



Hash

河南省实验中学
信息技术组

Hash

字符串 Hash

例题

雪花雪花

兔子与兔子

字符串模式匹配

最长回文子串

后缀数组

其他案例

小结与练习

小结

练习

Hash

河南省实验中学信息技术组

2026年02月26日



Hash

Hash

河南省实验中学
信息技术组

Hash

字符串 Hash

例题

雪花雪花
兔子与兔子
字符串模式匹配
最长回文子串
后缀数组
其他案例

小结与练习

小结
练习

- Hash 指的是将复杂信息映射为一个整数的以实现快速判断信息相同的方法。
- 例如，在长度为 n 的整数序列 a 中统计每个数出现的次数。
 - 如果 a 中的值都比较小时，可以直接用数组标记统计即可。
 - 如果 a 中的值比较大时，可以排序后依次统计即可 (类似离散化)。
 - 或者定义一个函数 $H(x) = (x \bmod P) + 1$ ，其中 P 一般是一个质数。然后利用该函数将序列 a 中的元素的值映射到 $[1, P]$ ，那么对应相同的数字，它们映射的结果也是相同的，对于不同的数字，它们映射的结果大概率是不相同的 (如果相同，需要额外处理)。
- 上述案例中，定义的函数 $H(x)$ 被称为 Hash 函数。
- 上述案例中，数据映射的范围为 $[1, P]$ ，那么可能存在不同的数字映射成相同的结果，这就被称为 Hash 冲突。应对 Hash 冲突有很多方案，一般有如下两种：
 - 拉链法
 - 开放寻址法
 - 再 Hash 法



拉链法

Hash

河南省实验中学
信息技术组

Hash

字符串 Hash

例题

雪花雪花
兔子与兔子
字符串模式匹配
最长回文子串
后缀数组
其他案例

小结与练习

小结
练习

- 建立一个邻接表，将 Hash 函数的值相同的元素放到同一个链表中，链表的结点中可以保存原始的数据信息等方便判定。
- 对于新增元素或查询元素，只需要先计算 Hash 函数的值，然后到对应链表中插入或查询即可。
- 如果 Hash 函数设计的较好时，可以期望信息被均匀的分布再各个链表中，那么每次查询的时间复杂度期望为 $O(K)$ ，其中 K 是一个常数。



再 Hash 法

Hash

河南省实验中学
信息技术组

Hash

字符串 Hash

例题

雪花雪花

兔子与兔子

字符串模式匹配

最长回文子串

后缀数组

其他案例

小结与练习

小结

练习

- 设计另外一个 Hash 函数 $H_2(x)$ ，对原始信息再次进行 Hash。
- 这种方法需要计算两次 Hash 函数，计算量会略有增大。



Hash

Hash

河南省实验中学
信息技术组

Hash

字符串 Hash

例题

雪花雪花

兔子与兔子

字符串模式匹配

最长回文子串

后缀数组

其他案例

小结与练习

小结

练习

- Hash 函数是 Hash 方法的核心部分。
- Hash 函数应当尽量满足：值均匀分布、冲突少，计算量低。



字符串 Hash

Hash

河南省实验中学
信息技术组

Hash

字符串 Hash

例题

雪花雪花
兔子与兔子
字符串模式匹配
最长回文子串
后缀数组
其他案例

小结与练习

小结
练习

- 字符串 Hash 就是把任意长度的字符串映射成一个非负整数，并且其冲突概率几乎为零。
- 信息学奥赛中一般采用多项式哈希的方式：取一固定值 P^1 ，把字符串看作 P 进制数，并分配一个大于 0 的数值，代表每种字符。例如，对于小写字母构成的字符串，可以令 $a = 1, b = 2, \dots, z = 26$ ，那么字符串 "hasyzx" 可以转化为 $8 \times P^5 + 1 \times P^4 + 19 \times P^3 + 25 \times P^2 + 26 \times P^1 + 24 \times P^0$ 。
- 为了避免数值过大导致溢出，一般会取一个固定值 M ，求出该 P 进制数对 M 的余数，作为该字符串的 Hash 值。通常我们取 $M = 2^{64}$ ，即直接使用 **unsigned long long** 类型，可以实现自动溢出取模²。

¹一般来说，取 $P = 131$ 或 $P = 13331$ 。

²有出题人针对这种取余方式，设计数据使得出现 Hash 冲突，可以取一个较大的质数，最好是别人都不知道的。也可以取另外一些 P 和 M ，进行多次哈希，当结果都相同时才认为字符串相等。



字符串 Hash

Hash

河南省实验中学
信息技术组

Hash

字符串 Hash

例题

雪花雪花

兔子与兔子

字符串模式匹配

最长回文子串

后缀数组

其他案例

小结与练习

小结

练习

- 字符串 Hash 的好处是：如果预处理出所有字符串的前缀 Hash 值，就可以在 $O(1)$ 的时间内求出任意子串的 Hash 值。
- 对于字符串 $S = \text{"hasyzx"}$ ，其前缀 $T_1 = \text{"hasy"}$ 的 Hash 值为 $H(T_1) = 8 \times P^3 + 1 \times P^3 + 19 \times P^1 + 25 \times P^0$ 、 $T_2 = \text{"ha"}$ 的 Hash 值为 $H(T_2) = 8 \times P^1 + 1 \times P^0$ ，那么子串 $T = \text{"sy"}$ 的 Hash 值为 $H(T) = 19 \times P^1 + 25 \times P^0 = H(T_1) - H(T_2) \times P^2$ 。

```
1 const int N = 1e6 + 10;
2 const int P = 131; // 或 13331
3 unsigned long long p[N]; // 预处理 P 的指数
4 unsigned long long f[N]; // 字符串 Hash 前缀
5 // 预处理
6 f[0] = 0, p[0] = 1; // P^0
7 for(int i = 1; i <= n; ++i)
8 {
9     f[i] = f[i - 1] * P + (s[i] - 'a' + 1);
10    p[i] = p[i - 1] * P;
11 }
12 // 获取子串 s[l~r] 的 Hash 值
13 t = f[r] - f[l - 1] * p[r - l + 1];
```



【例】雪花雪花

Hash

河南省实验中学
信息技术组

Hash

字符串 Hash

例题

雪花雪花

兔子与兔子

字符串模式匹配

最长回文子串

后缀数组

其他案例

小结与练习

小结

练习

有 n 片雪花，每片雪花由六个角组成，每个角都有长度。第 i 片雪花六个角的长度从某个角开始顺时针依次记为 $a_{i,1}, a_{i,2}, a_{i,3}, a_{i,4}, a_{i,5}, a_{i,6}$ 。

因为雪花的形状是封闭的环形，所以从任何一个角开始顺时针或逆时针往后记录长度，得到的六元组都代表形状相同的雪花。例如，

$a_{i,1}, a_{i,2}, a_{i,3}, a_{i,4}, a_{i,5}, a_{i,6}$ 和 $a_{i,2}, a_{i,3}, a_{i,4}, a_{i,5}, a_{i,6}, a_{i,1}$ 就是形状相同的雪花，

$a_{i,1}, a_{i,2}, a_{i,3}, a_{i,4}, a_{i,5}, a_{i,6}$ 和 $a_{i,6}, a_{i,5}, a_{i,4}, a_{i,3}, a_{i,2}, a_{i,1}$ 也是形状相同的雪花。

我们称两片雪花形状相同，当且仅当它们各自从某一角开始顺时针或逆时针记录长度，能得到两个相同的六元组。

求这 n 片雪花中是否存在两片形状相同的雪花。



【例】雪花雪花

Hash

河南省实验中学
信息技术组

Hash

字符串 Hash

例题

雪花雪花

兔子与兔子

字符串模式匹配

最长回文子串

后缀数组

其他案例

小结与练习

小结

练习

【输入格式】

第一行输入一个整数 n ，代表雪花的数量。

接下来 n 行，每行包含 6 个整数描述一片雪花，分别代表雪花的六个角的长度 (这六个数即为从雪花的随机一个角顺时针或逆时针记录长度得到)。

【输出格式】

如果不存在两片形状相同的雪花，则输出：No two snowflakes are alike.

如果存在两片形状相同的雪花，则输出：Twin snowflakes found.

【样例输入】

```
2
1 2 3 4 5 6
4 3 2 1 6 5
```

【样例输出】

```
Twin snowflakes found.
```



【例】雪花雪花

Hash

河南省实验中学
信息技术组

Hash

字符串 Hash

例题

雪花雪花

兔子与兔子

字符串模式匹配

最长回文子串

后缀数组

其他案例

小结与练习

小结

练习

- 定义 Hash 函数

$$H(a_{i,1}, a_{i,2}, a_{i,3}, a_{i,4}, a_{i,5}, a_{i,6}) = \left(\sum_{j=1}^6 a_{i,j} + \prod_{j=1}^6 a_{i,j} \right) \bmod P$$

其中 P 是一个质数。显然，如果两片形状相同的雪花，它们六角的长度之和、长度之积都相等，它们的 Hash 值也相等。

- 但是有一定概率两片不同的雪花的 Hash 值相同，因此运用拉链法，将 Hash 值相同的雪花插入同一个链表中。
- 对于雪花 i ，需要查询其 Hash 值的链表，判断雪花是否形状相同。
- 期望时间复杂度： $O\left(\frac{N^2}{P}\right)$ ，取 P 为最接近 N 的质数。



【例】雪花雪花

Hash

河南省实验中学
信息技术组

Hash

字符串 Hash

例题

雪花雪花

兔子与兔子

字符串模式匹配

最长回文子串

后缀数组

其他案例

小结与练习

小结

练习

```
1 int Hash(int i)
2 {
3     int s = 0, r = 1;
4     for(int j = 0; j < 6; ++j)
5         s = (s + a[i][j]) % P, r = (long long)r * a[i][j] % P;
6     return (s + r) % P;
7 }
8
9 bool equal(int i, int j)
10 {
11     for(int y = 0; y < 6; ++y) // 枚举第二个雪花的记录起点
12     {
13         bool eq = true;
14         for(int k = 0; k < 6; ++k) // 顺时针记录
15             if(a[i][k] != a[j][(y + k) % 6]) eq = false;
16         if(eq) return true;
17         eq = true;
18         for(int k = 0; k < 6; ++k) // 逆时针记录
19             if(a[i][k] != a[j][(y - k + 6) % 6]) eq = false;
20         if(eq) return true;
21     }
22     return false;
23 }
```



【例】雪花雪花

Hash

河南省实验中学
信息技术组

Hash

字符串 Hash

例题

雪花雪花

兔子与兔子

字符串模式匹配

最长回文子串

后缀数组

其他案例

小结与练习

小结

练习

```
1 bool insert(int i)
2 {
3     int x = Hash(i);
4     for(int j = head[x]; j; j = nxt[j])
5         if(equal(i, j)) return true;
6     nxt[i] = head[x], head[x] = i;
7     return false;
8 }
9
10 scanf("%d", &n);
11 for(int i = 1; i <= n; ++i)
12 {
13     for(int j = 0; j < 6; ++j) scanf("%d", &a[i][j]);
14     if(insert(i))
15     {
16         puts("Twin snowflakes found.");
17         return 0;
18     }
19 }
20 puts("No two snowflakes are alike.");
```



【例】兔子与兔子

Hash

河南省实验中学
信息技术组

Hash

字符串 Hash

例题

雪花雪花

兔子与兔子

字符串模式匹配

最长回文子串

后缀数组

其他案例

小结与练习

小结

练习

很久很久以前，森林里住着一群兔子。

有一天，兔子们想要研究自己的 DNA 序列。我们首先选取一个好长好长的 DNA 序列（小兔子是外星生物，DNA 序列可能包含 26 个小写英文字母），然后我们每次选择两个区间，询问如果用两个区间里的 DNA 序列分别生产出来两只兔子，这两个兔子是否一模一样。

注意两个兔子一模一样只可能是他们的 DNA 序列一模一样。



【例】兔子与兔子

Hash

河南省实验中学
信息技术组

Hash

字符串 Hash

例题

雪花雪花

兔子与兔子

字符串模式匹配

最长回文子串

后缀数组

其他案例

小结与练习

小结

练习

【输入格式】

第一行一个 DNA 字符串 $s(|s| \leq 10^6)$ 。

第二行一个整数 $m(m \leq 10^6)$ ，表示 m 次询问。

接下来 m 行，每行四个数字 l_1, r_1, l_2, r_2 ，表示此次询问的两个区间，注意字符串的位置从 1 开始编号。

【输出格式】

对于每次询问，输出一行表示结果。

如果两只兔子完全相同输出 Yes，否则输出 No。

【样例输入】

```
aabbaabb
3
1 3 5 7
1 3 6 8
1 2 1 2
```

【样例输出】

```
Yes
No
Yes
```



【例】兔子与兔子

Hash

河南省实验中学
信息技术组

Hash

字符串 Hash

例题

雪花雪花

兔子与兔子

字符串模式匹配

最长回文子串

后缀数组

其他案例

小结与练习

小结

练习

- 对串 s 进行 Hash 前缀预处理，对于询问区间 l_1, r_1 和 l_2, r_2 ，只需判断 Hash 值是否相同即可。
- 时间复杂度 $O(|S| + Q)$ 。

```
1 p[0] = 1;
2 for(int i = 1; i <= n; ++i)
3 {
4     f[i] = f[i - 1] * P + s[i] - 'a' + 1;
5     p[i] = p[i - 1] * P;
6 }
7 int q; scanf("%d", &q);
8 while(q--)
9 {
10     int l1, r1, l2, r2;
11     scanf("%d%d%d%d", &l1, &r1, &l2, &r2);
12     unsigned long long x = f[r1] - f[l1 - 1] * p[r1 - l1 + 1];
13     unsigned long long y = f[r2] - f[l2 - 1] * p[r2 - l2 + 1];
14     if(x == y) puts("Yes"); else puts("No");
15 }
```



【例】字符串模式匹配

Hash

河南省实验中学
信息技术组

Hash

字符串 Hash

例题

雪花雪花

兔子与兔子

字符串模式匹配

最长回文子串

后缀数组

其他案例

小结与练习

小结

练习

- 给定长度为 n 的字符串 s 和长度为 m 的字符串 t ，判断字符串 t 是否为字符串 s 的子串，并求出字符串 t 在字符串 s 中出现的位置。
- 按照定义进行匹配的朴素算法的时间复杂度为 $O(NM)$ ，耗时过长。
- KMP 算法可以在 $O(N + M)$ 的时间复杂度内解决，但是学习成本高³。
- 那么利用字符串 Hash 如何解决呢？显然，按照定义从左向右分别求出与 t 串相同长度的子串的 Hash 值，如果与 t 串值相同，则说明 t 为 s 的子串。
- 当然，可能出现 Hash 冲突导致两个子串不同但 Hash 值相同，可以进行二次或多次 Hash 来解决。

³建议初学者先不学习 KMP 算法。



【例】字符串模式匹配

Hash

河南省实验中学
信息技术组

Hash

字符串 Hash

例题

雪花雪花

兔子与兔子

字符串模式匹配

最长回文子串

后缀数组

其他案例

小结与练习

小结

练习

```
1 // 二次 Hash
2 const int N = 1e6 + 10;
3 const unsigned long long MOD = 1e9 + 7; // 第二种 Hash 的模数
4 const int P1 = 131, P2 = 13331;
5 unsigned long long f1[N], f2[N]; // 字符串 s 的两种 Hash 前缀
6 unsigned long long p1[N], p2[N];
7
8 unsigned long long ht1 = 0, ht2 = 0; // 字符串 t 的 Hash 值
9 for(int i = 1; i <= m; ++i)
10 {
11     ht1 = ht1 * P1 + t[i] - 'a' + 1; // 自然溢出
12     ht2 = (ht2 * P2 % MOD + t[i] - 'a' + 1) % MOD;
13 }
14 for(int i = 1; i <= n - m + 1; ++i)
15 {
16     unsigned long long hs1 = f1[i + m - 1] - f1[i - 1] * p1[m];
17     unsigned long long hs2 = (f2[i + m - 1] + MOD - f2[i - 1] * p2[m] % MOD) % MOD;
18     if(ht1 == hs1 && ht2 == hs2) cout << i << endl; // 找到子串
19 }
```



【例】最长回文子串

Hash

河南省实验中学
信息技术组

Hash

字符串 Hash

例题

雪花雪花

兔子与兔子

字符串模式匹配

最长回文子串

后缀数组

其他案例

小结与练习

小结

练习

【题目描述】

如果一个字符串正着读和倒着读是一样的，则称它是回文串。
给定一个长度为 n 的字符串 s ，求它的最长回文子串。

【输入格式】

一行一个字符串 s ，输入保证字符串长度不超过 10^6 。

【输出格式】

一行一个整数，表示最长回文子串长度。

【样例输入】

abacacbaaaab

【样例输出】

6



【例】最长回文子串

Hash

河南省实验中学
信息技术组

Hash

字符串 Hash

例题

雪花雪花

兔子与兔子

字符串模式匹配

最长回文子串

后缀数组

其他案例

小结与练习

小结

练习

- 枚举所有子串，判断子串是否为回文。
- 可以用字符串 Hash 来进行回文判断，在 $O(1)$ 的时间内完成判定。
- 为了快速计算子串 Hash，除了预处理前缀 Hash 值，还需要预处理后缀 Hash 值。
- 时间复杂度 $O(N^2)$ 。⁴

```
1 int ans = 1;    // 长度为 1 的子串必然是回文串
2 for(int i = 2; i <= n; ++i)    // 子串长度
3     for(int l = 1; l <= n - i + 1; ++l)
4         {
5             int r = l + i - 1;
6             unsigned long long hs1 = f1[r] - f1[l - 1] * p[r - l + 1];
7             unsigned long long hs2 = f2[l] - f2[r + 1] * p[r - l + 1];
8             if(hs1 == hs2) { ans = i; break; }
9         }
```

⁴还有一种动态规划方法，请读者思考。



【例】最长回文子串

Hash

河南省实验中学
信息技术组

Hash

字符串 Hash

例题

雪花雪花
兔子与兔子
字符串模式匹配
最长回文子串
后缀数组
其他案例

小结与练习

小结
练习

- 回文串的特点是关于中心互为镜像，那么对于任意一个中心，可以判定其为中心的子串是否为回文串。
- 那么可以二分左右子串的长度，判断产生的子串是否满足 Hash 值相等，可以在 $O(\log N)$ 的时间内完成。
- 注意回文子串的长度为偶数和奇数时，中心位置不同。

回文子串长度为**奇数** a a b b c a | b | a c b a b

i

回文子串长度为**偶数** a c a b c d | d c b a b c

i

- 时间复杂度 $O(N \log N)$ 。



【例】最长回文子串

Hash

河南省实验中学
信息技术组

Hash

字符串 Hash

例题

雪花雪花
兔子与兔子
字符串模式匹配
最长回文子串
后缀数组
其他案例

小结与练习

小结
练习

```
1 // 枚举中心
2 for(int i = 1; i <= n; ++i)
3 {
4     // 偶回文子串
5     int low = 0, high = min(i, n - i); // 最长偶回文子串的长度的一半
6     while(low < high)
7     {
8         int mid = (low + high + 1) >> 1;
9         int l = i - mid + 1, r = i + mid;
10        unsigned long long hs1 = f1[r] - f1[l - 1] * p[r - l + 1];
11        unsigned long long hs2 = f2[l] - f2[r + 1] * p[r - l + 1];
12        if(hs1 == hs2) low = mid;
13        else high = mid - 1;
14    }
15    ans = max(ans, low << 1);
16    // 奇回文子串请读者自行完成
17 }
```



【例】最长回文子串

Hash

河南省实验中学
信息技术组

Hash

字符串 Hash

例题

雪花雪花

兔子与兔子

字符串模式匹配

最长回文子串

后缀数组

其他案例

小结与练习

小结

练习

- 定义 $d[i]$ 表示以 i 为右端点的最长回文子串长度，那么显然有 $d[i] \leq d[i-1] + 2$ ，因为以 i 为右端点的回文串去掉两端就是以 $i-1$ 为右端点的回文串。
- 因此从 $i-1$ 递推到 i 时只需要从 $d[i-1] + 2$ 开始递减判断以 i 为右端点的对应长度的子串是否为回文串即可。
- 由于 $+2$ 一共只有 $n-1$ 次，所以递减的总次数不会超过 $2(n-1)$ ，时间复杂度为 $O(N)$ 。
- 有一个名为 Manacher 的算法也可以在 $O(N)$ 的时间复杂度内解决问题，感兴趣自行研究。



【例】最长回文子串

Hash

河南省实验中学
信息技术组

Hash

字符串 Hash

例题

雪花雪花

兔子与兔子

字符串模式匹配

最长回文子串

后缀数组

其他案例

小结与练习

小结

练习

```
1 int ans = 1;
2 d[1] = 1;
3 for(int i = 2; i <= n; ++i)
4 {
5     for(int j = min(d[i - 1] + 2, i); j; --j)    // 注意越界
6     {
7         int l = i - j + 1;
8         unsigned long long hs1 = f1[i] - f1[l - 1] * p[i - l + 1];
9         unsigned long long hs2 = f2[i] - f2[l + 1] * p[i - l + 1];
10        if(hs1 == hs2) { d[i] = j; break; }
11    }
12    ans = max(ans, d[i]);
13 }
```



【例】后缀数组

Hash

河南省实验中学
信息技术组

Hash

字符串 Hash

例题

雪花雪花

兔子与兔子

字符串模式匹配

最长回文子串

后缀数组

其他案例

小结与练习

小结

练习

【题目描述】

后缀数组 (SA) 是一种重要的数据结构, 通常使用倍增或者 DC3 算法实现, 这超出了我们的讨论范围。在本题中, 我们希望使用快排、Hash 与二分实现一个简单的 $O(N \log^2 N)$ 的后缀数组求法。

详细地说, 给定一个长度为 $n(n \leq 3 \times 10^5)$ 的字符串 s (下标 $0 \sim n-1$), 我们可以用整数 $k(0 \leq k < n)$ 表示字符串 s 的后缀 $s(k \sim n-1)$ 。把字符串 s 的所有后缀按照字典序排列, 排名为 i 的后缀记为 $SA[i]$ 。

额外地, 我们考虑排名为 i 的后缀与排名为 $i-1$ 的后缀, 把二者的最长公共前缀的长度记为 $Height[i]$ 。我们的任务就是求出 SA 与 $Height$ 这两个数组。

【输入格式】

一行一个字符串 s , 输入保证字符串长度不超过 3×10^5 。

【输出格式】

第一行为 SA 数组, 数字之间用空格隔开。

第二行为 $Height$ 数组, 数字之间用空格隔开。特别地, 假设 $Height[1] = 0$ 。



【例】后缀数组

Hash

河南省实验中学
信息技术组

【样例 1 输入】

```
abadc
```

【样例 1 输出】

```
0 2 1 4 3
0 1 0 0 0
```

【样例 1 解释】

字符串 `abadc` 的所有后缀分别为 `abadc(0)`、`badc(1)`、`adc(2)`、`dc(3)`、`c(4)`，将他们排序后为 `abadc(0)`、`adc(2)`、`badc(1)`、`c(4)`、`dc(3)`，故 $SA[] = \{0, 2, 1, 4, 3\}$ ，显然 $Height[] = \{0, 1, 0, 0, 0\}$ 。

【样例 2 输入】

```
ponoiiipoi
```

【样例 2 输出】

```
9 4 5 6 2 8 3 1 7 0
0 1 2 1 0 0 2 1 0 2
```

Hash

字符串 Hash

例题

雪花雪花

兔子与兔子

字符串模式匹配

最长回文子串

后缀数组

其他案例

小结与练习

小结

练习



【例】后缀数组

Hash

河南省实验中学
信息技术组

Hash

字符串 Hash

例题

雪花雪花
兔子与兔子
字符串模式匹配
最长回文串
后缀数组
其他案例

小结与练习

小结
练习

- 直接枚举出所有后缀并排序，最坏的情况下时间复杂度为 $O(N^2 \log N)$ 。
- 显然字符串比较是算法的限制因素，可以使用二分法 + Hash，求出两个字符串的最长公共前缀长度，那么通过比较公共前缀的下一个字符即可确定子串大小，时间复杂度为 $O(\log N)$ ，结合快排后的整体时间复杂度为 $O(N \log^2 N)$ 。
- 求出 SA 后，对于相邻后缀，执行相同的二分 + Hash，即可求出 *Height*。



【例】后缀数组

Hash

河南省实验中学
信息技术组

Hash

字符串 Hash

例题

雪花雪花

兔子与兔子

字符串模式匹配

最长回文子串

后缀数组

其他案例

小结与练习

小结

练习

```
1 // 比较  $s[x \sim n-1]$  和  $s[y \sim n-1]$  的大小
2 // 传入 sort 函数即可实现排序
3 bool cmp(int x, int y)
4 {
5     int l = 0, r = n - max(x, y);    // 公共前缀上下界
6     while(l < r)
7     {
8         int m = (l + r + 1) >> 1;
9         int r1 = x + m - 1, r2 = y + m - 1;
10        unsigned long long hs1 = f[x] - f[r1 + 1] * p[m];
11        unsigned long long hs2 = f[y] - f[r2 + 1] * p[m];
12        if(hs1 == hs2) l = m;
13        else r = m - 1;
14    }
15    return s[x + 1] < s[y + 1];
16 }
17 // 求 height 数组的程序类似, 请读者自行完成
```



其他案例

Hash

河南省实验中学
信息技术组

Hash

字符串 Hash

例题

雪花雪花

兔子与兔子

字符串模式匹配

最长回文子串

后缀数组

其他案例

小结与练习

小结

练习

- 给定串 s 和 t ，求 s 的每个后缀子串与 t 的最长公共前缀。
 - 扩展 KMP 算法是标准解法，时间复杂度 $O(N)$ 。
 - Hash+ 二分也可以解决，时间复杂度 $O(N \log N)$ 。



小结

Hash

河南省实验中学
信息技术组

Hash

字符串 Hash

例题

雪花雪花
兔子与兔子
字符串模式匹配
最长回文子串
后缀数组
其他案例

小结与练习

小结
练习

- Hash 方法提供了一种快速判定复杂信息是否相同的方法，经典应用就是判定字符串是否相同。
- 注意：两个信息的 Hash 值不同，它们一定不同；两个信息的 Hash 值相同，它们不一定相同。
- 对于处理字符串子串类问题，可以通过 $O(N)$ 的预处理得出字符串前缀 Hash 或后缀 Hash，从而在 $O(1)$ 的时间复杂度内比较任意两个子串。
- 字符串 Hash 可以等价或近似解决的字符串经典算法⁵，对于初学者建议先学习字符串 Hash，在水平提高后学习字符串经典算法。

⁵KMP、扩展 KMP、后缀数组、Manacher 算法等。



Hash 冲突问题

Hash

河南省实验中学
信息技术组

Hash

字符串 Hash

例题

雪花雪花
兔子与兔子
字符串模式匹配
最长回文子串
后缀数组
其他案例

小结与练习

小结
练习

- 哈希错误率问题：哈希并不是万无一失的。将复杂信息压缩成一个整数，必然损失了信息，也就有概率出现误判的情况。
- 如果使用 10^9 左右的单模数做哈希，那么值域太小导致冲突率太高 (答案错误的概率可能到千分之一以上)，很有可能导致丢分。
- 如果使用自然溢出做哈希，很有可能被经验老道的出题人提前猜到，构造数据卡掉。
- 如果使用双模数做哈希，安全性上基本有保证，但是代码常数又太大。
- 没有一种完美的方法，需要看情况如何取舍。



卡大模数 Hash

Hash

河南省实验中学
信息技术组

Hash

字符串 Hash

例题

雪花雪花

兔子与兔子

字符串模式匹配

最长回文子串

后缀数组

其他案例

小结与练习

小结

练习

- 设 Hash 的取值空间 (所有可能出现的元素) 为 d , 要计算 Hash 值的元素个数为 n , 那么 Hash 冲突的概率为:

$$p(n, d) = \frac{n!}{d^n(d-n)!} \approx 1 - e^{-\frac{n(n-1)}{2d}}$$

- 为了卡掉 Hash, 需要满足以下条件:
 - d 一定要大于模数。
 - $1 - p(n, d)$ 要尽量小。

例如: 若字符集为大小写字母和数字, 模数为 $10^9 + 7$ 时:

- $\log_{62}(10^9 + 7) \approx 6$
- $p(10^6, 62^6) \approx 0.9$

所以如果随机生成 10^6 个长度为 6 的字符串, Hash 冲突的概率高达 90%。



卡自然溢出 Hash

Hash

河南省实验中学
信息技术组

Hash

字符串 Hash

例题

雪花雪花

兔子与兔子

字符串模式匹配

最长回文子串

后缀数组

其他案例

小结与练习

小结

练习

- 假设 Hash 函数是形如 $f(s) = \sum_{i=1}^l s_i \times b^{l-i} \pmod{M}$, 其中 $M = 2^{64}$ 。
- 如果 b 为偶数, 容易发现, 当 $l \geq 64$ 时, $s_i \times b^l \equiv 0 \pmod{M}$, 所以只需要构造形如 $aaa\dots a$ 、 $baa\dots a$ 且长度大于 64 的字符串就能冲突。
- 如果 b 为奇数, 定义 $!s$ 表示把 s 中的字符反转, 例如 $s = abaab, !s = babba$, 即将 a 变成 b , 将 b 变成 a 。此时, 不断构造 $s_i = s_{i-1} + !s_{i-1}$, 那么构造 12 次后的 s_{12} 和 $!s_{12}$ 就会导致 Hash 冲突。



参考文章

Hash

河南省实验中学
信息技术组

Hash

字符串 Hash

例题

雪花雪花

兔子与兔子

字符串模式匹配

最长回文子串

后缀数组

其他案例

小结与练习

小结

练习

- OI wiki 字符串 Hash
- 关于滚动 Hash 和反 Hash 测试背后的数学原理



练习

Hash

河南省实验中学
信息技术组

Hash

字符串 Hash

例题

雪花雪花

兔子与兔子

字符串模式匹配

最长回文子串

后缀数组

其他案例

小结与练习

小结

练习

- 雪花雪花 (COGS 3459)
- 兔子与兔子 (COGS 3151)
- 最长回文子串 (COGS 3473)
- 后缀数组 (COGS 3152)
- 矩阵 (COGS 2366)
- 树形地铁系统 (COGS 2368)
- 匹配统计 (COGS 3475)
- 最长双回文串 (洛谷 P4555)
- 字符串匹配 [NOIP 2020](COGS 3509)