

# 算法基础LAB5实验报告

PB19071535徐昊天

## 一.实验内容

1. 给定一个带权有向图G，输出平均权重最小的环。
2. 给定一个二分图G，找到图中最大匹配中权重最小的一个。
3. 对于一个未知的大整数N，已知它在 $p_i$ 进制下的尾数为 $r_i$ ，找到满足条件的最小正整数 $N_0$ 。
4. 给定一个字符串P，在 $O(n)$ 时间内输出所有存在子串x，满足 $P = x^r$ 的重复因子r。

## 二.实验目的

1. 利用OJ编程，强化编程能力，培养编程细节，提升编程素养。
2. 熟悉书本和课堂上涉及的图论、数论、字符串匹配算法，进一步巩固算法知识。

## 三.实验内容

### I.最小平均权重环

1. 选取顶点1作为源节点。
2. 利用一个二维vector value储存源节点到达所有节点的长度分别为0-n的最短路径权重。例如value[5][7]表示源节点到达结点8的路径长度为5的最短路径权重。
3. 对于每一个结点v遍历找到该结点对应的 $\max(\text{value}[n][v] - \text{value}[k][v])$ ， $k \in [0, n-1]$ 。
4. 遍历每一个结点对应的 $\max(\text{value}[n][v] - \text{value}[k][v])$ 并取最小值即为所求最小平均权重。

### II.最小权最大匹配

1. 顶标初始化。遍历每个从第一部分顶点出发的所有边，找到权重最大的边赋为对应顶点的顶标值，第二部分点全部初始化为0。
2. 寻找完备匹配。利用类匈牙利算法，对第一部分的顶点寻找增广路径，若找到了增广路径则完成匹配。
3. 若增广失败，则需要考虑扩大相等子图来继续找增广路。即首先通过访问所有交错树中的第一部分顶点相连的不在交错树中的第二部分顶点，对于每条这样的边找到顶标和减去边权值的最小值作为sub值。对第一部分的顶点顶标值加sub，对第二部分的顶点顶标值减sub。

### III.尾数还原

1. 首先参考算法导论中的中国余数定理部分，将该尾数还原问题用中国余数定理解决。其中需调用数论算法中的EXTENDED-EUCLID算法用于求解指定数据的乘法逆元。
2. 由于数据会超出编程语言中的long long范围，故需要实现高精度运算。高精度数用一个结构体储存，其中包括一个存储数字的字符串，一个用于存储字符串长度的整型数，一个用于储存高精度数的正负号的整型数。
3. 对于高精度加减乘法，需按位运算并将结果存储到一个新的高精度数中；对于除法，需将除数的位提高到被除数的位并执行减法，并将除数的位逐渐递减，最后将商和模分别存储在一个高精度数中。
4. 将中国余数定理中的加减乘除和取模运算用高精度算法代替，即可用于处理存在大数的情况。

## IV.重复子串因子

1. 对由两个P连接而成的字符串P'执行KMP字符串匹配，在P'中匹配字符串P，寻找所有能够匹配到P的位置下标。
2. 将P'中能匹配到P且满足不到P'一半(若超过P'一半则不能用于作为重复子串)的位置下标，用字符串P的长度除以位置下标即为所求重复因子。
3. 由于KMP字符串匹配是由前向后匹配，故匹配到的位置下标逐渐增大，重复因子逐渐减小。根据题意需从小到大输出所有重复因子，故可用栈储存所有重复因子后输出。

由于此算法是在KMP算法的基础上实现，且对应的两个参数字符串长度分别为 $2n$ 、 $n$ ，根据KMP算法的性质可知可在 $O(n)$ 时间内输出所有重复因子。

## 四.实验收获与感想

1. 利用OJ编程，养成了更好的编程习惯，促进自身追求更优美的代码风格，提升了编程素养。
2. 提升了编程能力，提前适应上机编程的难度，为将来刷算法题或机试打下了基础。
3. 增强了对代码细节的把控能力，对程序的追求不止于结果正确，而是追求更低的时间空间消耗、数据的完整性和精确性等。
4. 增强了对经典图论、数论、字符串匹配算法的理解，通过复现算法更加深入了解了算法的原理和过程。