# 算法设计与分析期末历年真题归纳总结

**1. 贪心算法和动态规划算法有什么共同点和区别？它们都有那些优势和劣势？**

共通点：动态规划和贪心算法都是一种递推算法 ，均有局部最优解来推导全局最优解

区别：贪心算法中，作出的每步贪心决策都无法改变，每一步的最优解一定包含上一步的最优解，而上一部之前的最优解则不作保留。

动态优化算法，全局最优解中一定包含某个局部最优解，但不一定包含前一个局部最优解，因此需要记录之前的所有最优解

动态规划算法利用子问题重叠性质，对每一个子问题只计算一次，将其解保存在一个表格中。不同的子问题个数随着输入问题的规模呈多项式增长，因此，动态规划算法通常只需要多项式时间，从而获得较高的解题效率。但它需要计算之前所有情况花费，更加耗费空间。

**贪心算法**所作的选择依赖于以往所作过的选择,但决不依赖于将来的选择,这使得算法在编 码和执行过程中都有一定的速度优势。**贪心算法**是只是找局部最优解，不一定是全局最优解。

**2. 试比较回溯法与分枝限界算法，分别谈谈这两个算法比较适合的问题？**

二者都是在解空间树里搜索问题的可靠解或最优解，但是搜索的方式不同，回溯法采用深度优先的方式，直到达到问题的一个可行解，或经判断沿此路径不会达到问题的可行解或最优解时，停止向前搜索，并沿原路返回到该路径上最后一个还可扩展的节点，然后，从该节点出发朝新的方向纵深搜索。分枝限界法采用的是宽度优先的方式，它将活节点存放在一个特殊 的表中，其策略是，在扩展节点处，首先生成其所有的儿子节点，将那些导致不可行解或导致非最优解的儿子节点舍弃，其余儿子节点加入活节点表中，然后，从活节点中取出一个节点作为当前扩展节点，重复上述节点中扩展过程。可以看出，回溯法一般用于求问题的一个可行解，而分枝限界可以用于求出问题的所有可行解。

**3. 何谓最优化原理？采用动态规划算法必须满足的条件是什么？动态规划算法是通过什么问题的什么特性提高效率的？**

一个最优化策略的子策略总是最优的。一个问题满足最优化原理又称其具有最优子结构性质。最优子结构性质，子问题重叠性质是计算模型采用动态规划算法求解的两个基本要素。

动态规划算法利用子问题重叠性质，对每一个子问题只计算一次，将其解保存在一个表格中。不同的子问题个数随着输入问题的规模呈多项式增长，因此，动态规划算法通常只需要多项 式时间，从而获得较高的解题效率

**4. 什么是多项式时间算法？**

若存在一个常数C，使得对于所有n>=0，都有|f(n)| <= C*|g(n)|，则称函数f(n)是O(g(n))。

时间复杂度是 O(p(n))的算法称为多项式时间算法，这里 p(n)是关于 n 的多项式。

时间复杂度为O(nlog(n))、O(n^3)的算法都是多项式时间算法，时间复杂度为O(n^log(n))、O(n!)、O(2^n)的算法是指数时间算法。

一个优化问题如果已经找到了多项式时间算法，则称该问题为多项式时间可解问题，并将这类问题的集合记为 P，因此多项式时间可解问题就称为 P 类问题。

**5. 多项式时间确定性算法与多项式时间非确定性算法的主要区别是什么？**

在算法计算复杂性的研究中，一个算法如果存在图灵机可计算的多项式时间计算复杂性算法，就将这个算法归入 P 类，如果存在非确定性图灵机可计算的多项式时间计算复杂性算法，就将其归入 NP 类

**6. 陈述算法在最坏情况下的时间复杂度和平均时间复杂度；这两种评估算法复杂性的方法各自有什么实际意义？**

最坏时间复杂度式算法在最差情况下的时间复杂度，也就是花费时间最多的情况。平均时间复杂度是因为它是期望的运行时间。它更有意义，现实中，平均运行时间很难通过分析得到，一般都是通过运行一定数量的实验数据后估算而来的。而最坏运行时间是一种保证，那就是运行时间不会再坏了。在应用中，这是最重要的需求，通常我们提到的运行时间都是最坏情况下的运行时间，时间复杂度是最坏情况下的时间复杂度。

**7. 在对算法进行复杂性分析时，强调渐进复杂性的意义是什么？**

当问题的规模 n 趋向无穷大时，影响算法效率的重要因素是 T(n)的数量级，而其他因素仅是使时间复杂度相差常数倍。使用渐进表达式可以略去低阶项所留下的主项，更加简单。

P11

**8. 在对算法进行复杂性分析时，时间复杂度用什么量反映的？其间做了什么假定？复杂性函数的渐进上界反映了复杂性函数的什么性质？**

通常我们提到的运行时间都是最坏情况下的运行时间，时间复杂度是最坏情况下的时间复杂度

我们假定 N 充分大，渐近上界也反映了复杂性函数在 N 充分大的情况下复杂性的上界

**9. 什么是 NPC 问题？证明一个问题是 NPC 问题一般采用哪几个步骤？**

第一，证明该问题是 NP 问题

第二，选 取一个已知的 NPC 问题

第三，构造一个从二中的问题到题目中问题的变换

第四，证明这个变换是多项式变换

**10. 已知求解问题的两个算法的时间复杂性函数分别为和。现在有两台计算机，它们的速度比为 64。如果采用算法，计算机求解问题的一个实例所用的时间为，那么，采用算法时，计算机能够在时间内求解问题的多大输入规模的实例？**

解决这类问题，只要列出式子即可

**11.在连通图无向图的宽度优先搜索树和深度优先搜索树中，哪棵树的最长路径可能会更长些？试说明你的理由。**

深度优先搜索树可能会长些，因为深度优先搜索是沿着顶点的邻点一直搜索下去的，直到当前被搜索的点不再有未被访问的邻点为止，且在此过程中记录的边构成了搜索树。深度搜索树的最长路径一定包含图中的最长路，而宽度搜索则不一定包括。因此，在每个顶点较少的图上两者等得到的树最长路径可能差不多，但是在每个顶点邻点较多的图上，深度搜索可以产生路径更长的搜索树。

12. 确定性图灵机模型与非确定性图灵机模型的主要区别在那里？确定性图灵机模型下算法的时间复杂度和空间复杂度指的是什么？

二者的区别就在于，确定性的每一步只有一种选择，而非确定性有多种选择，由些可见，非的计算能力比确定性强得多。时间复杂性即从开始直至进入停机状态所运行的步数，同理空间复杂度

**13.归并排序算法和快速排序算法各自强调了那个方面？各自提高效率的策略是什么？**

归并由分解与合并两部分组成。提高的话一个是当元素比较少时，可以直接进行排序，比如插入排序。这比分解合并要快得多。二是尽量采用链表结构，因链表结构的移动要快于数组

快排也是利用分治法排序。主要过程为划分。改进的第一条思路与归并排序同理，即在元素较少时，可以直接进行插入排序，第二条思路是在选择key值时，尽量选择与中值接近的值，其他的一些策略还有三路快速排序等。

**14. 在对算法进行复杂性分析时，时间复杂度用什么来度量？其间做了什么假定？渐进复杂性指的是什么？精确到什么程度？**

①程序中关键步骤的操作计数。

②假定每种基本操作所用时间都是一个单位。

③设T(n)是算法A的复杂性函数。如果存在函数 $\tilde{T}(n)$，使得当 $n \to \infty$ 时有 $(T(n)-\tilde{T}(n$

则称 $\tilde{T}(n)$ 是算法 A 当 $n \to \infty$ 的渐近复杂性说来，算法的渐进复杂性是该算法复杂度函数的主项，且一般只考虑渐进复杂性的阶。

④算法的渐进复杂性是该算法复杂度函数的主项，且一般只考虑渐进复杂性的阶。

⑤意义：简化算法复杂性分析的方法和步骤。以方便于对各类算法进行分析和比较。

**15. 什么是并行算法？并行计算有那几种主要模型？衡量并行算法优劣主要用那几个指标？**

并行算法就是用多台处理机联合求解问题的方法和步骤，其执行过程是将给定的问题首先分解成若干个尽量相互独立的子问题，然后使用多台计算机同时求解它，从而最终求得原问题的解。模型：PRAM模型，BSP模型，LogP模型，C^3模型。指标：执行时间、工作负载、存储性能。

## 相对近似与绝对近似

- 问题Π，实例I，最优值$F^*(I)$，算法A产生的可行解的值$F(I)$
- **绝对近似算法**：算法A称为问题Π的绝对近似算法，如果对问题Π的每个实例I，有 $|F^*(I)-F(I)|\le k$，其中，k是常数。
- **相对近似算法**：算法A称为问题Π的相对近似算法，如果对问题Π的每个实例I，有$|(F^*(I)-F(I))/F^*(I)|\le \varepsilon(n)$，其中，$\varepsilon(n)$是I的规模n的函数(一般是多项式)。常称为$\varepsilon(n)$-近似(算法)。

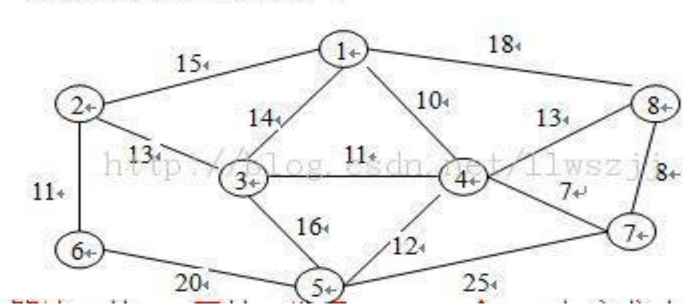
# 不一致性与约束传播

**约束传播**：约束满足问题有两类主要的求解方式：系统的搜索技术和局部搜索技术。前者是完备搜索技术，无论是否有解，当搜索完成时都可以知道，也称为回溯搜索。后者是不完备搜索技术，如果有解，可能给出结果，也可能不给出结果。一般来说，局部搜索比系统搜索能求解更大规模的问题实例，但是如果问题不可满足，局部搜索是没有意义的。在搜索过程中，**约束传递(Constraint propagation)**技术是用来检查一致性的一类重要技术，同时也是缩小变量的值域，从而减少之后的搜索空间的重要技术。 其基本思想是通过循环分析变量、值域和约束， 检验并删除不可能出现在可行解中的变量赋值， 从而约减变量值域.

**打破对称**：在 CSP （即约束满足问题）问题中对称性是是一个常见的性质，因为对称性的存在，导致搜索的时候搜索重复的空间，从而减低了搜索效率。特别是当问题无解时，完备算法需要搜索完整个搜索空间，打破对称技术就至关重要。对称有时也被称为同构（Isomorphism），打破对称（Symmetry Breaking）也被称为消除同构（Isomorphism Elimination）。

**三种方式**：打破对称可以从问题的三个阶段进行。第一，可以在问题建模的时候，尽量避免对称；第二，可以在问题求解开始的时候，通过添加约束来打破对称；第三，可以在求解过程中，动态地打破对称。

二．（20 分）试用 Prim 算法求解下面无向赋权图的最小生成树，指出最小生成树及该树中各边被选中的先后次序；写出算法的基本步骤。




解：根据 Prim 算法，从 V1 开始，选择 10　 V1 和 V4 加入集合

找出集合中顶点相邻的最小权值点 V7 加入集合

依次为 V 1 V4 V7 V8 V3V5 V2 V6

基本步骤：从第一个结点开始，加入集合 A

每次选择 A 中顶点与 A 外的顶点权值最小的顶点，加入集合 A

直到集合 A 包含所有顶点

三．（20 分）用 LC－分枝限界算法求解 0/1 背包问题：，物品重量和价值分别是：

w=(2,3,4,6,9) p=(8,9,10,12,18)

1.画出由算法生成的状态空间树，并标明各节点的优先级的值；

2.给出各节点被选作当前扩展节点的先后次序；

3.给出最优解。

解：30

具体步骤就不写了

四．（20 分）已知一组数满足，且被搜索的对象的概率分布是：

$$a_0=0.1,\ a_1=0.01,\ a_2=0.02,\ a_3=0.04,\ a_4=0.03,\ a_5=0.2$$
$$b_1=0.15,\ b_2=0.05,\ b_3=0.075,\ b_4=0.25,\ b_5=0.075$$

其中 a 表示被搜索对象在区间内的概率，b 表示被搜索对象为的概率，

使用动态规划算法求该搜索问题的最优二叉搜索树。

解：各子树的根：

1 1 1 4 4

0 2 3 4 4

0 0 3 4 4

0 0 0 4 4

0 0 0 0 5

最优二叉树结构：

k4 是 根k1

是 k4 的左孩子d0

是 k1 的左孩子k3

是 k1 的右孩子

k2 是 k3 的左孩子

d1 是 k2 的左孩子

d2 是 k2 的右孩子

d3 是 k3 的右孩子

k5 是 k4 的右孩子

d4 是 k5 的左孩子

d5 是 k5 的右孩子

| | | | | | |
|---|---|---|---|---|---|
| 0.1 | 0.37 | 0.54 | 0.89 | 1.645 | 2.425 |
| 0 | 0.01 | 0.11 | 0.345 | 0.85 | 1.63 |
| 0 | 0 | 0.02 | 0.195 | 0.64 | 1.42 |
| 0 | 0 | 0 | 0.04 | 0.39 | 1.17 |
| 0 | 0 | 0 | 0 | 0.03 | 0.535 |
| 0 | 0 | 0 | 0 | 0 | 0.2 |

五．(20 分) 假定已知“无向图的 Hamilton 回路”问题是 NPC 问题，证明“旅行商判定问题” 也是 NPC 问题。

解：首先，旅行商问题是 NP 的，因为对其解的任一猜想，要检验它是否是最优的，需要同所有其它的环游戏比较，这样的环游会有指数个，因而不可能 在多项式时间内完成考虑图的哈密顿回路问题，已知无向图 G |V|=n,构造其对应的旅行商问题为Dij={1,if（vi,vj）属于边，2，否则}显然，这一变换可以在多项式时间内完成，而且，G 有哈回路的充分必要条件是上述构建的旅行商问题有解，且解对应的路长度为 N，因为，若 G 中不含哈回路，则路长至少为 n+1 因为已知哈回路问题是 NPC 问题，并且上述变换为多项式变换，所以旅行商问题也为 NPC 问题。

六．设 $S=\{6,7,10,12,15,20\}$ 和 $M=35$，使用求定和子集问题的回溯算法找出 $S$ 的所有和数为 $M$ 的子集，并画出所生成的部分状态空间树。

七．用动态规划算法解下面0/1背包问题，写出各集合 $S^i, S_a^i$ 以及最优解的产生过程：
$n=5$, $(p_1,p_2,p_3,p_4,p_5)=(10,15,6,8,4)$, $(w_1,w_2,w_3,w_4,w_5)=(4,6,3,4,2)$, $M=12$

八．假定已知“无向图的Hamilton圈”问题是NPC问题，证明“旅行商判定问题”也是NPC问题。说明“旅行商最优问题”不是NPC问题。

九．设 $p_1,p_2,\cdots,p_n$ 是准备存放到长为 L 的磁带上的 n 个程序，程序 $p_i$ 需要的带长为 $a_i$。设 $\sum_{i=1}^{n} a_i > L$，要求选取一个能放在带上的程序的最大子集合（即其中含有最多个数的程序）$Q$。构造 $Q$ 的一种贪心策略是按 $a_i$ 的非降次序将程序计入集合。

1). 证明这一策略总能找到最大子集 $Q$，使得 $\sum_{p_i \in Q} a_i \le L$。

2). 设 $Q$ 是使用上述贪心算法得到的子集合，磁带的利用率 $\sum_{p_i \in Q} a_i / L$ 可以小到何种程度？

3）. 试说明1)中提到的设计策略不一定得到使 $\sum_{p_i \in Q} a_i / L$ 取最大值的子集合

十．虑无向图上的搜索算法

1．将宽度优先搜索算法中的队列改成栈，其它不变，给出一个搜索算法。

2．下面是一个无向图及其邻接链表，用你给出的算法画出图 $G$ 的搜索生成树，标出各节点被访问的次序号。

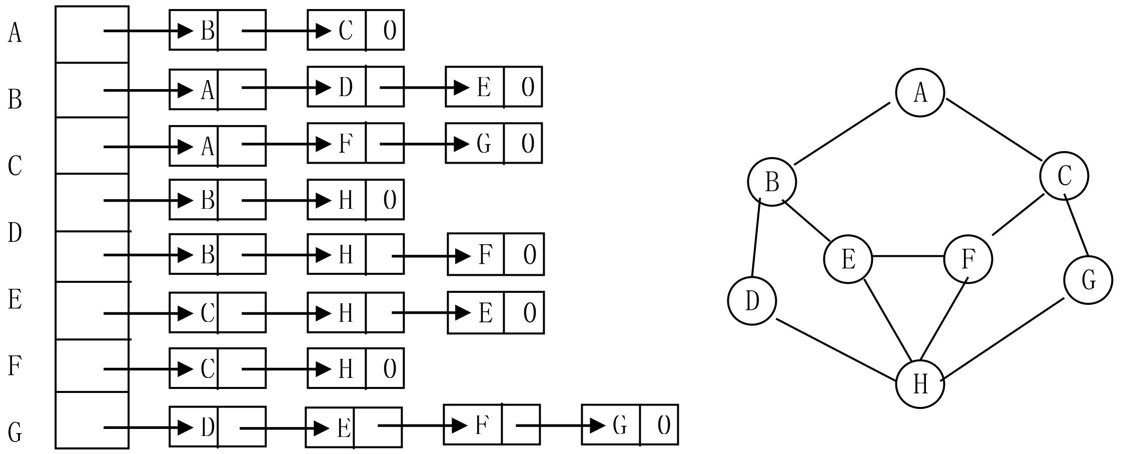



**图1 一个无向图 $G$ 和它的邻接链表**

十一． 已知求解问题 $\Pi$ 的两个算法 $A_1, A_2$ 的时间复杂性函数分别为 $T_1(n)=n2^{n/2}$ 和 $T_2(n)=n\log^2 n$。现在有两台计算机 $C_1, C_2$，它们的速度比为 64。如果采用算法 $A_1$，计算机 $C_1$ 求解问题 $\Pi$ 的一个实例 $I$ 所用的时间为 $T$，那么，采用算法 $A_2$ 时，计算机 $C_2$ 能够在时间 $T$ 内求解问题 $\Pi$ 的多大输入规模的实例？

十二． 有5个物体，其重量分别为3，5，7，8，9，价值分别为4，6，7，9，10。有一背包，载重量为22，物体不可分割地往背包里装。试画出用优先级队列式分枝限界算法LCKNAP解此0/1背包问题所生成的解空间树，并给出最优解。

十三． 假定已知“无向图的 Hamilton 圈”问题是 NP-完全问题.

1.证明旅行商问题判定模式也是 NP-完全问题；

2.证明旅行商问题优化模式不是NP-完全问题，但是NP难问题。

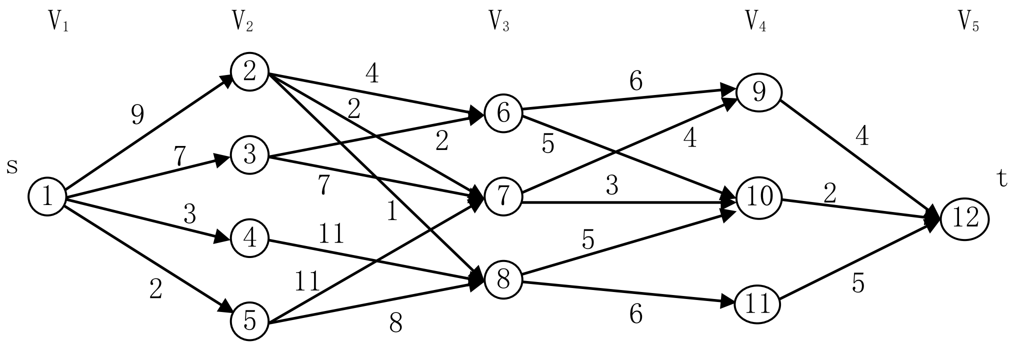
十四. （共 16 分）用优先队列式分枝限界算法解如下 0/1 背包问题：

$n=4,(p_1,p_2,p_3,p_4)=(10,15,6,4),(w_1,w_2,w_3,w_4)=(4,6,3,2),M=12$

1. 画出解空间搜索树，并标注说明；（12 分）

2. 给出最优解。（4分）

十五. 用动态规划算法解如下图示的多段图问题，




1. 说明多段图问题具有最优子结构性质；（5 分）

2. 写出多段图问题最优值的递推公式；（5 分）

3. 给出问题的一个最优解并在图上标注说明。（6分）

十六. 已知划分问题是 *NPC* 问题，试证明 0/1 背包（判定）问题也是 *NPC* 问题

十七. 装箱问题：将 $n$ 物品装入（不能分割）容积相等的若干个箱子。假定第 $i$ 件物品装入箱子所占的容积是 $v_i, 0<v_i\le 1(i=1,2,\cdots,n)$，箱子的容积都是 1。确定装箱方法，使所用的箱子个数尽量少。

1．试给出一个贪心算法，并说明算法的时间复杂度；

2．已知 $v_i=1/(i+1)(i=1,2,3,4,5,6,7,8)$，按照你给出的算法描述装箱过程。

3．你给出的贪心算法能够获得装箱问题的最优解吗？简单说明理由。

十八. 用优先队列式分子限界法解下面的旅行商问题，假定旅行商开始时处在第一个城市，各个城市（共 5 个城市）间的距离由下面的矩阵给出：

$$A=\begin{pmatrix} \infty & 7 & 3 & 12 & 8 \\ & \infty & 6 & 14 & 9 \\ & & \infty & 5 & 18 \\ & & & \infty & 11 \\ & & & & \infty \end{pmatrix}$$

1.说明你所使用的优先级函数和限界函数；

2.画出解空间树（即状态空间树），通过活结点表的变化说明搜索过；

3.给出一条旅行商最短的环行路线。

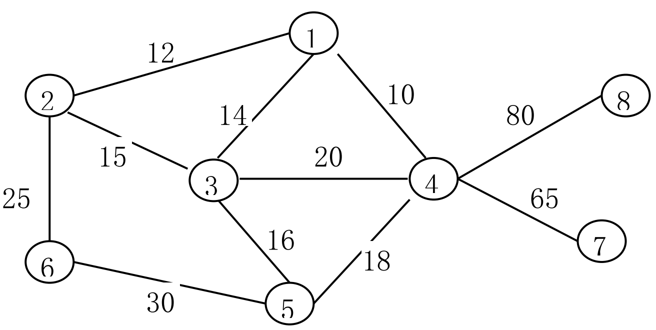
十九. 下面是二元可满足性问题，它是P类问题还是NPC问题？

**例**：给定逻辑语句$C=C_1\wedge C_2\wedge\cdots\wedge C_l$，其子句定义在布尔变量$X=\{x_1,x_2,\cdots,x_n\}$上，而且每个子句均由两个文字构成$C_i=y_i\vee z_i$， $y_i,z_i\in\{x_1,\overline{x}_1,\cdots,x_n,\overline{x}_n\}$， $i=1,2,\cdots,l$。

**问**：是否存在布尔变量的一个真值分配，使得语句$C$取真值？

二十. 分析Kruskal算法的时间复杂度；

二十一. 试用下面的例子说明用Kruskal算法求解最优生成树问题，并总结出算法的基本步骤：




二十二. 采用优先队列式分枝限界算法求解0/1背包问题：

$n=4,P=(15,15,18,27),W=(2,4,6,9),M=15$。

1.画出状态空间树，并在各个节点处标出目标值上界估值 Pvu 和下界估值 Pvl；

2.指出状态空间树中各节点被选作当前扩展节点的顺序（标号）。

二十三. 用动态规划的思想考虑0/1背包问题：证明：0/1背包问题具有最优子结构性质；

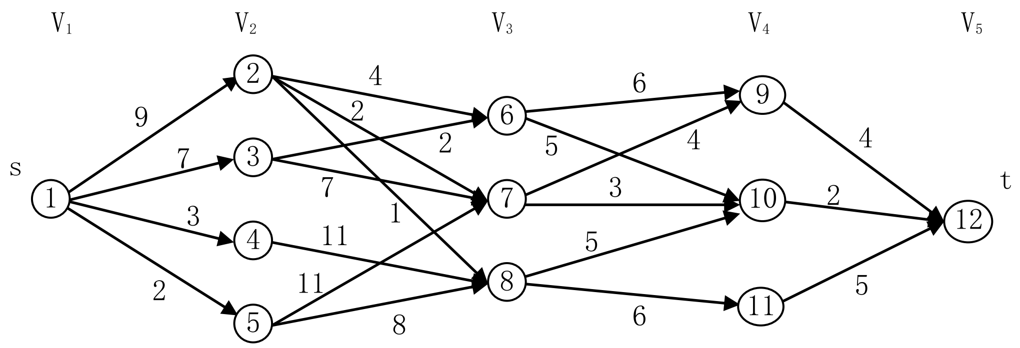
二十四. 考虑如下 0/1 背包问题：

$n = 4, (p_1, p_2, p_3, p_4) = (10,15,6,4), (w_1, w_2, w_3, w_4) = (4,6,3,2), M = 12$

试用动态规划方法解之（写出步骤和最优解）。

二十五. 用动态规划算法解多段图问题，即求从源点 s 到目标点 t 的最短路径：

1. 说明多段图问题具有最优子结构性质；
2. 写出多段图问题最优值的递推公式；
3. 给出下图问题的一个最优解并写出基本计算步骤。




**一个5阶段的网络图**

二十六. 用LC－分枝限界算法求解0/1背包问题： $n = 5$, $M = 12$ ，物品重量和价值分别是：

$$W = (2,3,4,6,9) \quad 和 \quad P = (8,9,10,12,18)$$

1. 画出由算法生成的状态空间树，并标明各节点的优先级的值；
2. 给出各节点被选作当前扩展节点的先后次序；
3. 给出最优解