HW4 solution. MAE 185. Spring 2006

All blames to Nasser Abbasi

June 24, 2013

1   Problem

Integrate the following using Romberg integration

$$\int_0^1 e^{x^2}\,dx, \quad \int_0^1 x^x\,dx, \quad \int_0^1 \sin\left(50\ x\right)dx, \quad \int_0^{10} \ln\left(x\right)dx, \quad \int_0^1 \frac{1}{\sqrt{x}}e^{x^2}\,dx, \quad \int_1^5 \frac{(x-1)^{0.2}}{x^2+1}dx$$

2   Answer

An important part is to select a tolerance which can be used to stop the Romberg iteration. This is the tolerance shown in the pseudocode on page 212 with the letter ϵ.

The source code is attached at the end, and the output.txt (in which the Romberg table is also printed for each problem).

The following is the Fortran source code to solve this HW. **??**

The following is the output from running the program. **??**

2.1   Problem 1

$$\int_0^1 e^{x^2}\,dx$$

No preprocessing needed. Use Romberg integration on the integral as shown.

The answer is 1.4626517459

2.2   Problem 2

$$\int_0^1 x^x\,dx$$

No preprocessing needed. Use Romberg integration on the integral as shown.

The answer is 0.7834305142

1

2.3 Problem 3

$$\int\limits_0^1 \sin(50\,x)\,dx$$

No preprocessing needed. Use Romberg integration on the integral as shown. Just make sure the tolerance is small enough. Use a multiple of machine double precision limit.

The answer is 0.0007006794

2.4 Problem 4

$$\int\limits_0^{10} \ln(x)\,dx$$

To handle the singularity at $x = 0$ we can either integrate from some very small $\epsilon > 0$ or use integration by parts to change the integrand to something without singularity.

Using integration by parts. $\int u\,dv = uv - \int v\,du$ hence let $u = \ln(x)$ and $dv = dx$. Hence $du = \frac{1}{x}dx$ and $v = x$.

$$\int\limits_0^{10} (\ln(x) \times 1)\ dx = [x\ \ln(x)]_0^{10} - \int\limits_0^{10} x\left(\frac{1}{x}dx\right) = [10 \times \ln(10)] - \int\limits_0^{10} dx$$

So the integral becomes

$$23.02585 - \int\limits_0^{10} dx$$

(Clearly we do not need to do numerical integration now, since the answer is 13.02585 by just looking at the above, but we need to use Romberg integration, so our integrand here is 1)

The answer is 13.0258509299

2.5 Problem 5

$$\int\limits_0^1 \frac{1}{\sqrt{x}} e^{x^2}\,dx$$

To handle the singularity at $x = 0$ we can either integrate from some very small $\epsilon > 0$ or use integration by parts to change the integrand to something without singularity.

Using integration by parts. $\int u\,dv = uv - \int v\,du$ hence let $u = e^{x^2}$ and $dv = \frac{1}{\sqrt{x}}dx$. Hence $du = 2xe^{x^2}dx$ and $v = \int x^{-\frac{1}{2}}dx = 2\sqrt{x}$

Hence

$$\int\limits_0^1 \frac{1}{\sqrt{x}} e^{x^2}\,dx = \left[2\sqrt{x}e^{x^2}\right]_0^1 - \int\limits_0^1 2\sqrt{x}\left(2x\ e^{x^2}\right)dx$$

$$= \left[2\sqrt{x}e^{x^2}\right]_0^1 - 4\int\limits_0^1 x^{\frac{3}{2}}\ e^{x^2}\,dx$$

2

And we apply Romberg integration on the second part above.
The answer is 2.5425742098

2.6   Problem 6

$$\int_1^5 \frac{(x-1)^{0.2}}{x^2+1}dx$$

No preprocessing needed. Use Romberg integration on the integral as shown.
The answer is 0.5613773544

3   Appendix

```fortran
!***************************************
!* Solve HW4, MAE 185. Spring 2006
!* UCI.
!* all blames to Nasser Abbasi
!*
!* Compiler: g95 on windows XP
!* (GCC 4.0.2 (g95) Mar  3 2006)
!***************************************
PROGRAM hw4
IMPLICIT NONE

REAL*8 result,solve_problem
INTEGER I

   OPEN(UNIT=999,FILE='hw4_output.txt')
   DO I=1,6
     result=solve_problem(I)
     WRITE(UNIT=999,FMT='(A,I1,A,F14.10)') '\n result problem ',I,'=',result
     WRITE(UNIT=999,FMT='(A)') '============================='
   END DO

END PROGRAM

!*************
!*
!*************
REAL*8 function solve_problem(I)
IMPLICIT NONE

real*8 :: romberg,TOL,xa,xb
real*8,external:: f_1,f_2,f_3,f_4,f_5,f_6
PARAMETER (TOL=1D5*EPSILON(1.0D0))  ! pick some tolerance value
integer :: RMAX,I
parameter (RMAX=12 )

    SELECT CASE(I)
      CASE(1)
        xa=0.0d0
        xb=1.d0
        solve_problem = romberg(xa,xb,f_1,TOL,RMAX)
      CASE(2)
        xa=0.0d0
        xb=1.d0
        solve_problem = romberg(xa,xb,f_2,TOL,RMAX)
      CASE(3)
        xa=0.0d0
        xb=1.d0
        solve_problem = romberg(xa,xb,f_3,TOL,RMAX)
      CASE(4)
        xa=0.0d0
        xb=10.d0
        solve_problem = xb*LOG(xb) - romberg(xa,xb,f_4,TOL,RMAX)
      CASE(5)
        xa=0.0d0
        xb=1.d0
        solve_problem = 2*SQRT(xb)*EXP(xb**2) - 4*romberg(xa,xb,f_5,TOL,RMAX)
      CASE(6)
        xa=1.d0
        xb=5.d0
        solve_problem = romberg(xa,xb,f_6,TOL,RMAX)

    END SELECT
END

!***********************
!* ROMBERG
!*
!***********************
REAL*8 function romberg(xa,xb,f,EPSILON,RMAX)
IMPLICIT NONE

real*8, external:: f
real*8  :: xa,xb,dx,epsilon,x,s,dI
integer :: RMAX,r,k,m,c

real*8  :: I(0:RMAX,0:RMAX) ! the romberg table

 r  = 0
 dx = xb-xa
 I(0,0) = dx* ( f(xa)+f(xb) )/2

 WRITE(UNIT=999,FMT='(F14.10)',advance='no') I(0,0)

 DO WHILE( .TRUE. )

  r  = r+1
  dx = dx/2
  k  = 2**(r-1)
  s  = 0

  DO m = 1,k
    x = xa + (2*m-1)*dx
    s = s  + f(x)
```

5

4 Output

```
 1.8591409142   1.5715831655   1.4757305825   1.4906788617   1.4637107604   1.4629094390
 1.4697122764   1.4627234147   1.4626575916   1.4626535940
 1.4644203101   1.4626563214   1.4626518485   1.4626517573   1.4626517501
 1.4630941026   1.4626520334   1.4626517476   1.4626517460   1.4626517459   1.4626517459
 1.4627623486   1.4626517639   1.4626517459   1.4626517459   1.4626517459   1.4626517459   1.4626517459

 result problem 1=  1.4626517459
 ===============================
 1.0000000000   0.8535533906   0.8047378541   0.8050352528   0.7888625402   0.7878041860
 0.7898353948   0.7847687754   0.7844958578   0.7844433446
 0.7852756861   0.7837557832   0.7836882504   0.7836754312   0.7836724198
 0.7839511939   0.7835096966   0.7834932908   0.7834901962   0.7834894698   0.7834892909
 0.7835752392   0.7834499209   0.7834459359   0.7834451842   0.7834450077   0.7834449642   0.7834449534
 0.7834702854   0.7834353008   0.7834343261   0.7834341418   0.7834340985   0.7834340878   0.7834340852   0.7834340845
 0.7834413454   0.7834316987   0.7834314586   0.7834314131   0.7834314024   0.7834313997   0.7834313991   0.7834313989
 0.7834334411   0.7834308063   0.7834307468   0.7834307355   0.7834307329   0.7834307322   0.7834307320   0.7834307320
 0.7834312986   0.7834305844   0.7834305696   0.7834305668   0.7834305661   0.7834305660   0.7834305659   0.7834305659
 0.7834307215   0.7834305291   0.7834305254   0.7834305247   0.7834305246   0.7834305245   0.7834305245   0.7834305245
 0.7834305668   0.7834305153   0.7834305144   0.7834305142   0.7834305142   0.7834305142   0.7834305142   0.7834305142

 result problem 2=  0.7834305142
 ===============================
-0.1311874269 -0.1317695885  -0.1319636423  -0.1319149685  -0.1319634285  -0.1319634142
-0.1319513035  -0.1319634151  -0.1319634143  -0.1319634143
 0.0000090831   0.0439958787   0.0557264982   0.0587057032   0.0594534252
 0.0005519662   0.0007329272  -0.0021512695  -0.0030699643  -0.0033122218  -0.0033735763
 0.0006646731   0.0007022421   0.0007001965   0.0007454578   0.0007604203   0.0007644013   0.0007654118
 0.0006917471   0.0007007717   0.0007006737   0.0007006813   0.0007005057   0.0007004471   0.0007004315   0.0007004275
 0.0006984506   0.0007006851   0.0007006793   0.0007006794   0.0007006794   0.0007006796   0.0007006797   0.0007006797
 0.0007001225   0.0007006798   0.0007006794   0.0007006794   0.0007006794   0.0007006794   0.0007006794   0.0007006794
 0.0007005402   0.0007006795   0.0007006794   0.0007006794   0.0007006794   0.0007006794   0.0007006794   0.0007006794

 result problem 3=  0.0007006794
 ===============================
 10.0000000000 10.0000000000  10.0000000000
 result problem 4= 13.0258509299
 ===============================
 1.3591409142   0.9065562270   0.7556946645   0.7715289249   0.7265198243   0.7245748349
 0.7356981803   0.7237545987   0.7235702503   0.7235543046
 0.7265665518   0.7235226756   0.7235072141   0.7235062135   0.7235060249
 0.7242670291   0.7235005216   0.7234990446   0.7234989150   0.7234988863   0.7234988794
 0.7236901351   0.7234978371   0.7234976581   0.7234976361   0.7234976311   0.7234976299   0.7234976296
 0.7235456142   0.7234974406   0.7234974141   0.7234974103   0.7234974094   0.7234974092   0.7234974091   0.7234974091
 0.7235094351   0.7234973754   0.7234973710   0.7234973703   0.7234973702   0.7234973702   0.7234973701   0.7234973701
 0.7235003819   0.7234973642   0.7234973634   0.7234973633   0.7234973633   0.7234973633   0.7234973633   0.7234973633
 0.7234981171   0.7234973622   0.7234973621   0.7234973620   0.7234973620   0.7234973620   0.7234973620   0.7234973620
 0.7234975507   0.7234973619   0.7234973618   0.7234973618   0.7234973618   0.7234973618   0.7234973618   0.7234973618
 0.7234974090   0.7234973618   0.7234973618   0.7234973618   0.7234973618   0.7234973618   0.7234973618   0.7234973618

 result problem 5=  2.5425742098
 ===============================
 0.1015006085   0.2804899753   0.3401530975   0.4135232782   0.4578677125   0.4657153535
 0.4910381645   0.5168764599   0.5208103764   0.5216849006
 0.5294978311   0.5423177200   0.5440138040   0.5443821124   0.5444711211
 0.5472423873   0.5531572393   0.5538798740   0.5540364783   0.5540743385   0.5540837258
 0.5551790314   0.5578245794   0.5581357354   0.5582032888   0.5582196292   0.5582236813   0.5582246923
 0.5586791270   0.5598458256   0.5599805753   0.5600098585   0.5600169431   0.5600187000   0.5600191383   0.5600192479
 0.5602123788   0.5607234627   0.5607819719   0.5607946924   0.5607977702   0.5607985335   0.5607987239   0.5607987715
 0.5608818563   0.5611050155   0.5611304524   0.5611359838   0.5611373222   0.5611376541   0.5611377369   0.5611377576
 0.5611737116   0.5612709967   0.5612820621   0.5612844686   0.5612850509   0.5612851953   0.5612852314   0.5612852404
 0.5613008450   0.5613432227   0.5613480378   0.5613490850   0.5613493384   0.5613494013   0.5613494170   0.5613494209
 0.5613562034   0.5613746563   0.5613767518   0.5613772076   0.5613773179   0.5613773453   0.5613773521   0.5613773538

 result problem 6=  0.5613773544
 ===============================
```