

1241：二分法求函数的零点

题目描述

有函数： $f(x)=x^5-15x^4+85x^3-225x^2+274x-121$

已知 $f(1.5)>0, f(2.4)<0$ 且方程 $f(x)=0$ 在区间 $[1.5, 2.4]$ 有且只有一个根，请用二分法求出该根。

输入

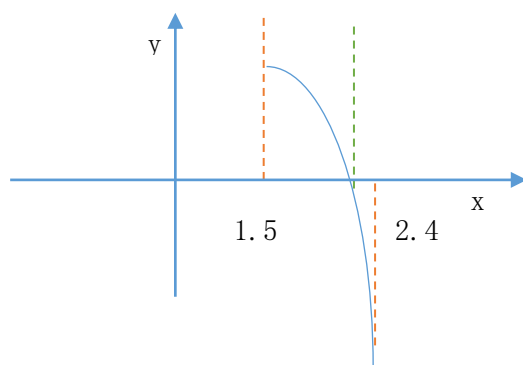
无

输出

该方程在区间 $[1.5, 2.4]$ 中的根。要求四舍五入到小数点后6位。

分析

首先，这道题目是一个高阶函数。因为题目中给出了 $f(1.5)>0, f(2.4)<0$ ，所以我们可以很轻松的画出函数图像，如下图所示：



其中，绿色的虚线和x轴的交点，便是我们所要求的值。

现在，我们可以很清晰的看出，本题就是一道以1.5和2.4为左右边界，求二者之间的一个值x，使其能够满足 $f(x)=0$ 。

函数的二分模型：

如果 $f(x1)*f(mid)<0$ ，则代表解在左边，则右边界左移，即 $mid=x2$ 。

如果 $f(x2)*f(mid)<0$ ，则代表解在右边，则左边界右移，即 $mid=x1$ 。

循环的终点即为 $x1-x2$ 的绝对值小于所需要的精度，为了保证更加安全，我们可以将精度增加一位，例如题目要求精度是6位小数，那么我们则需要保证 $x1-x2$ 的绝对值小于 $1e-7$ 即可。

秦九韶算法

秦九韶算法是中国南宋时期的数学家秦九韶提出的一种多项式简化算法。在西方被称作霍纳算法。

秦九韶算法是一种将一元 n 次多项式的求值问题转化为 n 个一次式的算法。其大大简化了计算过程，即使在现代，利用计算机解决多项式的求值问题时，秦九韶算法依然是最优的算法。

一般地，一元 n 次多项式的求值需要经过 $[n(n+1)]/2$ 次乘法和 n 次加法，而秦九韶算法只需要 n 次乘法和 n 次加法。在人工计算时，大大简化了运算过程。

举例：

$$\begin{aligned} & 4x^5 + 2x^4 + 3.5x^3 - 2.6x^2 + 1.7x - 0.8 \\ = & x(4x^4 + 2x^3 + 3.5x^2 - 2.6x + 1.7) - 0.8 \\ = & x(x(4x^3) + 2x^2 + 3.5x - 2.6) + 1.7) - 0.8 \\ = & x(x(x(4x^2 + 2x + 3.5) - 2.6) + 1.7) - 0.8 \\ = & x(x(x(x(4x + 2) + 3.5) - 2.6) + 1.7) - 0.8 \end{aligned}$$

编码

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

using namespace std;

double a, b, c, d;

const double eps = 1e-7; //定义我们计算的精度

double f(double x) {
    return x * (x * (x * (x * (x - 15) + 85) - 225) + 274) - 121;
}

//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[]) {
    printf("%.6lf", BinarySearch(1.5, 2.4));
    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。

