《数值分析》课程实验报告

实验名称 非线性方程求根

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **班级** | 信息1701 | **姓名** | 艾春辉 | **学号** | 201710000101 | **序号** |  |
| **教师** | 赵美玲 | **地点** | 数学实验中心 | | | **评分** |  |
| 1. 实验目的   ① 掌握二分法、牛顿迭代法等常用的非线性方程迭代算法；  ② 了解迭代算法的设计原理及初值对收敛性的影响。  二、实验过程和结果 1.二分法的算法描述 计算的根的二分法如下：  ① 输入有根区间，根的容许误差和的容许误差，置二分次数，并计算，如果 ，转②；否则，算法失败，结束；  ② 当时，计算  ，；  分情况处理：  若，则停止计算，输出近似根以及二分次数；  否则  若，则；  否则，；  ，转②  ③ ；  ④ 输出近似根以及二分次数。  二分法流程图：            若，根为端点，否则算法失败  否  是  ，      输出  是  否        是 是         2.牛顿迭代法的算法描述 给定初始值，为根的容许误差，为的容许误差，为最大迭代次数。置迭代次数，进行如下计算：  ① 如果或，则算法失败，结束；  否则执行②  ② 计算，；  ③ 若或，则输出近似根及迭代次数，程序结束；  否则执行④  ④ 令，转向①  牛顿迭代法流程图：      是  算法失败  否          输出  是  否     3. 牛顿迭代法的改进 1、单点弦截法  牛顿法的突出优点是收敛速度快，但它还有个明显的缺点：每一步迭代都要计算，增加了计算难度和计算量。为了避开导数的计算，可以考虑用差商替换，从而得到迭代公式    称为单点弦截法。  该迭代公式仅为**线性收敛**，故用该方法消除牛顿迭代中的导数计算，在收敛速度方面付出了很大代价。  2、两点弦截法（割线法）  改用差商代换牛顿法中的，可得迭代公式    称之为两点弦截法。  该方法是超线性收敛的，与单点弦截法相比有所改善。  综上所述，单点弦截法和两点弦截法都不需要计算导数，但是它都需要提供两个初值，而且收敛速度都不如牛顿迭代算法。  题目：求方程在1.5 附近的根.（误差限为）  （1）编程实现二分法，并求解上述非线性方程的根（有根区间自己确定）。  （2）编程实现牛顿法的计算程序，并且选择**不同**的初值（给出至少5个），观察初值对算法收敛性的影响，当算法收敛时，记录所需的迭代次数和迭代结果，并进行比较。  （3）分别设计单点弦截法和两点弦截法，计算原方程的根，在初值和容许误差相同的条件下比较它们的收敛速度。 参考答案 原方程的根为  答：（1）程序代码test1.py  test1.py执行  41NKZS595E06`{@3HB3{_@1   1. 程序test2.py   test2.py运行示例：  _L94_M0(QK{T{9GZ}TTV1LI  选取多个点，转换   |  |  |  |  |  | | --- | --- | --- | --- | --- | | **初值** | 是否收敛 | 迭代次数 | 迭代结果x | 迭代结果f | | **1.3** | 是 | 5 | 1.7320508075688834 | 5.861977570020827e-14 | | **1.4** | 是 | 5 | 1.7320508075688772 | -1.7763568394002505e-15 | | **1.5** | 是 | 4 | 1.7320508075697012 | 7.7973183465474e-12 | | **1.7** | 是 | 3 | 1.7320508075689427 | 6.190603585309873e-13 | | **2** | 是 | 4 | 1.73205080756904 | 1.539213201340317e-12 |   （3）  给定初值1.5，最大迭代次数1000  程序test3.py,单点弦截法  test3.py执行:  8$4%C7XY_$VATI`}0MSEQQD  程序test4.py两点弦截法  test4.py执行结果:  [24A[F~_{A2U{_HGDA%%$%8  从结果可知两点弦截法收敛速度比单点弦截法慢  三、思考题分析解答  ① 比较二分法和牛顿法在非线性方程求根中的优缺点和收敛速度。  答：二分法简单易行，但只有线性收敛速度，每次都只能将区间缩小一半，收敛速度慢，且二分法对于端点的函数值有限制条件，对于重根的情况无法解决；  牛顿法对于单根情形具有二阶局部收敛速度，收敛快，尤其是初值相当接近真值时，因而对初值的选择比较困难，在实际的计算中，要先获得真值的粗糙近似值，其次牛顿法每次迭代要计算，增加了计算量，对于重根情形仅线性收敛。  ②改进牛顿迭代法，使其对于重根也具有较高的收敛阶，试写出你所能想到的改进思路及其迭代格式，并简单分析收敛速度。  若有某根为重，则设  令，即可按照单根的情况计算。迭代格式为，整合公式得到最终的迭代格式为    至少平方阶收敛  四、重点难点分析  重点：   1. 掌握非线性方程采用迭代法的方式求根，从最开始的二分法到不动点迭代法，再到牛顿法迭代法的出现，使得非线性方程求根的速度得以提高。 2. 学会改进牛顿迭代法使得计算更加简便，如用导数的定义来替代求导，弦割法等等；学会针对不同根的情况，去改进牛顿迭代法，如对于重根来说牛顿迭代法的收敛速度是，需要改进。   难点：   1. 牛顿迭代法的收敛速度较快，但是需要选取初始点尽量与实际点接近。 2. 本实验编写程序的过程较为简单，但是理论部分中的如何推导出收敛阶较难，关键在于构造。 | | | | | | | |