PHYS 551 Midterm Project
Due: November 2, 2000

(Counts as 5 ordinary HW assignments

Write short programs to perform the following chores:

1. Use 5-point Gauss-Hermite integration (look up the points and weights in Abramowitz & Stegun) to evaluate the following integrals:

   (a) $\int_{-\infty}^{\infty} dx\, e^{-x^2} \cos(x)$

   (b) $\int_{-\infty}^{\infty} dx\, e^{-x^2} \cos(2x)$

   (c) $\int_{-\infty}^{\infty} dx\, e^{-x^2} \cos(3x)$

   (d) $\int_{-\infty}^{\infty} dx\, e^{-x^2} \cos^2 x$

   (e) $\int_{-\infty}^{\infty} dx\, e^{-x^2} (x^4 - 2x^2 + 1)$

   Compare the results with the exact answers and discuss the loss of precision (if any!). (Look them up in tables if you do not know how to evaluate these integrals, or those in Question 2 below.)

2. Use 5-point Gauss-Laguerre integration to evaluate the integrals

   (a) $\int_0^{\infty} dx\, e^{-x} \cos(x)$

   (b) $\int_0^{\infty} dx\, e^{-x} \cos(2x)$

   (c) $\int_0^{\infty} dx\, e^{-x} \sin(x)$

   (d) $\int_0^{\infty} dx\, e^{-x} (x^{10} - 2x^5 + 1)$

   and compare with the exact results.

3. Write an adaptive quadrature subroutine base on Simpson's Rule. The program should keep all evaluated function points until the segment of the integral that uses them has converged and been stored. Test the routine on some of the above integrals, as well as on the integral

$$\int_{-1}^{1} dx \sqrt{1 - x^2}\,.$$

4. Use the Brute Force Monte-Carlo method to evaluate the integral

$$\int_0^1 dx\, x^2 \left(1 - x^2\right) .$$

   Estimate the uncertainty of the result.

5. Use the regular Monte-Carlo method to evaluate the same integral as above. Estimate the uncertainty of the result.

6. Write a program to simulate a drunkard walking on an infinite two-dimensional grid of square city blocks. Every time he comes to a street corner he forgets where he is and makes a random turn down one of the (4) available streets.

   (a) Tabulate his distance from his starting point, as a function of N, the number of corners he has reached (and gotten confused at). Run the program enough times to get reasonable statistics.

   (b) Modify the program to count how many times his path crosses itself, as a function of N. Again run the program enough times to get reasonable statistics.

   Be sure to estimate the precision of your results for (a) and (b) above. Also be sure not to start the random number generator with the same seed on each run!