如果数据很多，依次查找会很慢，因此可以使用二分法，将时间复杂度由O(n)降低到O(logn)

1. 数据结构
   1. 数组
2. 算法

long long binarySearch(int [] n, long long left, long long right,int tar) {

long long mid;

while(left <= right) {

mid = (left + right) / 2;

if(n[mid]>tar) right=mid-1;

if(n[mid]<tar) left=mid+1;

if(n[mid]=tar) return mid;

}

return -1;

}

Algorithm中有现成的函数

二分查找

int myints[] = {1,2,3,4,5,4,3,2,1};

vector<int> v(myints,myints+9);

sort(v.begin(),v.end());

binary\_search (v.begin(), v.end(), 3)；返回找没找到bool型