**吉林大学数学实验中心实验报告**

**2021年 10月 14日**

|  |
| --- |
| **课程名称：科学计算方法实验实验 题目：求解多项式的根**  **姓名：田泽禹 年级专业：**  **指导教师：王双**  **算法描述：分别使用牛顿法与割线法求解多项式方程x^3-3x-1=0**  **相关内容如下**  代码1 (MATLAB, Newton)：  function [x0,step]=Newtonsolve(x0,epsi)  step=0;  delta=ffd(x0);  while (abs(delta)>epsi)  delta=(ffd(x0));  x0=x0-delta;  step=step+1;  end  end  function [delta]=ffd(x0)  f=x0^3-3\*x0-1;  fd=3\*x0^2-3;  if fd~=0  delta=f/fd;  else  disp('wrong');  end  end  运行结果1：    代码2 (MATLAB Secant)：  function [x,step]=secsolve(x,epsi)  step=0;  delta=(x(1)-x(2));  while (abs(delta)>epsi)  delta=(ffd(x));  x(2)=x(1);  x(1)=x(1)-delta;  step=step+1;  end  end  function [delta]=ffd(x)  f=x.^3-3.\*x-1;  delta = (x(1)-x(2))\*(f(1))/(f(1)-f(2));  end  运行结果2；    代码3 (C, Newton)  #include<stdio.h>  #include<math.h>  float ffd(float x);  int main()  {  int step=0;  float x;  float epsi;  printf("input x0 epsi\n");  scanf("%f%f",&x,&epsi);  float delta=1;  while (fabs(delta)>epsi)  {  delta=ffd(x);  x=x-delta;  step=step+1;  }  printf("the solution is %f \n",x);  printf("the steps used are %d \n",step);  return 0;  }  float ffd(float x)  {  float f=x\*x\*x-3\*x-1;  float fd=3\*x\*x-3;  return f/fd;  }  运行结果3    代码4 (C, secant)  #include<stdio.h>  #include<math.h>  float ffd(float x[]);  float func(float x);  int main()  {  int step=0;  float x[2];  float epsi;  printf("input x0 x1 epsi\n");  scanf("%f%f%f",&x[0],&x[1],&epsi);  float delta=x[0]-x[1];  while (fabs(delta)>epsi)  {  delta=ffd(x);  x[1]=x[0];  x[0]=x[0]-delta;  step=step+1;  }  printf("the solution is %f \n",x[0]);  printf("the steps used are %d \n",step);  return 0;  }  float ffd(float x[])  {  float y[2];  y[0]=func(x[0]);  y[1]=func(x[1]);  float delta = (x[0]-x[1])\*y[0]/(y[0]-y[1]);  return delta;  }  float func(float x)  {  return x\*x\*x-3\*x-1;  }  运行结果4.    **总结：**  牛顿法求解非齐次方程具有收敛速度快的特点，为2次收敛，但是由于计算了导数值，增加了计算量。而割线法用割线斜率近似导数值  在大型问题中免去了导数的繁琐计算，但其收敛性也稍差，在f性质足够好时其收敛性在1.618。同时根据初始两点的选择，割线法会出现收敛效果好、很差甚至不收敛的情况。 |