

1238：一元三次方程求解

题目描述

形如： $ax^3+bx^2+cx+d=0$ ，这样的一个一元三次方程。

给出该方程中各项的系数(a, b, c, d均为实数)，并约定该方程存在三个不同实根(根的范围在-100至100之间)，且根与根之差的绝对值 ≥ 1 。要求由小到大依次在同一行输出这三个实根(根与根之间留有空格)，并精确到小数点后2位。

输入

一行，包含四个实数a, b, c, d，相邻两个数之间用单个空格隔开。

输出

一行，包含三个实数，为该方程的三个实根，按从小到大顺序排列，相邻两个数之间用单个空格隔开，精确到小数点后2位。

输入样例

```
1.0 -5.0 -4.0 20.0
```

输出样例

```
-2.00 2.00 5.00
```

解析

求解方程我们一般采用**零点存在性定理**。该定理描述如下：

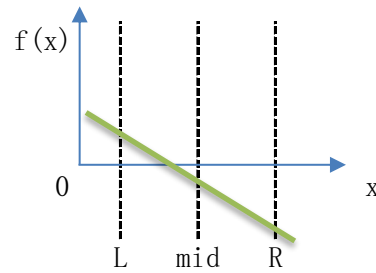
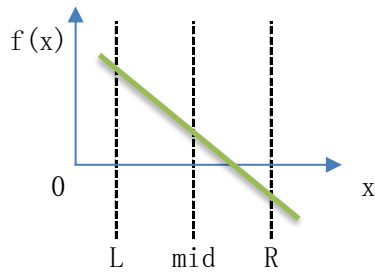
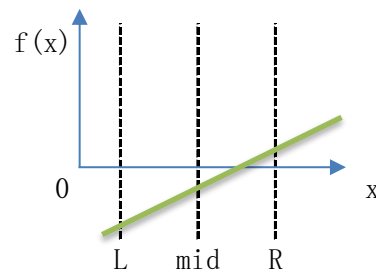
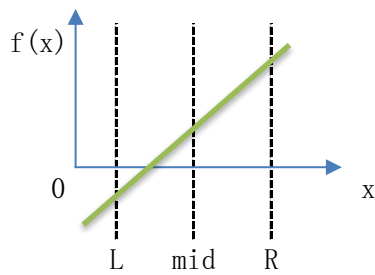
对连续函数 $f(x)$ ，若有 $f(a)f(b)<0$ ($a<b$)，则 $f(x)$ 在区间 (a,b) 上至少存在一个解。这样就可以判断一个区间中是否存在解。

令“条件”为 $f(x)>=0$ ，显然在上述区间 (a,b) 上“条件具有单调性”：在根的一侧 $f(x)$ 都是负数，另一侧 $f(x)$ 都是正数。题目里说明了任意两根之差不小于1，那么可以把 $[-100,100]$ 等分成若干小段 $[i,i+1)$ (这里左闭右开是为了防止端点处是零点导致得到重复解)。在每个小段中至多只有一个零点，这意味着这个区间上的“条件”具有单调性。

于是一个定义在实数区间上的二分呼之欲出：如果重点的函数值和某端点的正负性相同，那么零点一定在中点的另一侧。注意函数的单调性和“条件”的单调性无关。

另外需要注意的一点：实数之间不能直接比较是否相等，而是判断之间的差值是否小于 ϵ 。

题目中要求保留两位小数，因此我们可以把精度定义为 10^{-3}



编码

```
#include <bits/stdc++.h>

using namespace std;

double a, b, c, d;

const double eps = 1e-3; //定义我们计算的精度

//定义高阶函数
double f(double x) {
    return a * x * x * x + b * x * x + c * x + d; //求函数值
}

//执行二分搜索
double BinarySearch(double l, double r) {
    double mid;
    //通过与eps作比较来判断是否找到根
    while (r - l > eps) {
        //计算中值
        mid = l + (r - l) / 2;
        if (fabs(f(mid)) < eps) {
            return mid;
        }
        //r和mid的正负性相同，那么零点在左侧
        else if (f(mid) * f(r) > 0) {
            r = mid;
        }
        //l和mid的正负性相同，那么零点在右侧
        else if (f(mid) * f(l) > 0) {
            l = mid;
        }
    }
}
```

```

    }
    return l;
}

int main(int argc, char *argv[]) {
    cin >> a >> b >> c >> d;
    double l, r;
    for (double i = -100; i <= 100; i++) {
        l = i;
        r = i + 1;
        //如果l是根, 直接输出
        if (fabs(f(l)) < eps) {
            printf("%.2lf ", l);
        }
        //如果r是根, 跳过, l最终也会走到r的位置
        else if (fabs(f(r)) < eps) {
            continue;
        }
        //在l,r上有根, 开始进行二分搜索
        else if (f(l) * f(r) < 0) {
            double res = BinarySearch(l, r);
            printf("%.2lf ", res);
        }
    }
    return 0;
}

```

逻辑航线培优教育，信息学奥赛培训专家。

扫码添加作者获取更多内容。

