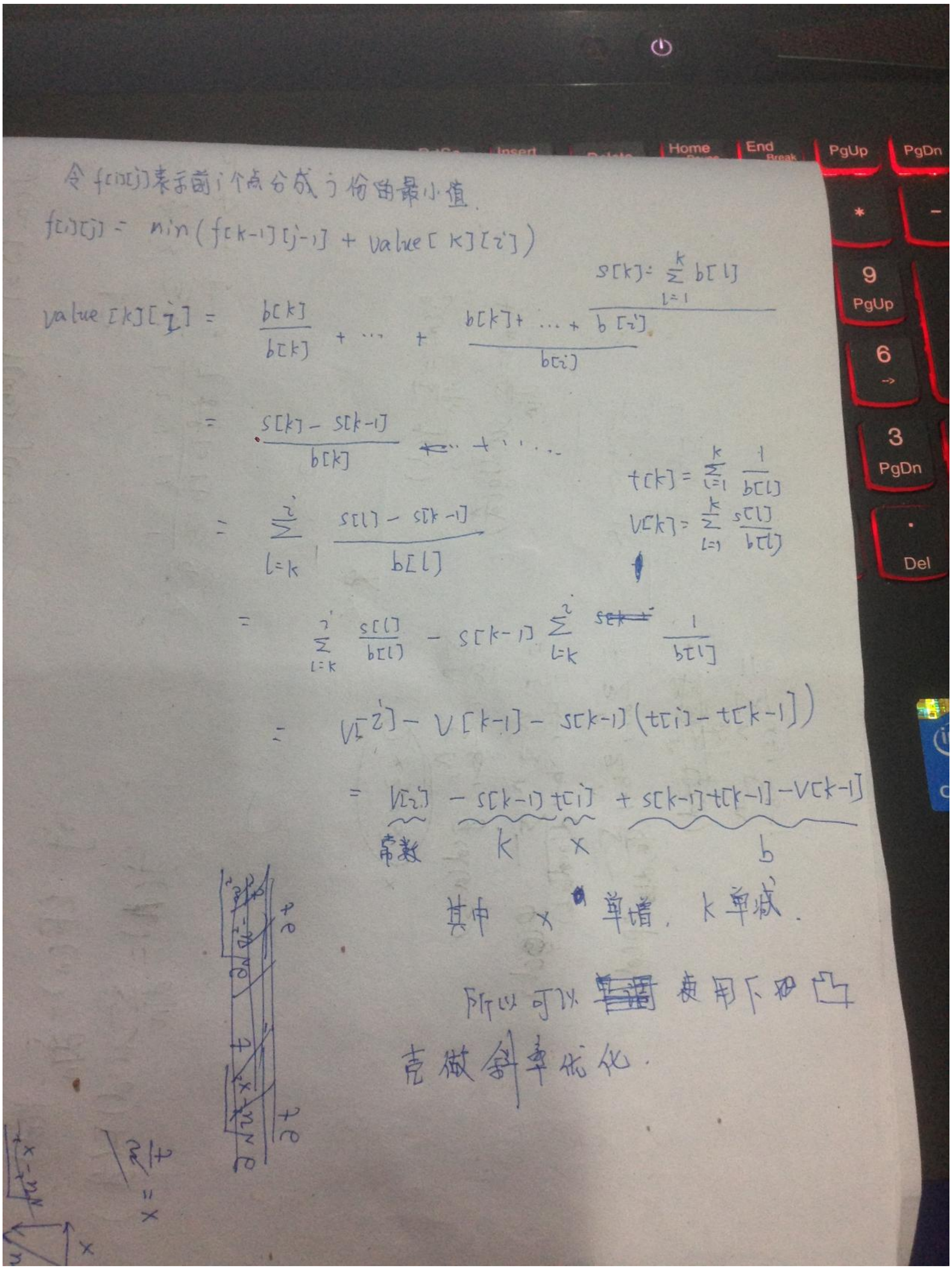
Math：


令f[i][j]表示前i个点分成j份的最小值.

$$f[i][j] = \min(f[k-1][j-1] + value[k][i])$$

$$s[k] = \sum_{l=1}^{k} b[l]$$

$$value[k][i] = \frac{b[k]}{b[k]} + \cdots + \frac{b[k] + \cdots + b[i]}{b[i]}$$

$$= \frac{s[k] - s[k-1]}{b[k]} + \cdots$$

$$= \sum_{l=k}^{i} \frac{s[l] - s[k-1]}{b[l]}$$

$$t[k] = \sum_{l=1}^{k} \frac{1}{b[l]}$$

$$V[k] = \sum_{l=1}^{k} \frac{s[l]}{b[l]}$$

$$= \sum_{l=k}^{i} \frac{s[l]}{b[l]} - s[k-1] \sum_{l=k}^{i} \frac{1}{b[l]}$$

$$= V[i] - V[k-1] - s[k-1](t[i] - t[k-1])$$

$$= \underbrace{V[i]}_{常数} \underbrace{- s[k-1]}_{k} \underbrace{t[i]}_{x} + \underbrace{s[k-1]t[k-1] - V[k-1]}_{b}$$

其中 x 单增, k 单减.

所以可以 使用下凸壳做斜率优化.

Shortest：

考虑最简单的最短路做法。Dij+堆。

但是在这里数字会很大，具体地说，每个值都要用一个长度为 n 的 01 数组记下来，所以不能保证复杂度。

但是我们观察 Dij 中的每次转移，其实他都只改了出发点的一段连续区间和一个点的值。因此我们就考虑使用函数式线段树来解决这个问题。

为了比较两个值的大小，我们可以在线段树中记录 hash 值从而找到第一个不同的位置。所以时间复杂度为 O(nlogn)。

Maximum

用网络流来做这道题。

先不考虑限制，那么把每个发电站拆成 r[i]-l[i]+2 个点，每个点代表一个不同的 level。

我们从源点 S 向每个发电站的第一个点连一条流量为无穷大的边，从每个发电站的最后一个点向汇点 T 连一条流量为无穷大的边，中间的相邻点之间连边，流量为 MAX-此点代表的发电量。跑一次 dinic，答案即为 MAX*n-MAXFLOW。

现在考虑限制，对于每个限制 u，v，x，对于每个 u 的 level L，拿 u 的这个 level L 朝 v 的 level L – x 连一条费用为无穷大的边。

跑一次 dinic，答案即为 MAX*n-MAXFLOW。