《计算方法》实验指导书

“计算方法”课程的任务有两个：其一，掌握以计算机为计算工具的各类数值计算方法及与此相关的理论；其二，利用学到的计算机编程语言，将各类数值计算方法，编写计算程序、上机调试通过。因此，本课程不仅具有数学的抽象性与严格性，还具有与计算机密切结合的实用性。

“计算方法”课程的上机实习，是实现上述课程教学目的的重要手段。是将学到的计算机编程语言，结合具体任务，进行实践的必要过程。是学生计算机能力培养的一个重要环节。

按照课程教学要求，本课程的总学时为：48 学时，其中理论教学为 28 学时，实验教学学时为 20 学时。20 学时的上机实习中，设计有 8 个实习内容，每个内容的学时安排如下：

实验 1：舍入误差与数值稳定性　　验证性　（2 学时）
实验 2：方程求根—1——二分法　　综合性　（2-4 学时）
实验 3：方程求根—2——牛顿迭代法　　综合性　（2-4 学时）
实验 4：线性方程组数值解法　　设计性　（4-6 学时）
实验 5：插值法　　设计性　（2 学时）
实验 6：曲线拟和　　综合性　（4-6 学时）
实验 7：数值积分　　设计性　（2 学时）
实验 8：常微分方程的数值解法　　设计性　（2 学时）

序号	项目名称	实验课时	内容提要	教学要求（了解观察、熟悉、掌握、熟练掌握及运用等）	实验类别	实验方式	适用专业	备注
1	舍入误差与数值稳定性	2	录入、编辑、调试程序，熟悉 C 语言；通过不同算法的选用，了解舍入误差与数值的稳定性。	熟悉 C 语言程序设计及上机操作； 了解舍入误差所引起的数值不稳定性。	验证性	教师指导，学生独立完成	测控技术与仪器	必修
2	方程求根—1（二分法）	2-4	学生根据给定的要求，编写二分法程序，并调试通过； 通过实验输入不同的参数，了解该方法的特点。 完成实习报告。	掌握方程求根的二分法的算法； 熟悉二分法的特点。	综合性	根据实验指导书要求，学生独立完成，教师解惑	测控技术与仪器	必修
3	方程求根—2（牛顿迭代法）	2-4	学生根据给定的要求和牛顿迭代法的算法，编写牛顿迭代法程序，并调试通过； 通过实验输入不同的参数了解牛顿迭代法的特点。 完成实习报告。	掌握方程求根的牛顿迭代法的算法； 熟悉牛顿迭代法的特点。	综合性	根据实验指导书要求，学生独立完成，教师解惑	测控技术与仪器	必修

4	线性方程组数值解法（列主元高斯消去法）	4-6	熟悉掌握列主元高斯消去法算法； 学生根据给定的要求编写列主元高斯消去法程序，并调试通过； 编程能力强的在完成上面任务的基础上，还可以选做本章其他方法。	熟悉列主元高斯消元法解线性方程组的算法； 掌握二维数组、函数、输入输出语句的使用； 掌握列主元高斯消去法的编程。	设计性	根据实验指导书要求，学生独立完成，教师解惑	测控技术与仪器	必修
5	插值法（拉格朗日插值法）	2	熟悉拉格朗日插值多项式； 学生编写、录入程序，并调试通过； 通过实验了解该方法的特点。	掌握拉格朗日插值多项式，注意其特点。	设计性	根据实验指导书要求，学生独立完成，教师解惑	测控技术与仪器	选修
6	曲线拟合（最小二乘法）	4-6	熟悉最小二乘法的算法； 编写最小二法进行曲线拟合的程序，并调试通过； 用程序解决实际问题。 完成实习报告。	了解最小二乘法的基本原理； 熟悉最小二乘算法； 掌握最小二乘进行曲线拟合的编程； 通过程序解决实际问题。	综合性	根据实验指导书要求，学生独立完成，教师解惑	测控技术与仪器	必修
7	数值积分（龙贝格算法）	2	熟悉龙贝格求积的算法； 自编或录入龙贝格求积程序，并调试通过，体会该程序的巧妙处；	掌握龙贝格求积的算法； 掌握该算法的编程，了解该程序的巧妙设计处。	设计性	根据实验指导书要求，学生独立完成，教师解惑	测控技术与仪器	选修
8	常微分方程的数值解法（改进欧拉方法）	2	熟悉改进欧拉方法； 自编或录入改进欧拉程序，并调试通过； 有能力的还可完成龙格−库塔程序。	掌握改进欧拉算法； 掌握改进欧拉算法的编程。	设计性	根据实验指导书要求，学生独立完成，教师解惑	测控技术与仪器	选修

实验报告的要求：

（实验项目，例：）实验一　舍入误差与数值稳定性

班级：____________________　学号：______________________　姓名：___________________

一、目的和要求

二、实习内容

三、算法

四、实验步骤

五、实验结果

六、分析和讨论

七、心得（*可选）
①调试过程中遇到的问题和解决对策；②经验体会等。

实验一　舍入误差与数值稳定性

目的与要求：

1）通过上机编程，复习巩固以前所学程序设计语言及上机操作指令和方法；
2）通过上机运算，了解舍入误差所引起的数值不稳定性和大数吃小数的情况。

实验内容：

1）数值稳定与不稳定的计算公式
2）实验习题一的第1小题。（大数吃小数的问题）

思考：
1） 什么是数值稳定与不稳定的计算公式？
2） 习题一的第 1 小题：怎样是正确的？

其他（略）

（见课本 P185-187）

实验二　方程求根 1——二分法

目的与要求：

1）通过对二分法的编程练习，掌握方程求根的二分法的算法；
2）通过对二分法的上机运算，进一步体会二分法的特点。

算法：

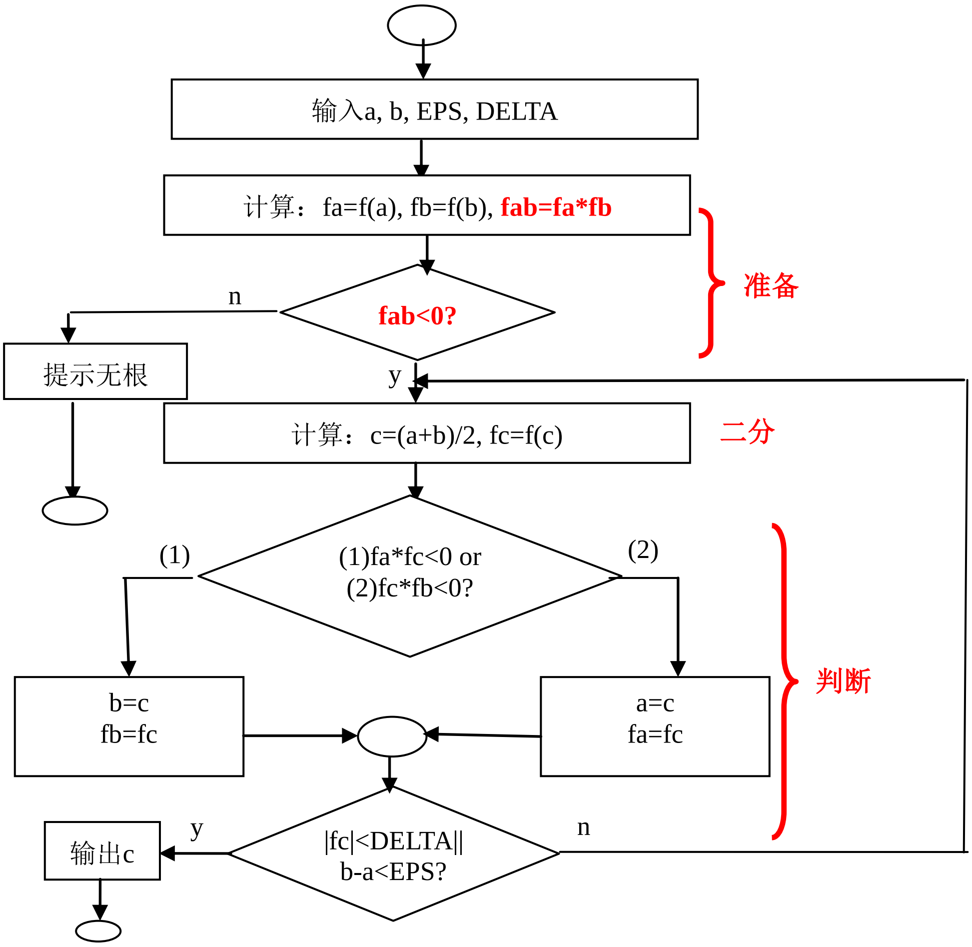
1）准备：计算 $f(x)$在有根区间[a, b]端点处的值 $f(a), f(b)$。

2）二分：计算$f(x)$在区间中点c=$\frac{a+b}{2}$处的函数值$f(c)$。

3）判断

- 若$f(c)$与$f(a)$异号，则根位于区间[a, c]内，以c代替b；
- 若$f(c)$与$f(a)$同号，则根位于区间[c, b]内，以c代替a；

反复执行步 2 和步 3，直到区间[a, b]长度缩小到允许误差范围之内或 $f(c)$=0，此时区间中点 c 即可作为所求的根。



实验内容：

1）二分法的编程实现。
2）进行有根区间和误差限的比较和讨论。

编程要求：

1）根的容许误差限 EPS 用输入语句输入。
2）根的区间 a,b 要求用输入语句输入。
3）输入初始值后，在调用二分法函数之前，先估算二分次数并输出。
4）二分法要写成函数形式：如函数 float Bisection(float a, float b,float EPS)　(*)

实验步骤：

1）完成二分法的程序设计及录入；
2）完成程序的编译和链接，并进行修改；
3）用书上的例子对程序进行验证，并进行修改；
4）对比估算次数与实际二分次数；
5）输入不同的区间初值 a, b，查看二分次数的变化；
6）输入不同的误差限，查看二分次数的变化；
7）完成实验报告。

（实验报告的编写见下一页）

实验报告　方程求根——二分法

班级：____________________　　学号：______________________　　姓名：__________________

一、目的和要求

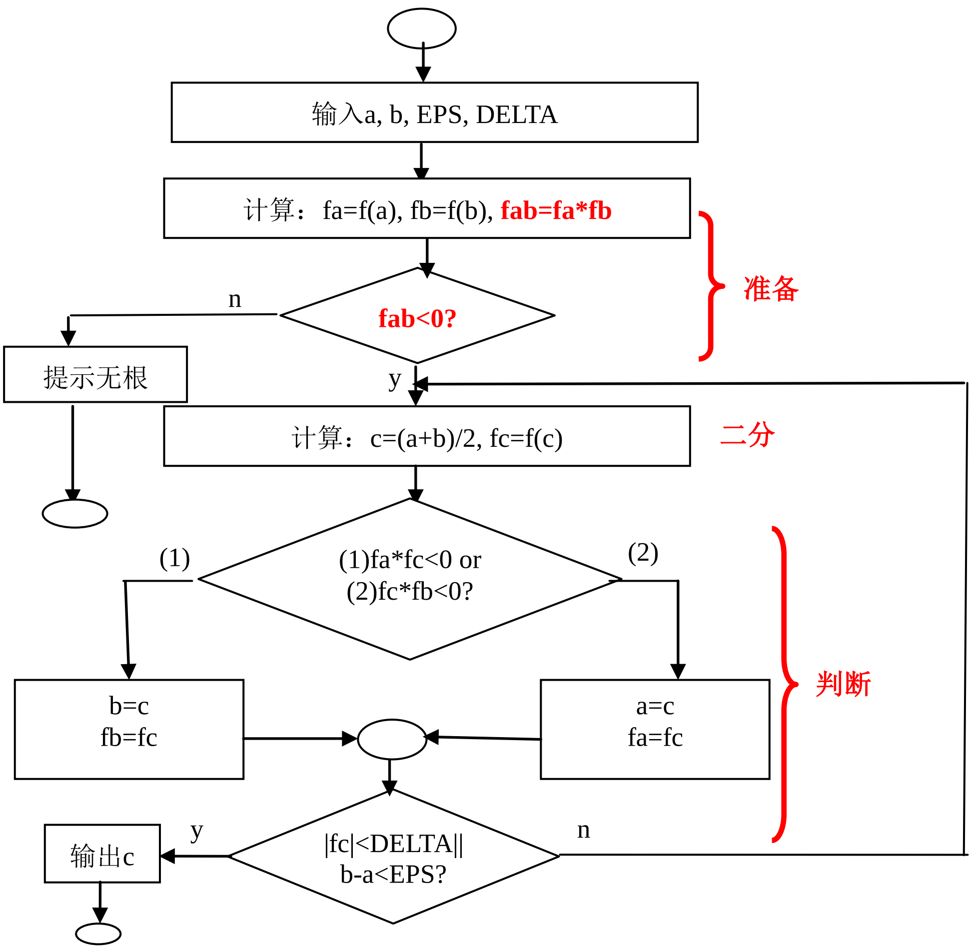
1）通过对二分法的编程练习，掌握方程求根的二分法的算法；
2）通过对二分法的上机运算，进一步体会二分法的特点。

二、实习内容

1）二分法的编程实现。
2）进行有根区间和误差限的比较和讨论。

三、算法

流程图：



1）准备：计算 $f(x)$在有根区间[a, b]端点处的值 $f(a), f(b)$。

2）二分：计算$f(x)$在区间中点c=$\frac{a+b}{2}$处的函数值$f(c)$。

3）判断

- 若$f(c)$与$f(a)$异号，则根位于区间[a, c]内，以c代替b；

● 若$f(c)$与$f(a)$同号，则根位于区间[c, b]内，以c代替a；

反复执行步 2 和步 3，直到区间$[a, b]$长度缩小到允许误差范围之内或 $f(c)=0$，此时区间中点 c 即可作为所求的根。

四、实验步骤

1）完成二分法的程序设计及录入；
2）完成程序的编译和链接，并进行修改；
3）用书上的例子对程序进行验证，并进行修改；
4）对比估算次数与实际二分次数；
5）输入不同的区间初值 a, b，查看二分次数的变化；
6）输入不同的误差限，查看二分次数的变化；
7）完成实验报告。

五、实验结果

1.　经编译、链接及例子验证结果正确的源程序：

2. 实例验证结果：
1）方程：$f(x)=x^3+x^2-3x-3=0$

2）输入初始参数：a=1, b=2, EPS=5e-6

3）结果输出：

3. 改变 a, b 的值为：a=0, b=2, EPS 不变，仍为 5e-6，其结果为：

4. 改变 EPS 的值为：EPS=5e-4, a, b 不变，仍为 a=1, b=2，其结果为：

六、分析和讨论

1. 估算次数与实际二分次数的分析和讨论

2. 输入不同的区间初值 a, b，二分次数的变化情况

3. 输入不同的误差限 EPS，二分次数的变化情况

七、心得

①调试过程中遇到的问题和解决对策；②经验体会等。

实验三　方程求根 2——牛顿迭代法

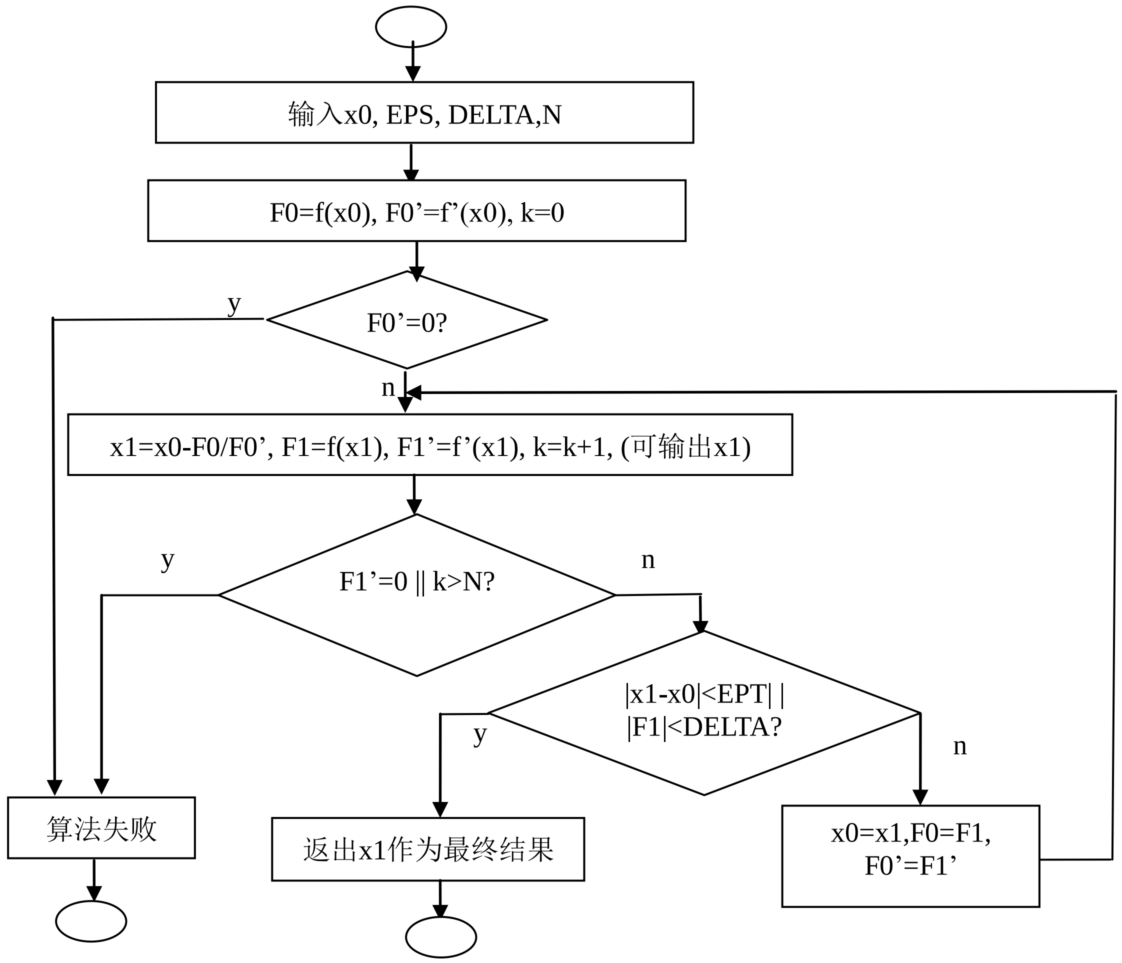
目的与要求：

1）通过对牛顿迭代法的编程练习，掌握方程求根的牛顿迭代法的算法；
2）通过对牛顿迭代法的上机运算，进一步体会牛顿迭代法的特点。

算法：

用迭代法的结构，增设 4 个工作单元 F0, F0’, F1, F1’，并把用作终止迭代的误差控制改为两个|x1-x0|<EPS 或|f(x1)|<DELTA。

1. 准备：选定初始值 x0, **计算** F0=f(x0); F0’=f’(x0), 如果 F0’=0，则输出“方法失败”并结束。
2. 迭代：对 k=1,2,…,N，做：
   1) x1=x0-F0/F0’ ，
   2) 计算 F1=f(x1); F1’=f’(x1)
   3) 若 F1’=0，则输出“方法失败”并结束。
3. 控制：若|x1-x0|<EPS 或|F1|<DELTA，则输出近似解 x1 和迭代次数 k 并结束；否则，x0=x1; F0=F1; F0’=F1’。
4. k>N 时输出“经 N 次迭代无满足要求的近似解”结束。



实验内容：

1）牛顿迭代法的编程实现。
2）进行初值和误差限的比较和讨论。

编程要求：

1）根的容许误差限 EPS 用输入语句输入。
2）根的初始值 x0 要求用输入语句输入。
3）牛顿迭代法要写成函数形式：如 float Newton(float x0, float EPS, int N)。(*)

实验步骤：

1）完成牛顿迭代法的程序设计及录入；
2）完成程序的编译和链接，并进行修改；
3）用书上的例子对程序进行验证，并进行修改；
4）分别输入两组不同的根的误差限，观察运算次数的变化；
5）分别取不同的初时值 x0，观察运算结果的变化；
6）完成实验报告。

（实验报告的编写见下一页）

实验报告 方程求根——牛顿迭代法

班级：___________________ 学号：_____________________ 姓名：__________________

一、目的和要求

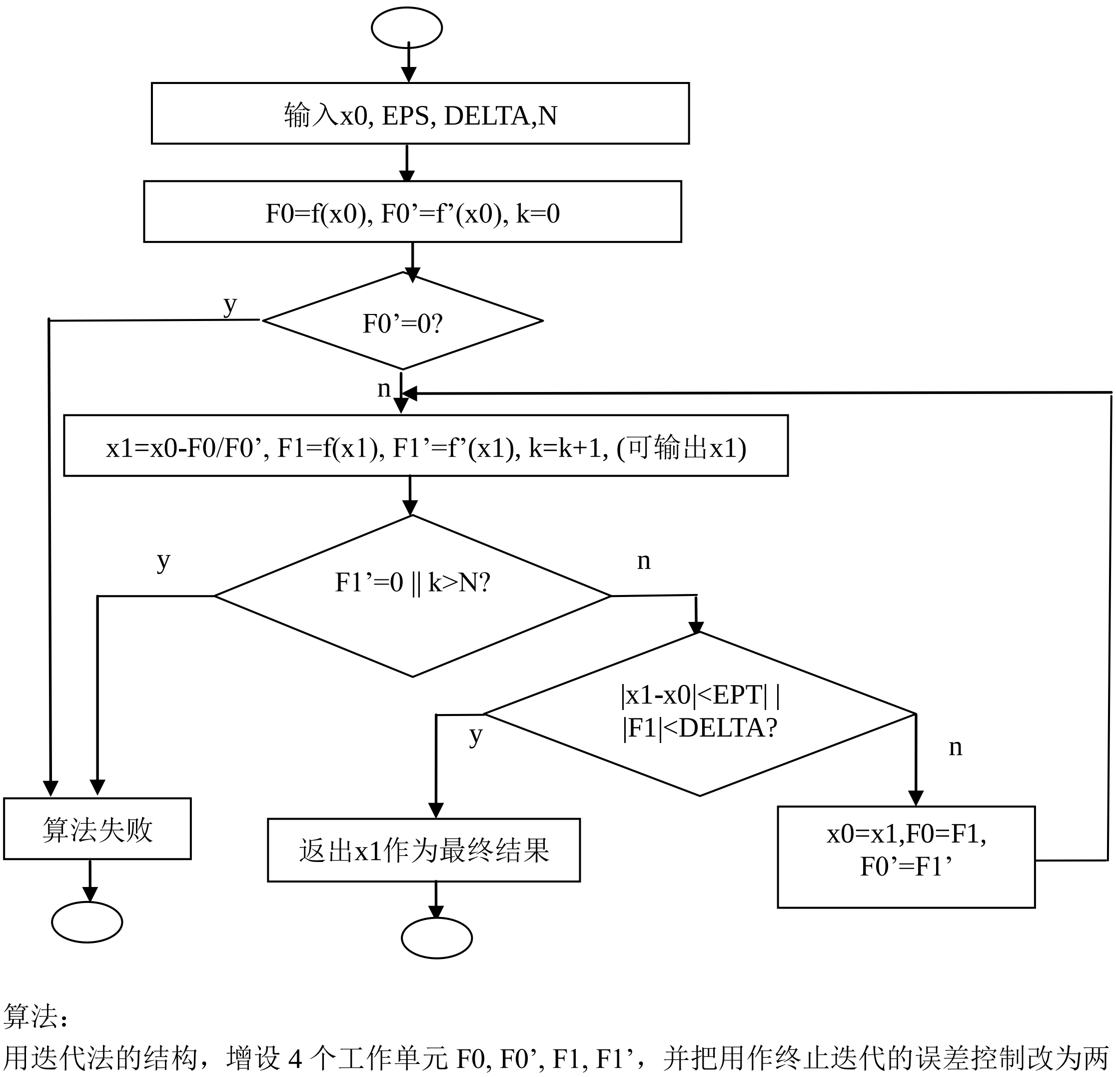
1）通过对牛顿迭代法的编程练习，掌握方程求根的牛顿迭代法的算法；
2）通过对牛顿迭代法的上机运算，进一步体会牛顿迭代法的特点。

二、实习内容

1）牛顿迭代法的编程实现。
2）进行初值和误差限的比较和讨论。

三、算法

流程图：



算法：
用迭代法的结构，增设 4 个工作单元 F0, F0’, F1, F1’，并把用作终止迭代的误差控制改为两个|x1-x0|<EPS 或|f(x1)|<DELTA。
1. 准备：选定初始值 x0, **计算** F0=f(x0); F0’=f’(x0), 如果 F0’=0，则输出“方法失败”并结束。
2. 迭代：对 k=1,2,…,N，做：

1) x1=x0-F0/F0’，

2) 计算 F1=f(x1); F1’=f’(x1)

3) 若 F1’=0，则输出“方法失败”并结束。

3. 控制：若|x1-x0|<EPS 或|F1|<DELTA，则输出近似解 x1 和迭代次数 k 并结束；否则，x0=x1; F0=F1; F0’=F1’。

4. k>N 时输出“经 N 次迭代无满足要求的近似解”结束。

四、实验步骤

1）完成牛顿迭代法的程序设计及录入；
2）完成程序的编译和链接，并进行修改；
3）用书上的例子对程序进行验证，并进行修改；
4）分别输入两组不同的根的误差限，观察运算次数的变化；
5）分别取不同的初时值 x0，观察运算结果的变化；
6）完成实验报告。

五、实验结果

1. 经编译、链接及例子验证结果正确的源程序：

2. 实例验证结果：

1）方程：$f(x)=x^3+x^2-3x-3=0$

2）输入初始参数：x0=1, EPS=1e-6

3）结果输出：

3. 改变初值 x0 的值为：x0=1.5,　EPS 不变，仍为 1e-6，其结果为：

4. 改变初值 x0 的值为：x0=0.1,　EPS 不变，仍为 1e-6，其结果为：

5. 改变 EPS 的值为：EPS=5e-4,　x0 不变，仍为 1，其结果为：

6. 改变 EPS 的值为：EPS=1e-3,　x0 不变，仍为 1，其结果为：

六、分析和讨论

1. 输入不同的初值 x0，迭代次数的变化情况

2. 输入不同的误差限 EPS，迭代次数的变化情况

七、心得

①调试过程中遇到的问题和解决对策；②经验体会等。

实验四：线性方程组数值解法——列主元高斯消去法

目的与要求：

1）熟悉列主元高斯消元法解线性方程组的算法；
2）掌握列主元高斯消去法的编程。

实验内容：

列主元高斯消去法的编程实现。

算法：

消元：对 k=0,2,…,n-2，按下列步骤进行：

- 选主元：找出 m∊{k,k+1,…,n-1}，使$\left|a_{m,k}^{(k)}\right| = \max\limits_{k \le i \le n}\left|a_{i,k}^{(k)}\right|$
- <u>若$|a_{m,k}|$<DELTA，则 A 奇异，结束程序，否则继续</u>
- 若 m≠k，交换第 k 行与第 m 行对应的元素（换行）：$a_{kj} \leftrightarrow a_{mj} \qquad j = k, \cdots, n$
- 消元：
  对 i=k+1,…,n-1，计算 $l=l_{ik}=a_{ik}/a_{kk}$;
  对 j=k+1,…,n-1,n，计算 $a_{ij}=a_{ij}-l_{ik}*a_{kj} =a_{ij}-l*a_{kj}$

回代：
- 若$|a_{nn}|$<DELTA，则 A 奇异，结束程序，否则继续
- $x_{n-1}=a_{n-1,n} / a_{n-1, n-1}$
- 对 i = n-2 ,…, 1, 0，计算：$x_i = \left(a_{i,n} - \sum\limits_{j=i+1}^{n-1} a_{ij}x_j\right) \Big/ a_{ii}$

编程要求：

1）方程组的矩阵系数用二维数组表示，不用指针，且其值要求用输入语句输入。（数组形式的完成，经检查后，有能力的可以改用指针方式）
2）方程组的元数 n 要求用输入语句输入。
3）加入中间运算结果的显示，以便于检查程序。
4）列主元解方程组方法写成函数形式。如：void ColPivot(float a[10][11], int n, float x[10])

实验步骤：

1）完成列主元高斯消去法解线性方程组的程序设计及录入、编辑；
2）完成程序的编译和链接，并进行修改；
3）用书上 P52 例 2 的例子对程序进行验证，并进行修改；
4）用完成的程序解算习题中的题目。

实验五：插值法——拉格朗日插值多项式

目的与要求：

1） 熟悉拉格朗日插值多项式，注意其特点。
2） 掌握拉格朗日插值法的编程。

实验内容：

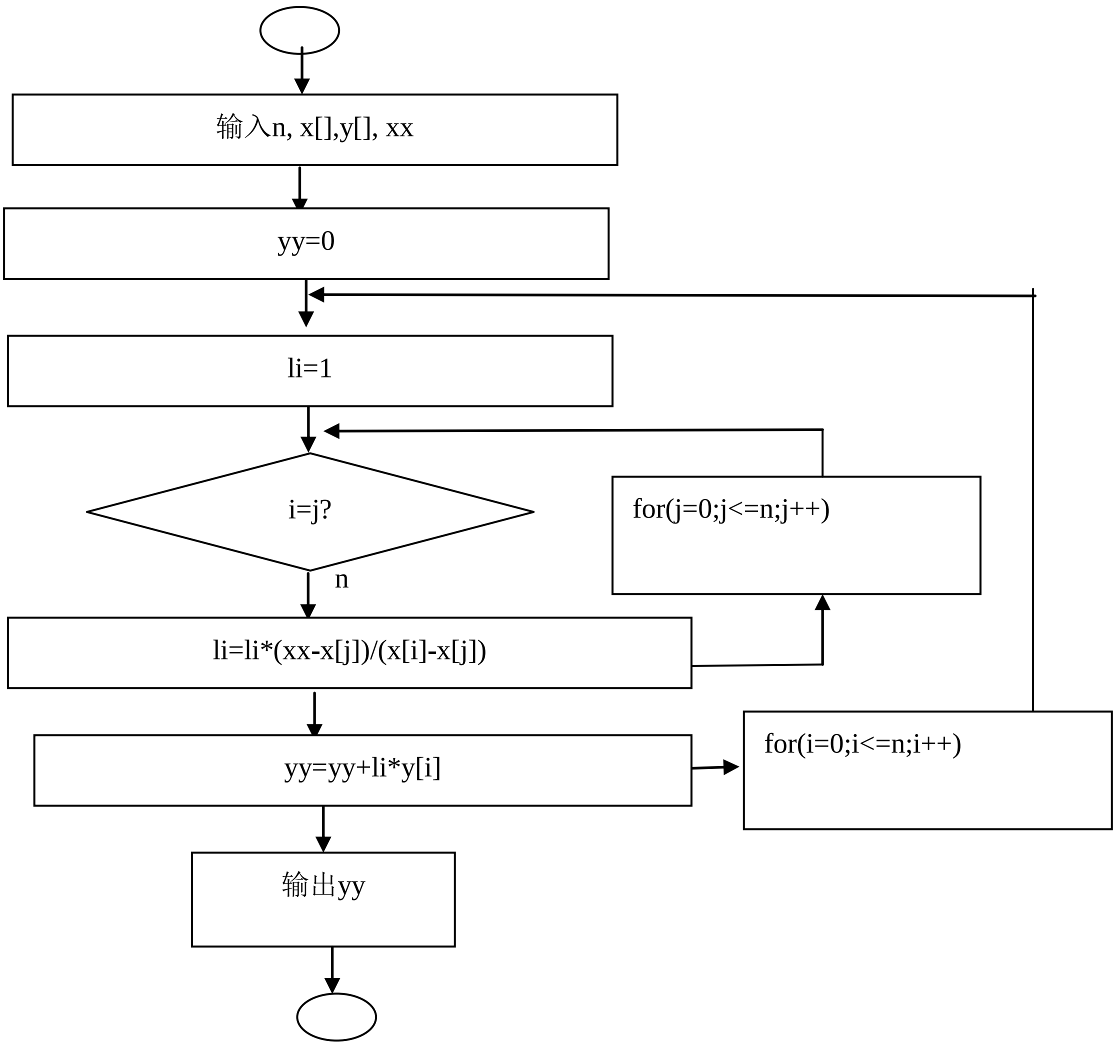
拉格朗日插值法的编程实现。

算法：

1）输入数据：n，x[i]，y[i]（i=0，1，2，……，n-1），xx
2）初始化：li=1，yy=0
3）计算： i 循环(外循环):

j 循环(内循环) (j=0,1,2,…,n-1)： $li = li * (xx - x[j]) / (x[i] - x[j])$

$yy = yy + li * y[i]$ （i=0,1,2,…,n-1）



编程要求：

1）插值节点个数 n 用输入语句输入。
2）用输入语句输入各节点的数据 x_i, y_i (i=0,1,2,…,n) 。
3）用输入语句输入待插值点的 x 值 xx。
4）拉格朗日插值计算用函数形式：如函数 float Lagrange(float x[],float y[],float xx,int n)。（*）

实验步骤：

1）完成拉格朗日插值多项式进行插值的程序设计及录入、编辑；
2）完成程序的编译和链接，并进行修改；
3）用书上的例子对程序进行验证，并进行修改；
4）用完成的程序解算习题中的题目。

实验六：曲线拟合——最小二乘法

目的与要求：

1）了解最小二乘法的基本原理，熟悉最小二乘算法；
2）掌握最小二乘进行曲线拟合的编程，通过程序解决实际问题。

实验内容：

1）最小二乘进行多项式拟合的编程实现。
2）用完成的程序解决实际问题。

算法：

1）输入数据节点数 n，拟合的多项式次数 m，循环输入各节点的数据 x_j, y_j(j=0,1,…,n-1)
2）由 x_j 求 S；由 x_j,y_j 求 T：

$$\mathbf{S_k} = \sum_{j=0}^{n-1} x_j^k \quad \mathbf{(\ k=0,1,2,\ \cdots\ 2*m\ )}$$

$$\mathbf{T_k} = \sum_{j=0}^{n-1} y_j x_j^k \quad \mathbf{(\ k=0,1,2,\cdots\ m\ )}$$

3）由 S 形成系数矩阵数组 $c_{i,j}$：c[i][j]=S[i+j] (i=0,1,2,…m, j=0,1,2,…,m)；由 T 形成系数矩阵增广部分 $c_{i,m+1}$：c[i][m+1]=T[i]　(i=0,1,2,…m)

4）对线性方程组 CA=T[或 $\overline{C}A$]，用列主元高斯消去法求解系数矩阵 $A=(a_0,a_1,\ldots,a_m)^T$

AX=B 或[$\overline{A}X$]

编程要求：

1）直接调用高斯消去法解方程组的函数。
2）数据节点数 n，拟合次数 m，及数据 x_j, y_j(j=0,1,…,n-1)要求用输入语句输入。
3）数据使用数组方式，不用指针方式。
4）实验报告中写出计算公式，所有脚标要求与程序中的循环变量对应。
5）加入中间运算结果的显示，以便于检查程序。
6）拟合部分用函数形式：如函数 void Approx(float x[], float y[], int n, int m, float a[])。（*）

实验步骤：

1）完成最小二乘法进行曲线拟合的程序设计及录入、编辑；
2）完成程序的编译和链接，并进行修改；
3）用书上 P105 例 2 的例子对程序进行验证，并进行修改；
4）用完成的程序求解下面的实际问题。
5）完成实验报告。

问题：

作物体运动的观测实验，得出以下实验测量数据，用最小二乘拟合求物体的运动方程。

时间 t(秒)	0	0.9	1.9	3.0	3.9	5.0
距离 s(cm)	0	10	30	50	80	110

解题步骤：

（1） 画草图

（2） 确定拟合方程

（3） 用完成的程序输入数据，求取拟合方程中的未知数，得出方程。

（实验报告见下一页）

实验报告　曲线拟合——最小二乘法

班级：__________________　　学号：____________________　　姓名：_________________

一、目的和要求

1）了解最小二乘法的基本原理，熟悉最小二乘算法；
2）掌握最小二乘进行曲线拟合的编程，通过程序解决实际问题。

二、实习内容

1）最小二乘进行多项式拟合的编程实现。
2）用完成的程序解决实际问题。

三、算法

1）输入数据节点数 n，拟合的多项式次数 m，循环输入各节点的数据 x_j, y_j(j=0,1,…,n-1)
2）由 x_j 求 S；由 x_j,y_j 求 T：

$$\mathbf{S_k} = \sum_{j=0}^{n-1} x_j^k \quad \mathbf{(\ k=0,1,2,\ \cdots\ \ 2*m\ )}$$

$$\mathbf{T_k} = \sum_{j=0}^{n-1} y_j x_j^k \quad \mathbf{(\ k=0,1,2,\cdots\ \ m\ )}$$

3）由 S 形成系数矩阵数组 $c_{i,j}$：c[i][j]=S[i+j] (i=0,1,2,…m, j=0,1,2,…,m)；由 T 形成系数矩阵增广部分 $c_{i,m+1}$：c[i][m+1]=T[i]　(i=0,1,2,…m)

4）对线性方程组 CA=T[或 $\overline{C}A$]，用列主元高斯消去法求解系数矩阵 $A=(a_0,a_1,\ldots,a_m)^T$

AX=B 或[$\overline{A}X$]

四、实验步骤

1）完成最小二乘法进行曲线拟合的程序设计及录入、编辑；
2）完成程序的编译和链接，并进行修改；
3）用书上 P105 例 2 的例子对程序进行验证，并进行修改；
4）用完成的程序求解下面的实际问题。
5）完成实验报告。

五、实验结果

1. 经编译、链接及例子验证结果正确的源程序：

2. 实例验证结果：

1）输入初始参数：

n=9，m=2
X： 1 3 4 5 6 7 8 9 10
Y：10 5 4 2 1 1 2 3 4

2）结果输出：

3. 实际应用

问题：
作物体运动的观测实验，得出以下实验测量数据，用最小二乘拟合求物体的运动方程。

时间 t(秒)	0	0.9	1.9	3.0	3.9	5.0
距离 s(cm)	0	10	30	50	80	110

解题步骤：
1）画草图

2）确定拟合方程次数为 1：

用完成的程序输入数据，求取拟合方程中的未知数，得出方程：

计算误差：

3）确定拟合方程次数为 2：

用完成的程序输入数据，求取拟合方程中的未知数，得出方程：

计算误差：

六、分析和讨论

结合实际问题，进行拟合次数的分析和讨论：

七、心得（*可选）

①调试过程中遇到的问题和解决对策；②经验体会等。

实验七：数值积分——龙贝格算法

目的与要求：

1）熟悉龙贝格求积的算法；
2）掌握该算法的编程，了解该程序的巧妙设计处。

实验内容：

龙贝格算法的编程实现。

算法：

$$T_{2n}=\frac{1}{2}T_n+\frac{h_n}{2}\sum_{k=0}^{n-1}f(x_{k+\frac{1}{2}}) \qquad h_n=\frac{b-a}{n},x_{k+\frac{1}{2}}=(k+\frac{1}{2})h_n$$

$$S_n=T_{2n}+\frac{1}{3}(T_{2n}-T_n)$$

$$C_n=S_{2n}+\frac{1}{15}(S_{2n}-S_n)$$

$$R_n=C_{2n}+\frac{1}{63}(C_{2n}-C_n)$$

用事后误差估计法控制精度$\left|R_{2n}-R_n\right|<\varepsilon$

编程要求：

1）先自己根据公式编写，再与课本的例子对比，了解其设计的巧妙。
2）龙贝格算法用函数形式：如函数 float Romberg(float a, float b, float EPS)。

实验步骤：

1）完成龙贝格算法进行数值积分的程序设计及录入、编辑；
2）完成程序的编译和链接，并进行修改；
3）用书上的例子对程序进行验证，并进行修改；

注：可参考课本 P214-216。

实验八：常微分方程数值解法——改进欧拉方法

目的与要求：

1）熟悉改进欧拉算法；

2）掌握改进欧拉算法的编程。

实验内容：

改进欧拉算法的编程实现。

算法：

$$\begin{cases} \tilde{y}_{i+1} = y_i + hf(x_i, y_i), h = \dfrac{b-a}{n} \\ y_{i+1} = y_i + \dfrac{h}{2}[f(x_i, y_i) + f(x_{i+1}, \tilde{y}_{i+1})] \end{cases}$$

$$\begin{cases} y_p = y_i + hf(x_i, y_i) \\ y_c = y_i + hf(x_{i+1}, y_p) \\ y_{i+1} = \dfrac{1}{2}(y_p + y_c) \end{cases}$$

编程要求：

1）自己根据公式编写，再与课本的例子对比。

2）等分小区间数 n 用输入语句输入。

3）初值 a、b、ya，用输入语句输入。

4）改进欧拉算法用函数形式：如函数 void ModEuler(float a, float ya, float b, int n)

实验步骤：

1）完成改进欧拉算法求解常微分方程的程序设计及录入、编辑；

2）完成程序的编译和链接，并进行修改；

3）用书上的例子对程序进行验证，并进行修改；

注：可参考课本 P217-219。