匹配中任何两条边都没有公共顶点。

27

单选(1分)

下面说法错误的是（）

* **A.**

网络中存在割 (A, B) 使流值 v(f) = 割的容量cap(A, B)，则割 (A, B)是最小割。

* **B.**

给定二分图G = <V, E>中无孤立点，其最大流算法求得最大流f, 则 G的最小顶点覆盖数=n-f

* **C.**

有下界的流通问题不一定有可行流。

* **D.**

匈牙利算法中起点和终点都是未匹配点的交错路径称为可增广路径，有奇数条边。

28

单选(1分)

下面说法错误的是（）

* **A.**

舍伍德算法总是有解, 且解总是正确的，改进了算法的平均性能。

* **B.**

随机算法共同点是计算时间越多或运行次数越多,正确性越高

* **C.**

确定性算法求解同一实例用同一算法求解两次，所得结果完全相同。

* **D.**

借助随机预处理技术，不改变原有的确定性算法，仅对其输入进行随机洗牌，可收到舍伍德算法的效果。

29

单选(1分)

下面有关动态规划算法错误的是

* **A.**

区间动归使用链长，先计算小区间，再递增计算大区间。

* **B.**

 0-1背包问题的动态规划算法可以使用一维数组实现。

* **C.**

矩阵连乘的计算次序可以用完全加括号的方式来确定。

* **D.**

 0-1背包问题的动态规划算法是多项式时间算法。

30

单选(1分)

下面有关动态规划算法错误的是（）

* **A.**

状态转移方程表示状态间的递推关系，也是子问题间的递推关系。

* **B.**

动态规划算法的每一个子问题只解一次,存储子问题结果，避免重复计算。

* **C.**

动态规划算法本质上是时间换空间的算法

* **D.**

贪心和递推算法是线性解决问题，动态规划则是全面分阶段地解决问题。

31

多选(1分)

算法的性质有（）

* **A.**

有穷性

* **B.**

确定性

* **C.**

输出

* **D.**

输入

32

多选(1分)

下面公式正确的是（）

* **A.**

  , 常数a, b > 0.

* **B.**

对于任意 x > 0,  log n = o(n^x).

* **C.**

对任意 r > 1 和  d > 0,  n^d= o(r^n).

* **D.**

33

多选(1分)

枚举算法的优化方法有（）

* **A.**

减少枚举变量的值域

* **B.**

优化数据结构

* **C.**

优化数学模型

* **D.**

减少枚举变量

34

多选(1分)

子集生成方法有（）

* **A.**

法向量法

* **B.**

位向量法

* **C.**

二进制法

* **D.**

增量构造法

35

多选(1分)

通过降低子问题合并时间，降低分治算法时间复杂度的有（）

* **A.**

计数逆序

* **B.**

最接近点对

* **C.**

大整数乘法

* **D.**

线性时间选择

36

多选(1分)

时间复杂度为O(nlogn)的排序算法有（）

* **A.**

快速排序

* **B.**

堆排序

* **C.**

计数排序

* **D.**

合并排序

37

多选(1分)

时间复杂度为O（n^2）的排序算法有（）

* **A.**

快速排序

* **B.**

直接选择排序

* **C.**

冒泡排序

* **D.**

 插入排序

38

多选(1分)

最好情况下，时间复杂度为O（n）的排序算法有（）

* **A.**

直接选择排序

* **B.**

插入排序

* **C.**

 冒泡排序

* **D.**

计数排序

39

多选(1分)

分治算法的适用条件有（）。

* **A.**

**子问题可合并为问题的解**

* **B.**

**小规模子问题可解**

* **C.**

**子问题相互独立**

* **D.**

**问题可以分解为规模较小的子问题**

**40**

**多选(1分)**

**下面描述分治算法正确的是（）**

* **A.**

**最小堆中每个元素调整的次数不超过树高 (logn)。**

* **B.**

**三分法的判定树是三叉树**

* **C.**

**二分法子问题不独立的情况可以使用分治算法计算，但计算量大。**

* **D.**

**处理随机排列的数组时，合并排序比快速排序快。**

**41**

**多选(1分)**

**区间动态规划的计算次序是（）**

* **A.**

**先小区间后大区间**

* **B.**

**自底向上**

* **C.**

**先大区间后小区间**

* **D.**

**自顶向下**

**42**

**多选(1分)**

**最短路算法中适用于负权图的是（）**

* **A.**

**SPFA算法**

* **B.**

**Floyd算法**

* **C.**

**Dijkstra算法**

* **D.**

**Bellman算法**

**43**

**多选(1分)**

**OPT(i，w): 从1-i种物品中选择，放入容量为w的背包时的最大价值。这是（）问题动态规划算法的递推函数。**

* **A.**

**恰好装满的0/1背包**

* **B.**

**0/1背包**

* **C.**

**完全0/1背包**

* **D.**

**多重0/1背包**

**44**

**多选(1分)**

**回溯算法的效率在很大程度上依赖的因素有（）：**

* **A.**

**满足显约束的x[k]值的个数**

* **B.**

**产生x[k]的时间**

* **C.**

**剪枝时间：计算可行性约束函数constraint 和 上界函数bound的时间。**

* **D.**

**满足可行性约束函数和上界函数的所有x[k]的个数**

**45**

**多选(1分)**

**下面说法正确的是（）**

* **A.**

**剩余网络中从源s到汇t的最小费用路是剩余网络中从s到t的以费用为权的最短路**

* **B.**

**多源点和多汇点的网络流问题可以通过增加一个“超源点”和“超汇点”转化为单源点和单汇点的网络流问题。**

* **C.**

**无向图的每条边变为方向相反的两条边，容量是原边的容量，这样无向图的最大流问题变换为有向图的最大流问题。**

* **D.**

**最小费用最大流算法寻找从源点s到汇点t的最小费用路，然后沿最小费用路增流，直至找到最小费用流。**

**46**

**多选(1分)**

**给定二分图G = <V, E>中无孤立点，|V|=n，其最大流算法求得最大流f, 则 G的（）=f.**

* **A.**

**最大独立数**

* **B.**

**最小边覆盖**

* **C.**

**最小顶点覆盖**

* **D.**

**最大匹配数**

**47**

**多选(1分)**

**从活结点表中选择下一个扩展结点的不同方式将导致不同的分支限界法，最常见的方式有（    ）。**

* **A.**

**优先队列式分支限界法**

* **B.**

**队列式分支限界法**

* **C.**

**FIFO分支限界法**

* **D.**

**栈式分支限界法**

**48**

**多选(1分)**

**属于最短路增广路算法的有**

* **A.**

**ISAP算法**

* **B.**

**Dinic算法**

* **C.**

**FF算法**

* **D.**

**EK算法**

**49**

**多选(1分)**

**给定网络N=(V, E)的一个流 f，f需满足的条件是**

* **A.**

**汇点t的流入量=|f|**

* **B.**

**对于每个顶点 v  V–{s, t}: 净流量=0**

* **C.**

**对于每条边 e  E:    0≤f(e) ≤ c(e),  c(e)为边e的容量**

* **D.**

**源点s的流出量=|f|**

**50**

**多选(1分)**

**（ ）肯定获得最优解。**

* **A.**

**回溯算法**

* **B.**

**贪心算法**

* **C.**

**枚举算法**

* **D.**

**随机算法**

**51**

**判断(1分)**

**同一算法可以使用多种形式描述。**

* **A.**
* **B.**

**52**

**判断(1分)**

**n^(1/logn)=O(1)**

* **A.**
* **B.**

**53**

**判断(1分)**

**有n个结点的判定树的深度为logn+1**

* **A.**
* **B.**

**54**

**判断(1分)**

**f=ω(g)且g =ω(h)则f  =ω(h)**

* **A.**
* **B.**

**55**

**判断(1分)**

两个n\*n的矩阵相乘的时间复杂度是(n^2)

* **A.**
* **B.**

**56**

**判断(1分)**

**f=Ω(g) 且g = Ω(h)则f  =Ω (h)**

* **A.**
* **B.**

**57**

**判断(1分)**

对近似递增序列的线性表从小到大排序，使用合并排序比较好。

* **A.**
* **B.**

**58**

**判断(1分)**

**增量构造法生成子集，便于实现集合的操作。**

* **A.**
* **B.**

**59**

**判断(1分)**

**子集生成算法中一般需要对集合元素进行定序。**

* **A.**
* **B.**

**60**

**判断(1分)**

**如果e是图G中权重最小的边，它肯定是G的一颗最小生成树的边。**

* **A.**
* **B.**

**61**

**判断(1分)**

**最小生成树是唯一的。**

* **A.**
* **B.**

**62**

**判断(1分)**

**贪心选择的性质是贪心算法与动态规划算法的主要区别**

* **A.**
* **B.**

**63**

**判断(1分)**

每个迭代算法原则上总可以转换成与它等价的递归算法；反之不然

* **A.**
* **B.**

**64**

**判断(1分)**

递归是从问题的最终目标出发，逐渐将复杂问题化为简单问题，最终求得问题。

* **A.**
* **B.**

**65**

**判断(1分)**

不基于元素比较的排序算法可以在线性时间实现。

* **A.**
* **B.**

**66**

**判断(1分)**

Floyd算法是动态规划算法，稠密图效果最佳，边权可正可负。

* **A.**
* **B.**

**67**

**判断(1分)**

SPFA算法通过维护一个队列，使得一个节点的当前最短路径被更新之后没有必要立刻去更新其他的节点，大大减少了重复操作的次数。

* **A.**
* **B.**

**68**

**判断(1分)**

**SPFA算法是Bellman-Ford算法的一种队列实现，适用于稠密图。**

* **A.**
* **B.**

**69**

**判断(1分)**

**DAG动态规划算法中正推的开始点是无入边的顶点**

* **A.**
* **B.**

**70**

**判断(1分)**

对于稠密图，Floyd算法的效率要高于执行n次Dijkstra算法，也要高于执行n次SPFA算法

* **A.**
* **B.**

**71**

**判断(1分)**

回溯法搜索解空间时，在其它条件相当的前提下，让可取值最少的x[i]优先，可以减少计算。

* **A.**
* **B.**

**72**

**判断(1分)**

回溯法一般在满足约束条件的解中找出在某种意义下的最优解。

* **A.**
* **B.**

**73**

**判断(1分)**

分支限界法以深度优先的方式搜索解空间树。

* **A.**
* **B.**

**74**

**判断(1分)**

使用限界函数作优先级, 第一个加入队列的叶子就是最优解

* **A.**
* **B.**

**75**

**判断(1分)**

设G是n阶无孤立点的图，则V\*是G的最小顶点覆盖，当且仅当V-V\*是G的最大独立集。

* **A.**
* **B.**

**76**

**判断(1分)**

设G = <V, E>中无孤立点，|V|=n，则边覆盖数 + 匹配数 = n

* **A.**
* **B.**

**77**

**判断(1分)**

拉斯维加斯算法肯定得到正确解或找不到解, 一旦找到一个解,一定是正确解。

* **A.**
* **B.**

**78**

**判断(1分)**

现实计算机上无法产生真正的随机数

* **A.**
* **B.**

**79**

**判断(1分)**

当最坏和平均情况差别较大时, 舍伍德算法可以消除好坏实例的差别，达到平均实例的性能.

* **A.**
* **B.**

**80**

**判断(1分)**

增加蒙特卡罗算法的求解次数, 可使求解错误的概率任意小。

* **A.**
* **B.**

**81**

**填空(5分)**

**分析下列程序的上界O和下界。**

**p = 0.0**

**for i = n down to 0 do**

**power = 1**

**for  j= 1 to i do**

**power =  power \*x**

**p = p + a[i] \* power**

**return p**

**该程序时间复杂度的上界是O(\_\_\_\_)、下界是(\_\_\_\_\_)。**

**请输入答案**

**82**

**填空(5分)**

**分析下列程序的上界O和下界。**

**p=1**

**for i= n to 1  do**

**p=p+n**

**i=i/2**

**该程序时间复杂度的上界是O（\_\_\_\_）、下界是（\_\_\_\_\_）。**

**请输入答案**

**83**

**填空(5分)**

**T(*n*) = 3T(*n*/2) + n， T(1)=1，则 T(*n*) =（\_\_\_）**

**请输入答案**

**84**

**填空(5分)**

T(***n***) =9T(***n***/3) + n， T(1)=1，则 T(***n***) =（\_\_\_）