

ACM 模板

PKU cstdio,hzwer,Kuribohg

2016 年 8 月 29 日

目录

1	数学	3
1.1	线性筛	3
1.2	lucas	3
1.3	高斯消元	3
1.4	BSGS	3
1.5	FFT	3
1.6	素数测试	3
2	数据结构	3
2.1	splay	3
2.2	treap	3
2.3	主席树	3
2.4	可并堆	3
2.5	KDtree	3
2.6	可持久化线段树	3
2.7	可持久化 trie	3
3	字符串	3
3.1	kmp	3
3.2	manacher	3
3.3	后缀数组	3
3.4	后缀自动机	5
3.5	回文自动机	5
4	图论	5
4.1	最短路	5
4.2	最小生成树	5
4.3	网络流	5
4.4	tarjan	5
4.5	虚树	5
4.6	生成树计数	5
4.7	KM 算法	5

目录	2
4.8 ICT	5
4.9 2-SAT	5
4.10 带花树	5
4.11 曼哈顿最小生成树	5
5 计算几何	5
5.1 凸包	5
5.2 旋转卡壳	5
5.3 半平面交	5
5.4 辛普森积分	5
6 其它	5
6.1 高精度	5

1 数学

- 1.1 线性筛
- 1.2 lucas
- 1.3 高斯消元
- 1.4 BSGS
- 1.5 FFT
- 1.6 素数测试

2 数据结构

- 2.1 splay
- 2.2 treap
- 2.3 主席树
- 2.4 可并堆
- 2.5 KDtree
- 2.6 可持久化线段树
- 2.7 可持久化 trie

3 字符串

- 3.1 kmp
- 3.2 manachar
- 3.3 后缀数组

后缀排序，输出 SA 数组以及 height 数组

```
1 #include<cstdio>
2 #include<cstring>
3 #include<iostream>
4 #define ll long long
5 using namespace std;
6 int n,p,q=1,k;
7 char ch[100005];
8 int sa[2][100005],rk[2][100005];
9 int a[100005],h[100005],v[100005];
10 void calsa(int *sa,int *rk,int *SA,int *RK)
11 {
12     for(int i=1;i<=n;i++)v[rk[sa[i]]]=i;
```

```

13     for (int i=n; i; i--)
14         if (sa[i]>k)
15             SA[v[rk[sa[i]-k]]--]=sa[i]-k;
16     for (int i=n-k+1; i<=n; i++)
17         SA[v[rk[i]]--]=i;
18     for (int i=1; i<=n; i++)
19         RK[SA[i]]=RK[SA[i-1]]+(rk[SA[i-1]]!=rk[SA[i]]||rk[SA[i-1]+k]!=rk[SA[i]+k]);
20 }
21 void getsa()
22 {
23     for (int i=1; i<=n; i++)v[a[i]]++;
24     for (int i=1; i<=30; i++)v[i]+=v[i-1];
25     for (int i=1; i<=n; i++)
26         sa[p][v[a[i]]--]=i;
27     for (int i=1; i<=n; i++)
28         rk[p][sa[p][i]]=rk[p][sa[p][i-1]]+(a[sa[p][i]]!=a[sa[p][i-1]]);
29     for (k=1; k<n; k<=<=1, swap(p, q))
30         calsa(sa[p], rk[p], sa[q], rk[q]);
31     for (int i=1, k=0; i<=n; i++)
32     {
33         int j=sa[p][rk[p][i]-1];
34         while (a[i+k]==a[j+k])k++;
35         h[rk[p][i]]=k; if (k)k--;
36     }
37 }
38 int main()
39 {
40     scanf("%s", ch+1);
41     n=strlen(ch+1);
42     for (int i=1; i<=n; i++)a[i]=ch[i]-'a'+1;
43     getsa();
44     for (int i=1; i<=n; i++)
45         printf("%d_", sa[p][i]);
46     cout<<endl;
47     for (int i=2; i<=n; i++)
48         printf("%d_", h[i]);
49     return 0;
50 }

```

3.4 后缀自动机

3.5 回文自动机

4 图论

4.1 最短路

4.2 最小生成树

4.3 网络流

4.4 tarjan

4.5 虚树

4.6 生成树计数

4.7 KM 算法

4.8 ICT

4.9 2-SAT

4.10 带花树

4.11 曼哈顿最小生成树

5 计算几何

5.1 凸包

5.2 旋转卡壳

5.3 半平面交

5.4 辛普森积分

6 其它

6.1 高精度