1 flow - solution

1.1 point 1

枚举所有可能被选的点集合并计算最大收益。
时间复杂度 $O(2^N N)$ 。

1.2 point 2, 3

每个点是否需要付出额外代价仅与其父亲有关，用动态规划解决。
令 $f[i][k]$ 表示以 i 为根的子树中，保证 i 点的选取状态为 k (k = 1 表示选取) 的最大收益。
转移很简单，略。
时间复杂度 $O(N)$ 。

1.3 point 4, 5, 6

用 x_i 表示点 i 是否被选。 $x_i = 1$ 表示选择 i 点，则我们的收益式为：

$$Ans = max\left[\sum V_i x_i + \sum max(x_i - min(x_{f_i}, x_{f_{f_i}}, ..., x_{f_i^k}), 0)P_i\right]$$

其中， f_i^k 表示 i 的第 k 个父亲。
最大收益 = 总收益 - 最小损失 ，转化式子：

$$Ans = \sum_{V_i>0} V_i - min\left[\sum_{V_i>0}(1-x_i)V_i + \sum_{V_i<0} x_i|V_i| + \sum max(x_i - min(x_{f_i}, x_{f_{f_i}}, ..., x_{f_i^k}), 0)P_i\right]$$

上式可转化经典的最小割模型：

$$min \sum_{E(i,j)} max(x_i - x_j, 0)W_{i,j}$$

其中， $W_{i,j} \geq 0, x_i \in [0, 1]$ 。
$x_i = 1$ 表示点 i 与源点 S 连通， $x_i = 0$ 表示与汇点 T 连通。

再回到我们之前的式子。我们分别考虑三个求和式。

1、$(1 - x_i)V_i \Rightarrow max(1 - x_i, 0)V_i \Rightarrow max(x_S - x_i, 0)V_i$

2、$x_i|V_i| \Rightarrow max(x_i - 0, 0)|V_i| \Rightarrow max(x_i - x_T, 0)|V_i|$

3、$max(x_i - min(x_{f_i}, x_{f_{f_i}}, ..., x_{f_i^k}), 0)P_i$

用一个点 y_i 代替 $min(x_{f_i}, x_{f_{f_i}}, ..., x_{f_i^k})$ ，则变为 $max(x_i - y_i, 0)P_i$ 。
重点是如何构图，才能使 y_i 能够代替 min 式。
考虑将 min 拆分成 $y_i \leq x_{f_i}$, $y_i \leq x_{f_{f_i}}$, ..., $y_i \leq x_{f_i^k}$ 。
对于一个 $y_i \leq x_k$ ，可以构边 $+\infty max(y_i - x_k, 0)$ ，这样就保证不会有 $y_i = 1$ 且 $x_k = 0$ 。
如此第三个求和式的构图也解决。最终构图完成后跑最小割即可。
网络流的点数为 $O(N)$ ，边数为 $O(N^2)$ 。

1.4 point 7, 8, 9, 10

注意到对于一个点 i ， y_i 需要向其**连续**的若干个父亲的 x 连容量为无穷大的边。
如果树是一条链的话，就相当于是 y_i 向一个区间内的所有 x 连边。
可以用类似线段树的思想解决。
再回到树的情况。则可以用树链剖分解决。
最终，网络流的点数为 $O(N)$ ，边数为 $O(N \log^2 N)$ 。