Program-M11

Devansh Shukla
I18PH021

a) Solve the following integration analytically.

$$\int_0^1 dx \ \left(x^2 + 4x + 9\right)^3 \ (2x + 4)$$

b) Write and execute a Fortran program to solve the above problem by Monte Carlo hit-or-miss integration method and generate a merged plot for darts of hit and miss(with different colours) and the function. Compare your analytical and programmed integration results.

1 Theory

A Monte Carlo solution to this problem via the 'hit-or-miss' method is to draw a box extending from a to b and from 0 to y_0 where $yo > f(x)$ throughout this interval. Using random numbers drawn from a uniform distribution, we drop N points randomly into the box and count the number, N_0, which fall below $f(x)$ for each value of x.
An estimate for the integral is then given by the fraction of points which fall below the curve times the area of the box, i.e

$$y = y_0(b - a)\frac{N_0}{N} \tag{1}$$

This estimate becomes increasingly precise as $N \to \infty$ and will eventually converge to the correct answer. This technique is an example of a 'simple sampling' Monte Carlo method and is obviously dependent upon the quality of the random number sequence which is used.

2 Numerical Solution

$$I = \int_0^1 dx \ \left(x^2 + 4x + 9\right)^3 \ (2x + 4)$$

$$\text{Let,} \quad t = x^2 + 4x + 9$$
$$dt = (2x + 4)dx$$

Integrating from x=0 to x=1 $\implies$ t=9 to t=14

$$I = \int_9^{14} dt \ t^3$$

$$I = \left[\frac{t^4}{4}\right]_9^{14}$$

$$\boxed{I = 7963.75}$$

3 Program Algorithm

NOTE: Blue-colored text represents variables in the algorithm, eg. variable.

1. Program open.
2. Define x, y, a, b, max, min, step, t, i, r, f, area, int_area, j, hits, n, fmt.
3. Open a file "int_area.dat" with write access.
4. Define the function f(x).
5. Get the user input for lower limit (a), upper limit (b), steps size (step) and max darts (n)
6. Set i = a, max = f(i) and min = f(i)
7. Open a do-while loop for index (i) till upper limit (b)
8. Compute t = f(i)
9. If the value of t > max, then max = t
10. If the value of t < min, then min = t
11. Increment the index (i) by the step size (step)

12. End-do-while loop.
13. Set j = 0.0
14. Open a do-while loop for index (j) till the total no of darts (n).
15. Compute a random number (r) and calculate the value of x, such that $x = a + (b - a) * r$
16. Compute a random number (r) and calculate the value of y, such that $y = y_{max}r$
17. If y $\leq$ f(x), then increment hits by 1.
18. Increment the index(j) by 1.
19. End-do-while loop.
20. Compute the integration area.
21. Print the no. of hits and the integration area for the user.
22. Close file.
23. Program close.

4 Program

4.1 Fortran program:

For computing the parameters

```
!==========================================
! integration.f90
! Author: Devansh Shukla
!------------------------------------------
program integration
    implicit none
    ! Define the variables
    real*8 :: x, y, a=0.0, b=0.0, max=0.0, min=0.0, step=0.1, t=0, i=0.0, r=0.0
    real*8 :: f, area, int_area, j
    integer :: hits=0, n=0
    character(len=*), parameter :: fmt="(xF10.3,xF10.3,xF10.3)"

    ! Define the function
    f(x) = (x**2 + 4*x + 9)**3 * (2*x + 4)

    ! Open the data file
    open(unit=8, file="int_area.dat")

    ! Get input from the user
    print *, "Enter a, b"
    read *, a, b
    print *, "Enter step, n"
    read *, step, n
    print *, "-------------------"

    ! Max of f(x) in [a, b]
    i = a
    max = f(i)
    min = f(i)
    do while (i .le. b)
        t = f(i)
        if (max .lt. t) then
            max = t
        endif
        if (min .gt. t) then
            min = t
        endif
        i = i + step
    enddo
    print "(A,F10.3,xxA,F10.3)", "MIN:", min, "MAX:", max
    j = 0.0
    ! Compute the integration
    do while(j .le. n)
        call random_number(r)
        x = a + (b-a)*r
        call random_number(r)
        y = abs(max*r)
        ! print "(F10.3, xxF10.3)", y, f(x)
        write (8, fmt) x, y, f(x)
        if (y .le. abs(f(x))) then
                hits = hits + 1
        endif
    j = j + step
    enddo
```

```
    area = (b-a) * max
    int_area = area * hits * step / real(n)
    ! Print for the user
    print "(A,I6)", "HITS=", hits
    print "(A,F10.3)", "INTEGRATION AREA=", int_area
    print *, "-------------------"

    ! Close the data file
    close(8)

end program integration
```

4.2 Python program: Plots

```
#!/usr/bin/env python
"""
Author: Devansh Shukla
"""
# In[0]
import pandas as pd
import numpy as np
import matplotlib as mpl
import matplotlib.pyplot as plt
import matplotlib.gridspec as gridspec

plt.style.use("rcStyleSheet.mplstyle")
# mpl.use("pgf")
# plt.ioff()

# In[2]
x = np.linspace(0, 1, 1000)
func = (x**2 + 4*x + 9)**3 * (2*x + 4)

for n in [1000, 2000, 5000, 10000]:
    df = pd.read_csv(f"int_area_{n}.dat", engine="python", delimiter=" ", header=None, skipinitialspace=True, comment="#
        ")
    missed_darts_x = []
    missed_darts_y = []
    hit_darts_x = []
    hit_darts_y = []

    for i in range(0, len(df[0])):
        if df[1][i] <= df[2][i]:
            hit_darts_x.append(df[0][i])
            hit_darts_y.append(df[1][i])
        else:
            missed_darts_x.append(df[0][i])
            missed_darts_y.append(df[1][i])

    fig = plt.figure(figsize=(6,4))
    gs = gridspec.GridSpec(1, 1)
    ax = fig.add_subplot(gs[0, 0])
    ax.plot(hit_darts_x, hit_darts_y, "x", markersize=3, color="C0", label=r"Hit")
    ax.plot(missed_darts_x, missed_darts_y, "x", markersize=3, color="C1", label=r"Missed")
    ax.plot(x, func, "-", linewidth=2, markersize=2, color="C5", label=r"$f(x)$")
    ax.set_xlim(left=0, right=1.1)
    ax.set_ylim(0, 18000)
    ax.set_xlabel(r"$x$")
    ax.set_ylabel(r"$y$")
    ax.legend(loc="lower right")
    # plt.suptitle("Integration--Hit or miss")
    plt.title(f"Integration--Hit or miss (n={n})")
    plt.savefig(f"outputs/int_{n}.pdf")

# %%
```

5 Results

5.1 Terminal Output

5.1.1 n = 1000

```
 Enter a, b
0 1
 Enter step, n
1 1000
 -------------------
```

```
MIN:  2916.000  MAX: 16464.000
HITS=   486
INTEGRATION AREA=  8001.504
 -------------------
```

5.1.2 n = 2000

```
 Enter a, b
0 1
 Enter step, n
1 2000
 -------------------
MIN:  2916.000  MAX: 16464.000
HITS=   977
INTEGRATION AREA=  8042.664
 -------------------
```

5.1.3 n = 5000

```
 Enter a, b
0 1
 Enter step, n
1 5000
 -------------------
MIN:  2916.000  MAX: 16464.000
HITS=  2412
INTEGRATION AREA=  7942.234
 -------------------
```

5.1.4 n = 10000

```
 Enter a, b
0 1
 Enter step, n
1 10000
 -------------------
MIN:  2916.000  MAX: 16464.000
HITS=  4786
INTEGRATION AREA=  7879.670
 -------------------
```

5.2 Data file

First 15 lines from the data files:

5.2.1 n=1000

```
      0.692   1439.580   9896.093
      0.687  16434.379   9802.525
      0.270   5772.160   4747.724
      0.921  12487.015  14467.349
      0.080   5672.326   3373.902
      0.521   3385.990   7377.755
      0.881  13813.405  13549.712
      0.798  12020.211  11820.175
      0.177   6165.399   4024.906
      0.588   3588.527   8289.113
      0.811   6898.841  12068.639
      0.076   2910.669   3347.432
      0.028   3270.688   3069.353
      0.979    396.588  15913.615
      0.027  13995.890   3061.298
```

5.2.2 n=2000

```
      0.227  15022.729   4400.755
      0.662   2214.450   9400.041
      0.028   4167.096   3070.868
      0.649  13930.334   9196.467
      0.268  13818.178   4732.205
      0.367   8244.030   5644.085
      0.005   1341.379   2944.564
      0.821  13455.542  12272.351
      0.520  12970.420   7374.287
      0.272   8009.497   4766.538
      0.790   4929.982  11654.809
      0.164   8200.403   3932.176
      0.009  12358.934   2964.042
      0.438   5832.690   6387.523
      0.548  12044.597   7736.247
```

5.2.3 n=5000

```
0.885   8533.953  13650.801
0.198  15057.940   4177.385
0.613   5848.596   8642.460
0.616  10505.039   8693.425
0.349   9809.356   5464.368
0.965   9768.904  15560.457
0.405  14756.200   6029.395
0.554   1841.258   7816.494
0.794    327.680  11730.961
0.763  15326.246  11142.031
0.753  11329.285  10957.114
0.355   9594.986   5528.884
0.961  10534.360  15456.058
0.757   2724.988  11041.800
0.396   4720.083   5938.949
```

5.2.4 n=10000

```
0.624   7582.257   8809.351
0.023  14280.443   3040.532
0.023  14118.389   3039.229
0.262   1537.884   4688.192
0.378   7042.783   5750.841
0.268   4812.257   4738.644
0.017   2003.143   3008.568
0.900   4786.821  13984.678
0.777  15236.311  11409.511
0.355  12492.522   5519.944
0.270   7058.562   4750.357
0.372  10010.884   5689.219
0.408   5209.458   6064.748
0.001  15434.466   2918.727
0.521  13581.180   7383.239
```

5.3 Plots

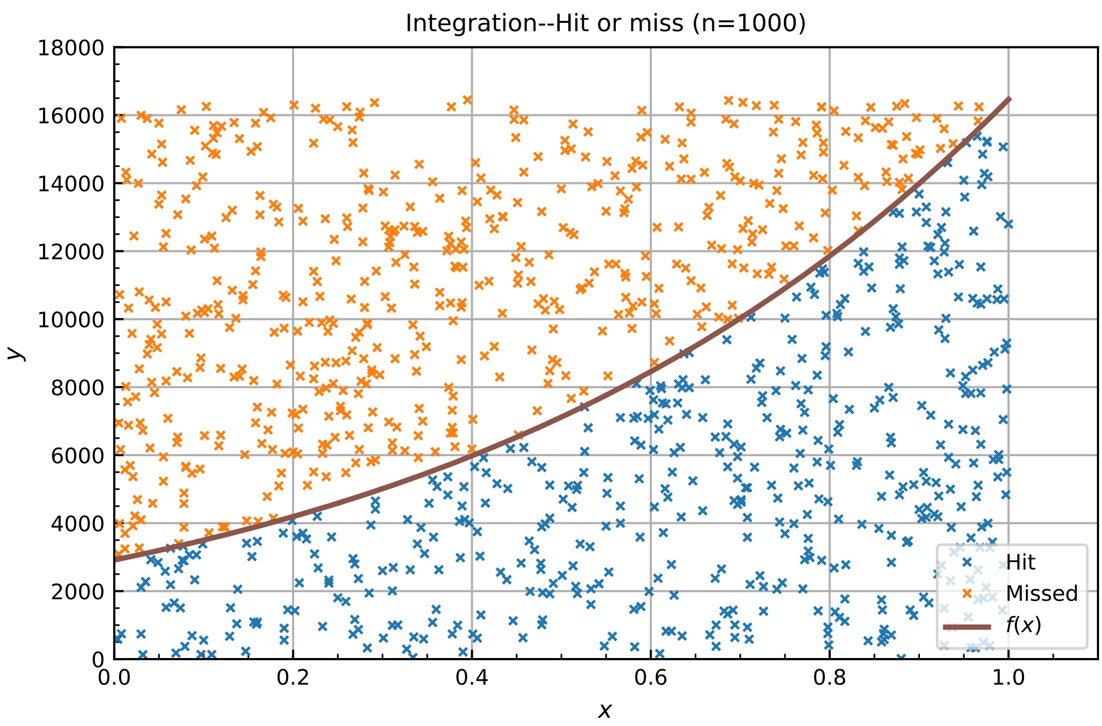

Figure 1: n=1000

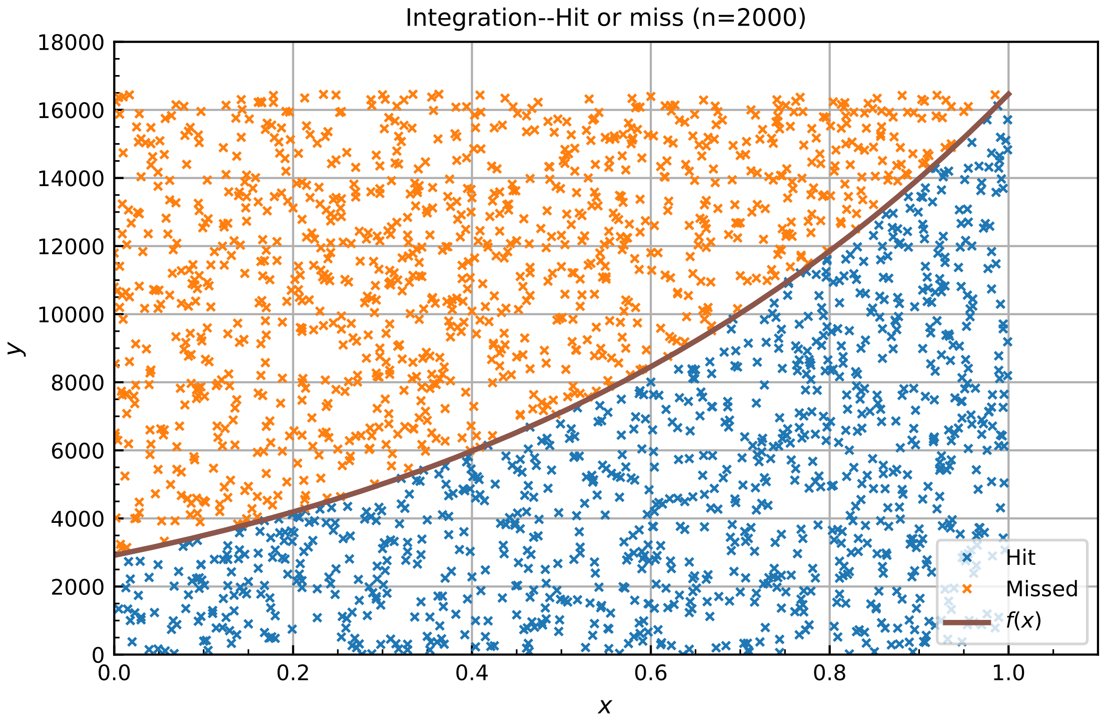

Figure 2: n=2000

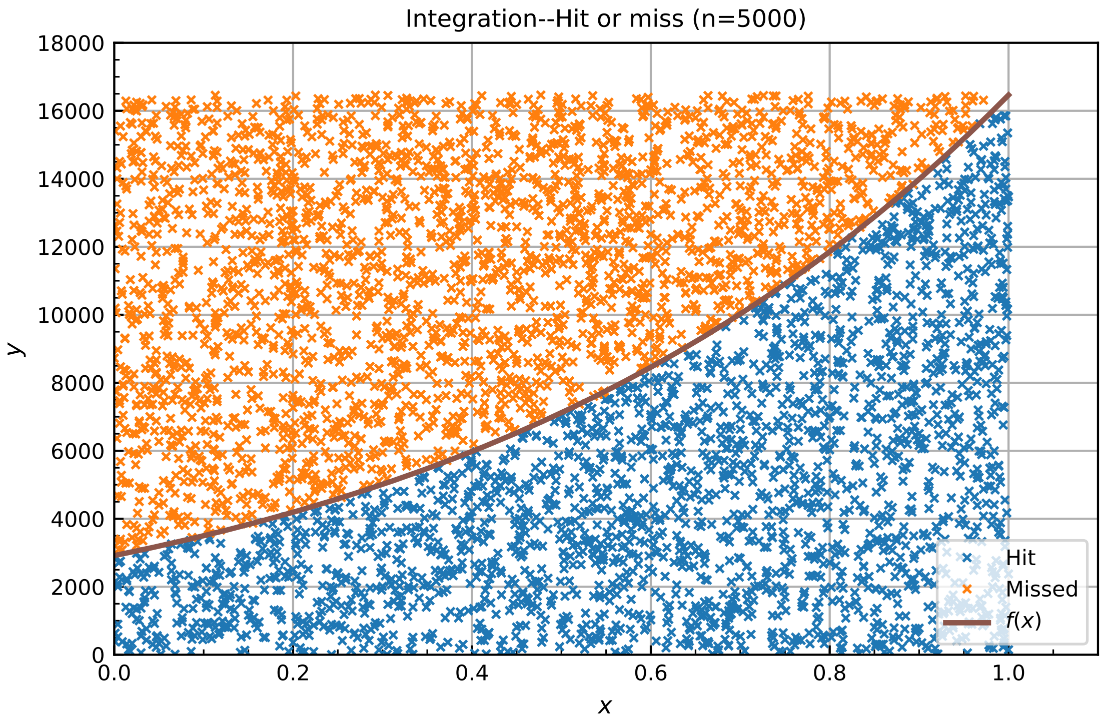

Figure 3: n=5000

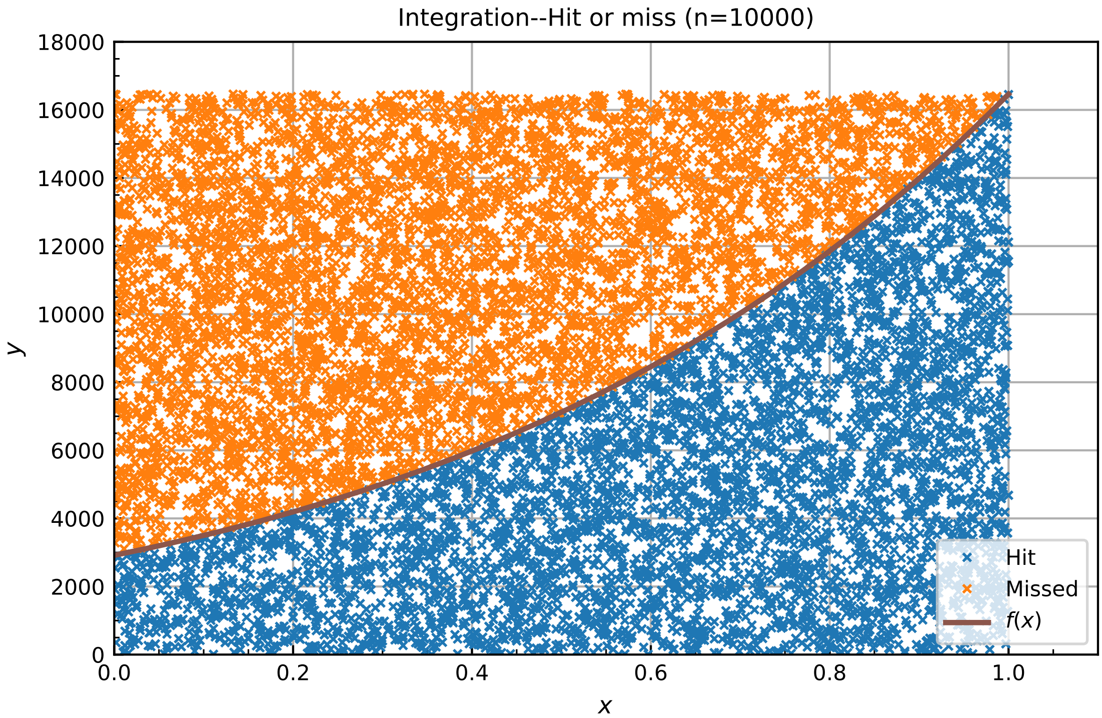

Figure 4: n=10000

6 Remarks

The programs can be used to numerically compute the integral, its especially useful in cases when the integral is difficult to compute analytically.

The parameters computed numerically and via the programs are within reasonable limits.

$$\begin{aligned} \%\text{ error for n=5000} &= \frac{|7963.750\ -\ 7942.234|}{7963.750} \\ &= 0.27\% \end{aligned} \tag{2}$$