## 括号序列

首先我们处理一下哪些括号是匹配的。

`W(X()X(Z()Z)X(Y()Y()Y)X()X)W`，我们把括号之间的间隔标号，对于一堆匹配的括号，要求两端标的号是相同的，那么一段序列是合法的括号序列当且仅当两边的间隔标的号是相同的。那么问题就转化成有若干个标号，你要统计位置两端有多少对标号是相同的。这个问题只要从左往右扫一下，维护一下变化量即可。

时间复杂度是$O(n)$。

## 序列划分

首先考虑$(D,10)=1$ 的情况，记$S_i=\left(\sum s_i\cdot 10^{|s|-i}\right) \bmod D$，也就是从$i$这个位置到结尾的值，那么$s[i\ldots j]$能被$D$整除，当且仅当$S_i=S_{j+1}$。

记$dp_{i,0/1}$表示前往后，划分到$i$，当前段能否被$D$整除，那么如果当前段能被整除，那么必须从$S_j=S_{i+1}$转移过来，否则从不被整除的位置转移过来，记录一下$S_j$等于某个值的 dp 值之和，从前往后转移即可。

如果$(D,10)\neq 1$，那么上面的条件不成立。令$D=D_1D_2$，其中$D_1$与 10 互质，$D_2$里只包含因子 2,5，令$S_i$表示从$i$这个位置到结尾的值对$D_1$取模，那么$s[i\ldots j]$能被$D$整除，当且仅当$S_i=S_{j+1}$，且这个数能被$D_2$整除。在十进制下，被 2,5 的幂次整除，我们只需要看后若干位即可，这里$D$不是很大，所以只需要看后 20 位。所以类似于上面的$dp$。从前面的位置$j$转移到$i$，如果$j$已经小于$i-20$，并且后 20 满足条件，那么用上述的方法转移即可。对于$i-20\sim i-1$，暴力判断转移即可。

时间复杂度是$O(n\log D)$。

## 矩阵删除

令$f_{i,j}$表示切到$(i,j)$处对应的不同轨迹数量，那么有转移$f_{i,j}=\sum_{l=j-k}^{j+k}f_{i-1,l}$。

但是注意到对于一行连续相同的一段数字，删除任意一个得到的是相同的结果，故直接计算轨迹数量显然会重，考虑去重，以$g_{i,j}$表示切到 $(i,j)$处所得到的轨迹中与切到$(i,j-1)$处得到的轨迹等价的数量，显然若$s_{i,j-1}$和 $s_{i,j}$不相同时，到达两者的轨迹不会等价。考虑$s_{i,j}=s_{i,j-1}$，其相交贡献区间为 $[j-k,j+k-1]$，那么有

$g_{i,j}=\sum_{l=j-k}^{j+k-1}f_{i-1,l}-\sum_{l=j-k+1}^{j+k-1}g_{i-1,l}$，

$f_{i,j}=\sum_{l=j-k}^{j+k}f_{i-1,l}-\sum_{l=j-k+1}^{j+k}g_{i-1,l}$。

注意到都是区间加，故求一遍前缀和即可，时间复杂度 $O(nm)$。

## 路径查询

类似最小生成树，考虑从小到大加边，如果加完一条边之后，$u, v$这两个点所在的连通块，只需要再加一条边就可以连通，那么这条边就是第二小的，也就是答案。

接着考虑如何维护这一过程。对于一个连通块$u$，记与这个连通块有边相连的点为$N(u)$。考虑启发式合并，对于两个连通块$u, v$合并后，那么可能$N(u)$中的点和$v$有边相连了，$N(v)$中的点和$u$有边相连了。如果直接遍历$N(u)$和$N(v)$中的点，那么时间复杂度是显然不对的。

我们可以做的是，遍历$N(u)$的所有点，然后查询是不是$v$相连，然后遍历$u$里面的所有询问，看是不是在$N(v)$之中即可。这样只需要遍历其中一部分的信息，所以可以启发式合并。每次我们按$N(u)$大小加询问个数比较，然后遍历小的部分，启发式合并即可。合并完之后，更新一下合并之后的连通块的邻居和询问信息。

时间复杂度是$O(n\log n)$的。