

作业描述：

3. 用二分法在区间 $[0,1]$ 内求解 $f(x) = e^x - 2$ （自定合理的收敛条件，要求至少迭代 10 次），（1）基于 C 语言实现求解算法；（2）根据收敛速度的定义估算此迭代过程的收敛速度；（3）给出用双精度浮点数求解此方程所能达到的最高精度的根。

求解算法

```
#include <stdio.h>
#include <math.h>
#define eps 1e-16

double f(double x) {
    return exp(x) - 2;
}

double bisection(double a, double b) {
    double c = b;
    int i=0;
    for (i = 0; i < 10 || fabs(f(c)) > eps; i++) {
        c = (a + b) / 2;
        if (f(a)*f(c) > 0){
            a=c;
        }
        else if (f(a)*f(c) < 0){
            b=c;
        }
    }
    printf("The answer is %.17lf", c);
    return 0;
}

int main(){
    bisection(0.0,1.0);
    return 0;
}
```

收敛速度

区间长度初始为 $l_0=b-a$,每次迭代后为原长度一半, 即 n 次迭代为 $l_n = \frac{b-a}{2^n}$,由定义, 此为线性收敛!

实现效果

```
The answer is 0.69314718055994529
-----
Process exited after 0.1058 seconds with return value 0
请按任意键继续 . . . |
```

当将收敛条件升格至 $\text{eps}=1\text{e-}17$ 时, 得到:

```
|
```

即死循环无输出结果, 这是由双精度浮点数精确范围决定的。

由以上运行与分析, 双精度浮点数求解次方程能得到的最精确的根即为: 0.69314718055994529。