# 实习五实习报告

**计21 战裕隆 2012011388**

## 实习要求

**分别使用二分法和牛顿法求解非线性方程2\*x\*x\*x-x\*x+3\*x-1=0.**

**使用迭代法求解2\*x\*x\*x-x-1=0，对于迭代法使用的方程进行比较效果。**

**使用牛顿法求解方程，比较初值对于解的影响。**

## 算法描述

1. **二分法：（1）、选定初始点**

**（2）、取中点，判断之后进行循环**

**（3）、当精度满足后输出结果**

1. **牛顿法：（1）、选定初始点**

**（2）、迭代计算**

**（3）、结果满足要求后终止算法**

1. **迭代法：（1）、选择函数**

**（2）、进行迭代计算**

## 程序清单

**1、#include<iostream>**

**#include<cmath>**

**using namespace std;**

**double jingdu=1e-6;**

**double hanshu(double x)**

**{**

**return 2\*x\*x\*x-x\*x+3\*x-1;**

**}**

**double daoshu(double x)**

**{**

**return 6\*x\*x-2\*x+3;**

**}**

**double x0=0.5;**

**double newton()**

**{**

**double x1=x0,x2;**

**bool panduan;**

**int times=0;**

**panduan=(hanshu(x1)>jingdu);**

**while(panduan)**

**{**

**cout<<times<<" : "<<x1<<" "<<x2<<endl;**

**x2=x1-hanshu(x1)/daoshu(x1);**

**times++;**

**panduan=(abs(hanshu(x2))>jingdu);**

**panduan=panduan && (abs(x2-x1)>jingdu);**

**x1=x2;**

**}**

**cout<<times<<endl;**

**return x2;**

**}**

**int main()**

**{**

**double jieguo=newton();**

**cout<<jieguo<<endl;**

**system("pause");**

**return 0;**

**}**

**2、#include<iostream>**

**#include<cmath>**

**#include<cstdlib>**

**using namespace std;**

**double x0=0.0;**

**double hanshu1(double x)**

**{**

**return pow((x+1.0)/2,1/3);**

**}**

**double hanshu2(double x)**

**{**

**return 2\*x\*x\*x-1.0;**

**}**

**double diedai1()**

**{**

**double x=x0;**

**for(int i=1;i<=10;i++)**

**{**

**x=hanshu1(x);**

**}**

**cout<<x<<endl;**

**}**

**double diedai2()**

**{**

**double x=x0;**

**for(int i=1;i<=10;i++)**

**{**

**double x1=hanshu2(x);**

**x=x1;**

**}**

**cout<<x<<endl;**

**}**

**int main()**

**{**

**diedai1();**

**diedai2();**

**system("pause");**

**return 0;**

**}**

**3、#include<iostream>**

**#include<cmath>**

**#include<cstdlib>**

**using namespace std;**

**double x1=-3.0,x2=3.0;**

**double jingdu=1e-6;**

**double hanshu(double x)**

**{**

**return 2\*x\*x\*x-x\*x+3\*x-1;**

**}**

**double erfen()**

**{**

**int times=0;**

**double x3=(x1+x2)/2.0;**

**double y1,y2,y3;**

**while(abs(hanshu(x3))>jingdu)**

**{**

**times++;**

**y1=hanshu(x1);**

**y2=hanshu(x2);**

**x3=(x1+x2)/2.0;**

**y3=hanshu(x3);**

**cout<<times<<": "<<x1<<" "<<x2<<endl;**

**if(y1\*y3<0)**

**{**

**x1=x1;**

**x2=x3;**

**continue;**

**}**

**else if(y2\*y3<0)**

**{**

**x1=x3;**

**x2=x2;**

**continue;**

**}**

**}**

**cout<<x3<<" "<<times<<endl;**

**}**

**int main()**

**{**

**erfen();**

**system("pause");**

**return 0;**

**}**

**4、#include<iostream>**

**#include<cmath>**

**using namespace std;**

**double jingdu=1e-6;**

**double hanshu(double x)**

**{**

**return x\*x\*x-x-1;**

**}**

**double daoshu(double x)**

**{**

**return 3\*x\*x-1;**

**}**

**double x0=1.5;**

**double newton()**

**{**

**double x1=x0,x2;**

**bool panduan;**

**int times=0;**

**panduan=(hanshu(x1)>jingdu);**

**while(panduan)**

**{**

**cout<<times<<" : "<<x1<<" "<<x2<<endl;**

**x2=x1-hanshu(x1)/daoshu(x1);**

**times++;**

**panduan=(abs(hanshu(x2))>jingdu);**

**panduan=panduan && (abs(x2-x1)>jingdu);**

**x1=x2;**

**}**

**cout<<times<<endl;**

**return x2;**

**}**

**int main()**

**{**

**double jieguo=newton();**

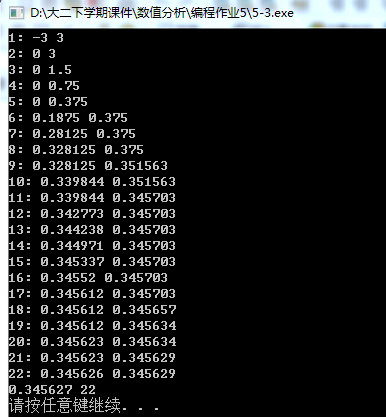
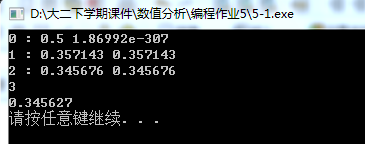
**cout<<jieguo<<endl;**

**system("pause");**

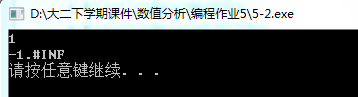
**return 0;**

**}**

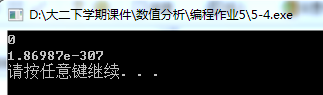
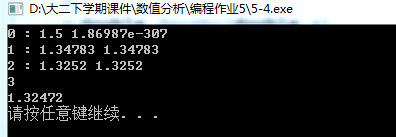
## 体会与问题

**1、**

**如图分别是使用牛顿法和二分法求解非线性方程的各次迭代结果，发现二分法的迭代次数明显较多，可见牛顿法是一个相对于二分法比较高效的算法。**

**2、**

**如图是使用迭代法，但使用了两个不相同的方程，第一个迭代方程收敛，可以得到一个正确的结果，而第二个迭代方程不收敛，最终发展到为负无穷大。**

**3、**

**可见初值对于牛顿法是否可以使用成功还是有很大影响的，这会得到一个相对于接近初值的解。**