1.

题目大意是给定一些字母的集合，每个集合里有若干字母，询问一个单词是否能用一些集合中的字母表示(其中每个集合只能用一次), 这一题可以看做一个二分图匹配。

要组成的单词的字母看成左边的结点，每个字母集合看成右边的一个结点，若是左边的字母在右边集合里，则这两个点之间连线，若是左边的每个点都能匹配到，那么这个单词可以被表示，反之则不行。

可以采取网络流算法建图或者使用匈牙利算法。

2.

①两次网络流

方法：先跑一次网络流，让所有满流的边为1，非满流的边为INF，再进行一次网络流，得到的最小割即为所求

分析：先跑一次网络流，若是某一条边在一组最小割里，这条边就一定是满流的(否则的话最大流就小于最小割了)。让所有满流的边为1，非满流的边为INF，再进行一次网络流，就可以去除其他边的影响，这时割去每条边的代价是一样的，那么一定是割去 原来图上最小边数的最小割 能以最小代价使图不连通。

②一次网络流

方法：所有边权 w = w * A + 1(A 为一个大数，A > E), 这时跑一次最大流，得到的最小割即为所求

分析：将所有的边权乘上一个大数再加1，原来割去边的值也乘上一个大数加1了，那原来的所有最小割里一定是边数最少的累加起来边权最少，而为了保证原来的最小割一定是现在的最小割，边权要乘大数。

26-1

a. 可以将每个顶点拆成两个顶点和一条边，若某个顶点 x 通过的最大值为 $l(x)$, 则拆分成 x_1 顶点和x_2 顶点，其中原来 x 的入边和 x_1 相连， x 的出边和 x_2 相连， 还要建立一条 x_1 指向 x_2 容量为 $l(x)$ 的边， 这样就可以划归成只有边限制的最大流问题。

b.可以先建立一个超级源点，向每个开始点通向一条容量为1的边；还要建立一个超级汇点，每个边界上的点向这个汇点连接一条容量为1的边，由于逃生路径不能相交，那么所有点的最大容量为1，和周围四个(边界点没有4个)点有一条容量为1的双向路径(其中每个点是 x 连向 x', 原来的入边和 x 相连， 出边和 x'相连)，这样求得最大流，如果等于开始点的个数，则原问题有解。

26-2

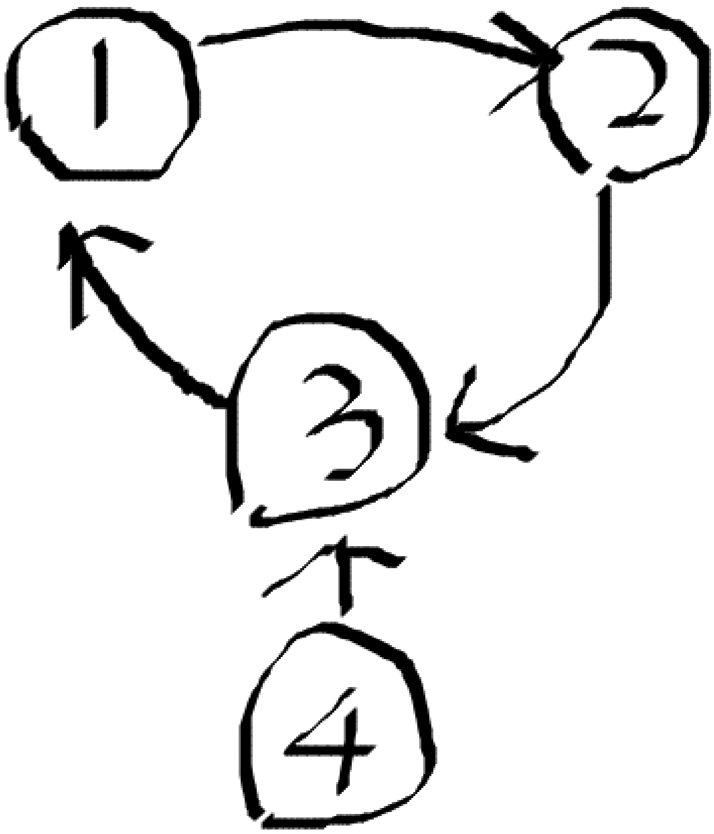
a. 构造方法：可以按照提示那样构造新的图，其中 x_0 表示源点，y_0 表示汇点，x_0 和 x_i 相连，y_i 和 y_0 相连, 若 $(i,j) \in E$, 则 x_i 和 y_j 相连，其中每一条边的容量为1

计算方法：跑完一次最大流之后，遍历 x_i ， 若 x_i 不与任何 y_j 有流，那么说明某一条路径的最后一个结点是 i,否则若 x_i 与 y_j 有流，则说明在某一条路径中 $i \to j$

分析：①由于x_i 被遍历了，所以所有的点都在路径中，而由于边的容量为1，不存在 $i_1 \to j,\ i_2 \to j,\ i_1 \neq i_2$ 的情形，则一个点只能在一条路径里，所以一次流的分配就对应一个路径覆盖

②x_i 与 y_j 之间共有 E 条边，若最小路径覆盖为 x, 则在原图中一定有 $x-1$ 条边没有在路径中，那么就有 $x-1$ 条边没有流满， 即流 $f+x-1=E$, 要使 x 最小，那么 f 就必须是最大流。

b.



这是个有环图，按照上面的方法如果从1开始的话，会找到 $1 \to 2 \to 3$ 和 $4 \to 3$ 两条路径，而实际只需要 $4 \to 3 \to 1 \to 2$ 一条路径