## 概要

并查集可以理解为对森林的操作，怎么表示森林呢，如果节点个数不会增删，那么将其中的树看做有根数，使用一个数组就能很方便的表示，因为对于有根数，每个节点虽然可能有多个孩子节点，但只有一个父节点，于是x[i]就代表节点i的父节点是x[i]（若x[i]==i，表示没有父节点）。每棵树代表一个集合，有一个代表元，代表元起中介作用，元素之间的关系可以通过它们分别和各自代表元的关系，以及它们代表元之间的关系来判断。

并查集主要有两种操作，find和union。

find操作用来寻找一个节点所在树的代表元，即寻找父节点，再寻找父节点的父节点，如此重复直到找到代表元（代表元的父节点是自己），一般来说树的结构并不重要，所以为了加快之后的查询速度在查询过程中可以进行路径压缩（一般使用递归实现），即将查询路径中遇到的节点的父节点都改成代表元，这样会改变树的结构。路径过长可能会爆栈，使用while循环就没问题，第一次沿着路径寻找到代表元，第二次再沿着路径将沿途节点的父节点改成代表元。

union操作将两棵树合为一棵树，只需将一棵树的代表元的父节点改成另一棵树的代表元即可。

最后，所有x[i]==i的节点即代表元，可以用来缩点。

## 种类并查集

种类并查集不仅可以用来判断两个元素是否属于一个集合，还可以判断它们是否属于一个种类以及种间关系。

需要另开一个数组，表示一个节点和其父节点的关系。关系可以自己定义值

来表示，是有方向的，比如节点a到节点b是x关系，节点b到节点c是y关系，则节点a到节点c的关系可以通过x到y的关系来确定，通用方法是开一个二维数组表示关系间的变换（关系数少时可以通过求和和取余来实现，如0、奇数、偶数各代表一种关系，求和后取余3）。所以沿着路径不断变换关系，就能得到节点与代表元的关系。

find操作路径压缩过程除了将每个节点父节点改成代表元外，还需将节点对父节点的关系更新成对代表元的关系。

union操作除了将一棵树的代表元的父节点改成另一个代表元外，也需更新它对父节点的关系。

## 带权并查集

……也许是更加通用的吧，种类并查集中另外记录的节点与父节点的关系看作权值。

## 求连通分支数

无向图，N个点（1-N编号），M条边，判断还需多少条边让图中任意2点连通。

就是求图的连通分支数嘛，减一即为所求。使用并查集表示出最终的图，统计代表元个数即为连通分支数。

## 并查集判断环

有许多由2种元素组成的简单化合物，依次装车，当车上存在k种化合物，正好包含k种元素时，爆炸。

问为保证不爆炸留下多少种化合物

元素看做点，化合物看做边，则图存在环表示爆炸，使用并查集，发生合并冲突（即已经合并，本来就属于同一个集合）表示有环。

## 判断城市 种类并查集

1-N个罪犯，有两座城市。

A a b 问a和b是否属于同一座城市。回答是不是或者无法判断。

D a b 表明a和b不属于同一座城市 。

新增y[i]表示i节点和父节点关系，奇数表示不属于同一个城市。每次询问时，若节点不属于同一个集合，无法判断，否则以通过代表元为中介求得二者关系，各自与代表元关系可以在find操作中顺便实现。

## 食物链 种类并查集

三类动物A,B,C，A吃B， B吃C，C吃A，现有N种动物（编号1到N），有许多句话顺序说出：

"1 X Y"，表示X和Y是同类。

"2 X Y"，表示X吃Y。

当X或Y大于N，或者出现自己吃自己，或者与前面话矛盾，则为假话，问有几句假话。

节点与父节点关系有同类、吃、被吃三种，分别用0、1、2表示，正好可以通过相加再取余3来实现关系递推。每次先判断关系是否已经存在，若存在判断是否与所说一致，否则更新关系。

## 离散化 种类并查集

长达10亿的01串，顺序说第i到第j的子串中有偶数个还是奇数个1，

问最多多少句没有矛盾。

前缀和思想，即i-1之前的串，与j之前的串同奇偶或异奇偶，转换为是否为同意种类，偶数是，奇数不是。

因为10亿很多，不可能开数组来索引，所以用map来将大数依次映射到0、1、2……

## 求到根节点距离 带权并查集

一共n个节点，编号1到n。

E a表示查找a到根节点距离。

I a b表示将b作为a的根（保证a一开始无根），且新边长为abs(a-b)%1000。

新增数组dis，dis[i]表示i节点到其根节点（代表元）的距离，在find和union操作中更新dis[i]即可。

## 并查集求每个集合元素个数

再开一个num数组作为代表元所在集合元素个数（代表元的才有效，其他不管），起初为各元素为1，合并时将一个代表元的加到另一个代表元上，最后通过find(x)是否等于x找出所以代表元，作为num下标即可求解。