计算方法第一次实验报告

姓名：焦培淇 学号：PB17151767

问题1：

1. 计算结果

|  |  |  |
| --- | --- | --- |
| x | (x^2+9)^0.5-3 | (x^2)/((x^2+9)^0.5+3) |
| 0.125000000000E-000 | 0.260303736608E-002 | 0.260303736608E-002 |
| 0.156250000000E-001 | 0.406898282228E-004 | 0.406898282230E-004 |
| 0.195312500000E-002 | 0.635782810399E-006 | 0.635782810234E-006 |
| 0.244140625000E-003 | 0.993410731454E-008 | 0.993410744612E-008 |
| 0.305175781250E-004 | 0.155220281073E-009 | 0.155220429099E-009 |
| 0.381469726563E-005 | 0.242517117499E-011 | 0.242531920473E-011 |
| 0.476837158203E-006 | 0.377475828373E-013 | 0.378956125739E-013 |
| 0.596046447754E-007 | 0.444089209850E-015 | 0.592118946467E-015 |
| 0.745058059692E-008 | 0.000000000000E+000 | 0.925185853854E-017 |
| 0.931322574615E-009 | 0.000000000000E+000 | 0.144560289665E-018 |

注：表中“x”一列从上到下依次是8-1,8-2，…

1. 算法分析

x值的计算公式为8^i(i=-1,-2,...-10)，程序计算得到结果为第一列；第一种方法的计算公式为(x^2+9)^0.5-3，程序计算得到的结果为第二列；第二种方法的计算公式为(x^2)/((x^2+9)^0.5+3)。这三列数据都可以利用python语言中的math库函数直接计算得到，数据类型则通过numpy模块的dtype来定义为float32。

1. 结果分析

从程序的计算结果可以看出只有第一组数据，两种方法所得到的计算结果相同。从第二组数据开始，随着x值的减小，两种方法所得到的计算结果相对差距不断增大，可以看出倒数第三组数据两者的比值已经超过了1.3，而对于最后两组数据，使用第一种方法所得到的结果已经为0，而第二种方法所得到的结果大约是10^-18量级。可以很明显的看出第二种方法得到的结果更加准确。虽然啊(x^2+9)^0.5-3和(x^2)/((x^2+9)^0.5+3)从数学上来说是两个相同的公式，但是当x趋于零时，第二个公式计算所得的值确更加准确，这是因为当x趋于零时，(x^2+9)^0.5趋于3，第一个公式是两个相近的数相减，而对于第二个公式，分子趋于0，分母趋于6就不会存在两个相近的数相减的问题。因此第二个公式计算的值更加准确。

1. 实验结论

在利用程序进行计算时应当避免两个相近的数相减，否则会产生很大的计算误差；此时应当利用数学知识，对计算式进行变化，来增加计算的精确度。

问题2：

1、计算结果

|  |  |  |  |
| --- | --- | --- | --- |
|  | **方法(a)** | **方法(b)** | **方法(c)** |
| **计算结果** | 1.025188E-10 | -1.564331E-10 | 0.000000E+0 |

2、算法分析

原始数据为4040.045551380452、-2759471.276702747、-31.64291531266504、2755462.874010974、0.0000557052996742893，5个数据均可采用numpy的float64类型存储，通过控制相加的顺序可以得到三种方法的结果见上表。

3、结果分析

采用精确计算可得这五个数的和精确值为8.66342893E-11，即0.866342893E-10，对比上述三种算法可见方法a的结果最为精确。

4、实验结论

观察相加的几个数据可以看出，最后一个数据的小数位数最多，因此可能造成最多的误差。对比几种算法，可以发现，第一种算法最后处理这个数据，因此造成的误差最小，其他方法提前计算最后一个数据，因此会产生较大误差。因此，在进行精确计算时，要将可能造成误差最多的数据放在最后处理，这样才能减小误差。