



NANJING UNIVERSITY

ACM-ICPC Codebook 3
Data Structures

May 25, 2017

Contents

1	Miscellaneous Data Structures	4
1.1	Union-find set	4

1 Miscellaneous Data Structures

1.1 Union-find set

Data structure for disjoint sets.

Usage:

<code>init(n)</code>	Initialize the sets from 0 to n , each includes one element.
<code>find(x)</code>	Return the representative of the set containing x .
<code>unite(u, v)</code>	Unite the two sets containing u and v . Return false if u and v are already in the same set; otherwise true.

Time complexity: $O(n)$ for initialization; $O(\log n)$ for find and union.

```
1 struct ufs{
2     vector<int> p;
3
4     void init(int n){
5         p.resize(n + 1);
6         for (int i=0; i<n; i++) p[i] = i;
7     }
8
9     int find(int x){
10         if (p[x] == x) return x;
11         return p[x] = find(p[x]);
12     }
13
14     bool unite(int u, int v){
15         u = find(u); v = find(v);
16         p[u] = v;
17         return u != v;
18     }
19 };
```