数值分析实验报告

1170503101 罗猛

一．实验要求

使用 **python3** 用不动点迭代法求出方程根

二．实验目的

对给定不同的 **g**（**x**）所求根的精度判断

三．实验过程截图，代码及结果

1. g1(x) = x - x3 - 4 * (x**2) + 10**

代码：

```
#!/usr/bin/env python3
# -*- coding:utf-8 -*-
from sympy import *
import random


def f(x):
   return x**3 + 4*(x**2) -10


x = symbols("x")
g1 = x - x**3 - 4 * (x**2) + 10
```

```
begin = 1    #区间
end = 2
MAXSTEP = 100 #最大迭代次数
step_count = 0 #实际迭代次数
x0 = random.uniform(begin, end) #在区间中随机取 x0
temp = g1.subs(x, x0)

while step_count < MAXSTEP and abs(temp - x0) > 1e-10:
        x0 = temp
        temp = g1.subs(x, x0)
        step_count += 1
if step_count == N:
    print("不收敛")
print("求得的根为: ",x0)
print("迭代次数: ",step_count)
print("将所求的根带回原方程: ",f(x0))
print("\n")                                        #end
```

结 果：

```
求得的根为:  1.10477925345398e+84592215243453412313207101496688661517185604216
迭代次数:  100
将所求的根带回原方程:  1.34842417537194e+25377664573036023693962130449006598845
```

2. g2(x) = (10/x - 4*x)0.5**

代码：将 **1** 中 **g1(x)**替换为 **g2(x)**，后面 **3**，**4**，**5** 同理

结果：

```
求得的根为:  1.10477925345398e+84592215243453412313207101496688661517185604216
迭代次数:  100
将所求的根带回原方程:  1.34842417537194e+2537766457303602369396213044900659845
```

3. g3(x) = 0.5 * (10 - x3)**0.5**

结果：

```
求得的根为:  1.36522967031038
迭代次数:  19
将所求的根带回原方程:  -5.66580760974489e-6
```

4. g4(x) = (10/(4 + x))0.5**

结果：

```
求得的根为:  1.36522972897371
迭代次数:  7
将所求的根带回原方程:  -4.69707695849308e-6
```

5. g5(x) = x - (x3 + 4 * x**2 - 10)/(3 * x**2 + 8*x)**

结果：

```
求得的根为:  1.36523002724976
迭代次数:  3
将所求的根带回原方程:  2.28473849617217e-7
```

四．实验结论

由实验结果给出的各个公式的迭代次数得出，**g5(x)**收敛速度最快