Question 1

1.为了评价在一个有向图中两个结点是“连通得有多好”，人们不仅可以看它们之间的最短路径的长度，而且也可以计数最短路径的条数。

在边的费用具有某些限制的条件，已经证明了这是一个可以有效求解的问题。假设给我们一个边上带有费用的有向图$G = (V, E)$；费用可能是正的或者负的，但是图中的每个圈严格有着正的费用。还给定两个结点$v, w \in V$。给出一个有效的算法计算G中最短路$v - w$路径的条数。（算法不必列出所有的路径；只要数目就足够了。）

可以采用Floyd算法计算每对结点的最短路径，每次更新时判断更新路径$d(v, t) + d(t, w)$与当前路径$d(v, w)$的大小。如果大$d(v, t) + d(t, w) < d(t, w)$，则设置$C(v, w) = 1$，如果$d(v, t) + d(t, w) = d(t, w)$，则将$C(v, w)$增加1。算法设计如下1.0.1。

Algorithm 1.0.1 Optimal path count

Input: graph $G(V, E)$, v, w

Initialize matrix $dist$ as direct path cost
Initialize matrix C as 0
for k from 1 to $|V|$ **do**
  for i from 1 to $|V|$ **do**
    for j from 1 to $|V|$ **do**
      if $dist[i][k] + dist[k][j] < dist[i][j]$ **then**
        $dist[i][j] = dist[i][k] + dist[k][j]$
        $C[i][j] = 1$
      else if $dist[i][k] + dist[k][j] = dist[i][j]$ **then**
        $C[i][j] += 1$
      end if
    end for
  end for
end for
Return $C[v][2]$

Question 2

2.某加油站有一个大的地下储油罐存储汽油；这个罐一次至多存L加仑。订购油是相当贵的，因此他们希望订货要比较少。每次订货，他们除了所订购油的费用之外，还需要付固定价格P的运费。但是，每加仑油多存**1**天的费用是c，因此提前订购会增加存储的费用。

他们计划在冬天休息一周，希望储罐到休业的时间是空的。幸运的是，基于多年的经验，他们对于直到这个时间之前的每一天将需要多少油有着精确的规划。假定到他们休业还有n天，对于$i = 1, 2, ..., n$的每一天i他们需要g_i加仑汽油。假定储罐在第**0**天结束时是空的。给出一个算法来决定他们应该在哪天订货，以及订多少，以使得他们的总费用最小。

令$W(i, j)$表示在第i天加油,第j天刚好用完的最小消耗(中间可以加油)。于是我们可以得到：

$$W(i,j) = \begin{cases} P, & (i = j) \\ \min\{\min_{k=i}^{j-1}\{W(i,k) + W(k+1,j)\}, P + \sum_{m=i+1}^{j}(m-i)cg_m\}, & (i < j) \end{cases} \tag{2.1}$$

算法设计如下 2.0.2

Algorithm 2.0.2 Optimal path count

Input: 4

```
Initialize matrix G and W
for i from 1 to n do
    for j from i + 1 to n do
        Set G[i][j] = sum_{m=i+1}^{j} (m − i) c g_m
    end for
end for
for i from 1 to n do
    for j from i to n do
        Set W[i][j][1] = min{ min_{k=i}^{j−1} {W[i][k] + W[k + 1][j]}, P + G[i][j] }
        Set W[i][j][2] to be its deriving
    end for
end for
Backtrack from W[1][n] by W[i][j][2]
The backtracking result
```

Question 3

3.改划选区（**Gerrymandering**）是以非常小心的方式划分选区的行为，以便得到有利于某个特定政党的选举结果。假设我们有一组n个选区$P_1, P_2, ..., P_n$，每个选区包含m个登记的选民。我们假设把这些选区分成两个地区，每个地区包含$n/2$个选区。现在对每个选区，我们有关于对两个政党的每一个有多少选民登记的信息。如果这组选区可以按下述方式划分成两个地区，使得同一个政党在两个地区中都占多数，我们将说这组选区对于改划选区是敏感的。

给出一个算法来确定给定的一组选区对改划选区是否敏感；你的算法的运行时间应该是n与m的多项式。

例.加入我们有$n = 4$个选区，下面是登记选民的信息。

Table 3.1 数据

Precinct	**1**	**2**	**3**	**4**
Number registered for part A	55	43	60	47
Number registered for part B	45	57	40	53

这组选区是敏感的，因为如果把选区1与4分到一个地区，选区2与3 分到另一个地区，那么政党A将在两个地区都占多数。这说明了改划选区中的不公平性；尽管政党A在整个人口中只保持微弱多数（205对195），但他却能拿到两个地区的全部选举人票。

假设A占大多数，有a个选票，如果是敏感的则有

$$a \geq mn/2 + 2$$

A在$n/2$的选区中要有$s \geq mn/4 + 1$。令$W(k, l, w)$表示从前k个选区中选择一组有l个选区，其中有w个A选民。a_n表示P_n选区中选民的数量，$true$表示敏感，$false$表示不敏感。则有：

$$W(k, l, w) = \begin{cases} true & W(k-1, l-1, w-a_k) = true \\ true & W(k-1, l, w) = true \\ true & l = 0, w = 0 \\ true & k = 1, l = 1, w = a_1 \\ false & otherwise \end{cases} \tag{3.1}$$

运行时间为$O(n^3 m)$。

Question 4

Decide whether you think the following statement is true or false. If it is true, give a short explanation. If it is false, give a counterexample.

Let G be an arbitrary flow network, with a source s, a sink t, and a positive integer capacity c_e on every edge e. If f is a maximum s-t flow in G, then f saturates every edge out of s with flow(i.e. for all edges e out of s, we have $f(e) = c_e$).

It is false. Consider the example in Pic 4.1. We can see that $e(s, v_4)$ is not equal to its

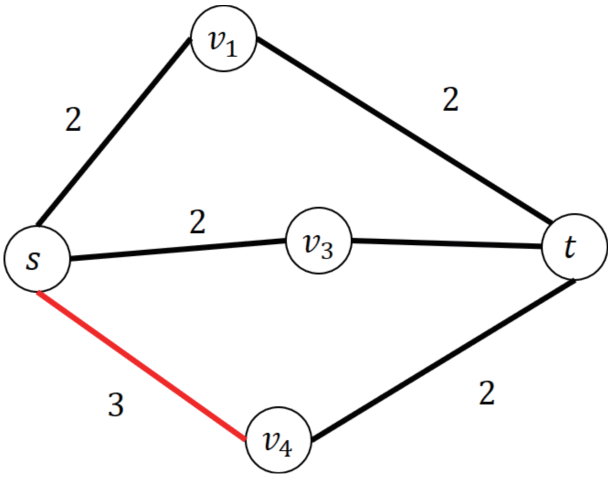

Figure 4.1 An example

capacity.

Question 5

Decide whether you think the following statement is true or false. If it is true, give a short explanation. If it is false, give a counterexample.

Let G be an arbitrary flow network, with a source s, a sink t, and a positive integer capacity c_e on every edge e; and let (A,B) be a minimum s-t cut with respect to these capacities $\{c_e : e \in E\}$. Now suppose we add 1 to every capacity; then (A,B) is still a minimum s-t cut with respect to these new capacities $\{1 + c_e : e \in E\}$.

It is false. Let see the example in Fig 5.1 and Fig 5.2. We can see that in Fig 5.1, the red

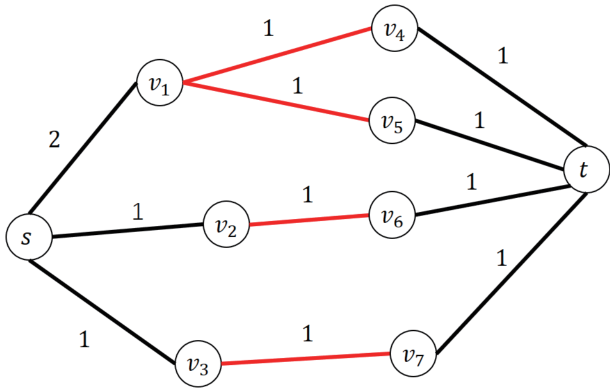

Figure 5.1 An example

edges compose a cut. When we add 1 to each edge, it no more than a cut instead the cut is the composition of the red edges in Fig 5.2.

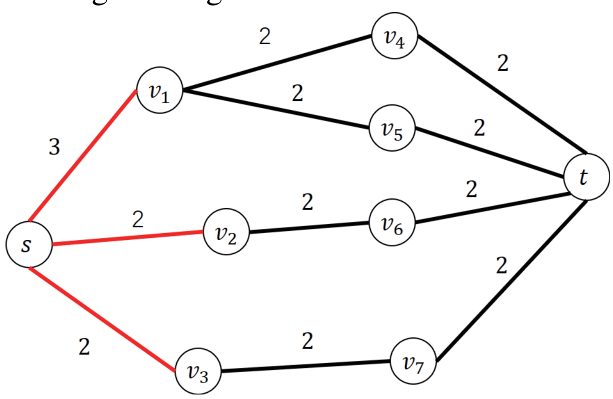

Figure 5.2 An example