程序设计实习

算法基础

张勤健
zqj@pku.edu.cn

北京大学信息科学技术学院

2024 年 5 月 8 日

二分查找

A 心里想一个 1-1000 之间的数，B 来猜，可以问问题，A 只能回答是或否。怎么猜才能问的问题次数最少?

二分查找

A 心里想一个 1-1000 之间的数，B 来猜，可以问问题，A 只能回答是或否。怎么猜才能问的问题次数最少?

是 1 吗？是 2 吗？……. 是 999 吗？平均要问 500 次

二分查找

A 心里想一个 1-1000 之间的数，B 来猜，可以问问题，A 只能回答是或否。怎么猜才能问的问题次数最少?

是 1 吗？是 2 吗？……. 是 999 吗？平均要问 500 次

大于 500 吗？大于 750 吗？大于 625 吗？…… 每次缩小猜测范围到上次的一半，只需要 10 次

二分查找函数

写一个函数 BinarySeach，在包含 size 个元素的、从小到大排序的 int 数组 a 里查找元素 p，如果找到，则返回元素下标，如果找不到，则返回-1。要求复杂度 $O(\log n)$

```
int BinarySearch(int a[],int size,int p) {
  int L = 0; //查找区间的左端点
  int R = size - 1; //查找区间的右端点
  while (L <= R) { //如果查找区间不为空就继续查找
    int mid = L + (R - L) / 2; //取查找区间正中元素的下标
    if (p == a[mid] ) return mid;
    else if (p > a[mid]) L = mid + 1; //设置新的查找区间的左端点
    else R = mid - 1; //设置新的查找区间的右端点
  }
  return -1;
} //复杂度 O(log(n))
```

二分查找函数

写一个函数 LowerBound，在包含 size 个元素的、从小到大排序的 int 数组 a 里查找比给定整数 p 小的，下标最大的元素。找到则返回其下标，找不到则返回-1

```
int LowerBound(int a[],int size,int p) {//复杂度 O(log(n))
  int L = 0; //查找区间的左端点
  int R = size - 1; //查找区间的右端点
  int lastPos = -1; //到目前为止找到的最优解
  while (L <= R) { //如果查找区间不为空就继续查找
    int mid = L + (R - L) / 2; //取查找区间正中元素的下标
    if (a[mid] >= p) {
      R = mid - 1;
    } else {
      lastPos = mid;
      L = mid + 1;
    }
  }
  return lastPos;
}
```

二分查找函数

注意:

```
int mid = (L + R) / 2; //取查找区间正中元素的下标
```

为了防止 (L+R) 过大溢出:

```
int mid = L + (R - L) / 2;
```

二分法求方程的根

求下面方程的一个根：$f(x) = x^3 - 5x^2 + 10x - 80 = 0$
若求出的根是 a，则要求 $|f(a)| \leq 10^{-6}$

二分法求方程的根

求下面方程的一个根：$f(x) = x^3 - 5x^2 + 10x - 80 = 0$
若求出的根是 a，则要求 $|f(a)| \le 10^{-6}$ 解法：
对 $f(x)$ 求导，得 $f'(x) = 3x^2 - 10x + 10 = 3(x - \frac{5}{3})^2 + \frac{5}{3} > 0$。
故 $f(x)$ 是单调递增的。
易知 $f(0) < 0$ 且 $f(100) > 0$, 所以区间 $[0, 100]$ 内必然有且只有一个根。
由于 $f(x)$ 在 $[0, 100]$ 内是单调的，所以可以用二分的办法在区间 $[0, 100]$ 中寻找根。

二分法求方程的根

```cpp
#include <cstdio>
#include <iostream>
#include <cmath>
using namespace std;
double EPS = 1e-6;
double f(double x) {
  return  x*x*x - 5*x*x + 10*x - 80;
}
int main()  {
  double root, x1 = 0, x2 = 100, y;
  root = x1 + (x2 - x1) / 2;
  int triedTimes = 1; //记录一共尝试多少次，对求根来说不是必须的
  y = f(root);
  while (fabs(y) > EPS) {
    if (y > 0) x2 = root;
    else x1 = root;
    root = x1 + (x2 - x1) / 2;
    y = f(root);
    triedTimes ++;
  }
  printf("%.8f\n", root);
  printf("%d", triedTimes);
  return 0;
}
```

例题寻找指定和的整数对

输入 n ($n \leq 100000$) 个整数，找出其中的两个数，它们之和等于整数 m(假定肯定有解)。题中所有整数都能用 int 表示

例题寻找指定和的整数对

输入 n ($n \leq 100000$) 个整数，找出其中的两个数，它们之和等于整数 m(假定肯定有解)。题中所有整数都能用 int 表示

解法 1：用两重循环，枚举所有的取数方法，复杂度是 $O(n^2)$ 的。

```
for (int i = 0;i < n - 1; ++i) {
  for (int j = i + 1; j < n; ++j) {
    if (a[i] + a[j] == m) break;
  }
}
```

$100000^2 = 10^{10}$, 超时

例题寻找指定和的整数对

输入 n ($n \leq 100000$) 个整数，找出其中的两个数，它们之和等于整数 m(假定肯定有解)。题中所有整数都能用 int 表示

解法 2：

1. 将数组排序，复杂度是 $O(n\log n)$
2. 对数组中的每个元素 $a[i]$, 在数组中二分查找 $m-a[i]$，看能否找到。复杂度 $\log n$，最坏要查找 $n-2$ 次，所以查找这部分的复杂度也是 $O(n\log n)$

这种解法总的复杂度是 $O(n\log n)$ 的。

例题寻找指定和的整数对

输入 n ($n \leq 100000$) 个整数，找出其中的两个数，它们之和等于整数 m(假定肯定有解)。题中所有整数都能用 int 表示

解法 3：

1. 将数组排序，复杂度是 $O(n \log n)$
2. 查找的时候，设置两个变量 i 和 j,i 初值是 0,j 初值是 $n-1$. 看 $a[i]+a[j]$, 如果大于 m，就让 j 减 1，如果小于 m, 就让 i 加 1，直至 $a[i]+a[j]=m$。

这种解法总的复杂度是 $O(n \log n)$ 的。

例题 Aggressive cows

农夫 John 建造了一座很长的畜栏，它包括 N $(2 \le N \le 100000)$ 个隔间，这些小隔间的位置为 $x_0, \ldots, x_{N-1}$ $(0 \le x_i \le 10^9$, 均为整数, 各不相同). John 的 C $(2 \le C \le N)$ 头牛每头分到一个隔间。牛都希望互相离得远点省得互相打扰。怎样才能使任意两头牛之间的最小距离尽可能的大，这个最大的最小距离是多少呢?

例题 Aggressive cows

解法 1：

先得到排序后的隔间坐标 $x_0, \ldots, x_{N-1}$

从 $10^9/(C-1)$ 到 1 依次尝试这个“最大的最近距离”D，找到的第一个可行的就是答案。

尝试方法：

1. 第 1 头牛放在 x_0
2. 若第 k 头牛放在 x_i ，则找到 x_{i+1} 到 x_{N-1} 中第一个位于 $[x_i+D, 10^9]$ 中的 x_j，第 $k+1$ 头牛放在 x_j。找不到这样的 x_j，则 $D=D-1$，转 1) 再试

若所有牛都能放下，则 D 即答案

例题 Aggressive cows

解法 1：

先得到排序后的隔间坐标 $x_0, \ldots, x_{N-1}$

从 $10^9/(C-1)$ 到 1 依次尝试这个“最大的最近距离”D，找到的第一个可行的就是答案。

尝试方法：

1. 第 1 头牛放在 x_0
2. 若第 k 头牛放在 x_i ，则找到 x_{i+1} 到 x_{N-1} 中第一个位于 $[x_i + D, 10^9]$ 中的 x_j, 第 $k+1$ 头牛放在 x_j。找不到这样的 x_j, 则 $D = D - 1$, 转 1) 再试

若所有牛都能放下，则 D 即答案
复杂度 $\frac{10^9}{C-1} * N$，即不小于 10^9, 超时!

解法 2：

先得到排序后的隔间坐标 $x_0, \dots, x_{N-1}$

在 $[L, R]$ 内用二分法尝试“最大最近距离”$D = (L + R)/2$ (L,R 初值为 $[1, \frac{10^9}{C-1}]$

若 D 可行，则记住该 D，然后在新 $[D + 1, R]$ 中继续尝试
若 D 不可行，则在新 $[L, D - 1]$ 中继续尝试

例题 Aggressive cows

解法 2：

先得到排序后的隔间坐标 $x_0, \ldots, x_{N-1}$

在 $[L, R]$ 内用二分法尝试“最大最近距离”$D = (L + R)/2$ (L,R 初值为 $[1, \frac{10^9}{C-1}]$

若 D 可行，则记住该 D，然后在新 $[D + 1, R]$ 中继续尝试
若 D 不可行，则在新 $[L, D - 1]$ 中继续尝试

复杂度 $\log \frac{10^9}{C-1} * N$