Computing Method - Programming 2

Zhixin Zhang, 3210106357

1 问题

用 Romberg 方法计算积分：

$$\frac{2}{\sqrt{\pi}}\int_0^1 e^{-x^2}\mathrm{d}x.$$

2 公式与算法

Romberg 算法流程如下：

1. 输入 a, b, ε.
2. 设置 $h = (b-a)/2, T_0^{(0)} = h(f(a)+f(b)), k = 1, n = 1$.
3. 设置 $F = 0$.对 $i = 1, 2, \cdots n$，计算 $F = F + f(a + (2i-1)h)$.
4. $T_0^{(k)} = T_0^{(k-1)}/2 + hF$.
5. 对 $m = 1, 2, \cdots k$，计算

$$T_m^{(k-m)} = \frac{4^m T_{m-1}^{(k-m+1)} - T_{m-1}^{(k-m)}}{4^m - 1}$$

6. 若 $\left|T_m^0 - T_{m-1}^{(0)}\right| < \varepsilon$，输出 $I \approx T_m^{(0)}$，停机； 否则置 $\frac{h}{2} \Rightarrow h, 2n \Rightarrow n, k+1 \Rightarrow k$，返回步骤 3.

3 程序

```
a = 0;
b = 1;
e = 1e-6;
syms x;
f(x) = exp(-x^2) * 2 / sqrt(pi);
Romberg(f, a, b, e);

function res = Romberg(f, a, b, eps)
    T = zeros(1, 1);
    n = 1;
    k = 1;
    h = (b-a)/2;
    T(1, 1) = h * (f(a) + f(b));

    while true
       F = 0;
       for i = 1 : n
           F = F + f(a + (2*i-1)*h);
       end
       T(k+1, k+1) = 0;
       T(1, k+1) = T(k, 1) / 2 + h * F;
       for m = 1 : k
           T(m+1, k-m+1) = (4^m * T(m, k-m+2) - T(m, k-m+1)) / (4^m - 1);
       end
```

```
        if(abs(T(k+1, 1) - T(k, 1)) < eps)
            break;
        end
        h = h/2;
        n = 2*n;
        k = k+1;
    end
    disp('Romberg Table: ')
    for col = 1:k+1
        for row = 1:k+1-row+1
            fprintf('%.7f ', R(row, col));
        end
        fprintf('\n');
    end
    res = T(k+1, 1);
end
```

4 数据与结果

输入 $a = 0, b = 1, \varepsilon = 1e - 6$，程序运行结果如下：

```
>> main
Romberg Table:
0.7717433 0.8431028 0.8553977 0.8507744 0.8455215 0.8430004 0.8423627 0.8424262 0.8425874 0.8426794 0.8427083 0.8427097 0.8427051 0.8427019 0.8427007 0.8427005
0.8252630 0.8546293 0.8508466 0.8455420 0.8430028 0.8423628 0.8424262 0.8425873 0.8426794 0.8427083 0.8427097 0.8427051 0.8427019 0.8427007 0.8427005
0.8472877 0.8510830 0.8456249 0.8430128 0.8423634 0.8424262 0.8425873 0.8426794 0.8427083 0.8427097 0.8427051 0.8427019 0.8427007 0.8427005
0.8501342 0.8459660 0.8430536 0.8423660 0.8424261 0.8425873 0.8426793 0.8427083 0.8427097 0.8427051 0.8427019 0.8427007 0.8427005
0.8470081 0.8432356 0.8423767 0.8424259 0.8425871 0.8426793 0.8427083 0.8427097 0.8427051 0.8427019 0.8427007 0.8427005
0.8441787 0.8424304 0.8424251 0.8425865 0.8426792 0.8427083 0.8427097 0.8427051 0.8427019 0.8427007 0.8427005
0.8428675 0.8424254 0.8425840 0.8426789 0.8427083 0.8427097 0.8427051 0.8427019 0.8427007 0.8427005
0.8425359 0.8425741 0.8426774 0.8427082 0.8427097 0.8427051 0.8427019 0.8427007 0.8427005
0.8425645 0.8426709 0.8427077 0.8427097 0.8427051 0.8427019 0.8427007 0.8427005
0.8426443 0.8427054 0.8427097 0.8427051 0.8427019 0.8427007 0.8427005
0.8426901 0.8427094 0.8427052 0.8427019 0.8427007 0.8427005
0.8427046 0.8427055 0.8427020 0.8427007 0.8427005
0.8427053 0.8427022 0.8427007 0.8427005
0.8427030 0.8427008 0.8427005
0.8427013 0.8427005
0.8427007
```

图 1 Result of Romberg Algorithm

可以发现，一共进行了 15 次迭代，最终答案为 $T_{15}^{(0)} = 0.8427007$.

5 结论

该积分的真值近似为 $T \approx 0.842700792950$. 该算法的求解误差为 $\left|T - T_{15}^{0}\right| \approx 9e - 8 < 1e - 7$.