University of Toronto
Department of Computer Science
CSC 270 - Fall 1994

Assignment 1: Adaptive Quadrature

Due: in tutorial, October 17

You are to write a program which performs Numerical Integration. Your program will provide an approximation to the integral:

$$I = \int_a^b f(x)\,dx \tag{1}$$

Part I

For the first part of this assignment, your program should provide an estimate of the integral using a straightforward application of the trapezoidal formula:

$$I \approx (f(x_0)/2 + f(x_1) + f(x_2) + \cdots + f(x_{n-1}) + f(x_n)/2)\Delta x, \tag{2}$$

where $\Delta x = x_i - x_{i-1}$ for $i = 1, 2, \cdots, n;\ x_0 = a,\ x_n = b$

in which the user specifies the number of subintervals, n.

Part II

To improve the approximation of the integral, you should now use the Adaptive Quadrature method described in class. With this method, the interval of integration is initially divided into equal-sized subintervals. For each subinterval $[x_{i-1}, x_i]$, an approximation to the integral is calculated as:

$$a_i = (f(x_i) + 2f((x_{i-1} + x_i)/2) + f(x_i))(x_i - x_{i-1})/4, \tag{3}$$

and an estimation of the error as:

$$e_i = (-f(x_{i-1}) + 2f((x_{i-1} + x_i)/2) - f(x_i))(x_i - x_{i-1})/12. \tag{4}$$

The program should then repeatedly select the subinterval with the largest absolute value of error, $|e_i|$ and divide this subinterval, adjusting the total estimate of the integral and its associated error, as appropriate. The process stops when either the estimate of total error is less than some *tolerance*, or the number of subintervals exceeds *max_intervals*.

Your program should be supplied with the following input:

- $f(x)$, the function to be integrated
- $[a, b]$, the bounds of the interval of integration

- *tolerance*, the maximum error considered acceptable
- *init_intervals*, the initial number of intervals
- *max_intervals*, an upper bound on the number of intervals allowed

and provide the following as output:

- *approx*, the approximation of the integral, above
- *errest*, an estimate of the error
- *numints*, the final number of intervals used by the program

Your program should indicate whether it achieved a satisfactory error, or if it stopped because *max_intervals* was exceeded.

For efficiency, your program should keep the necessary information for each subinterval in a **heap** data structure, using $|e_i|$ as the key so that the record whose $|e_i|$ is largest is at the top of the heap. Once this subinterval is divided and its record replaced by two new ones, the heap property must be restored.

Part III

Finally, to demonstrate the effectiveness of Adaptive Quadrature, your program should compare the approximation of the integral obtained by part I and part II, using an equal number of subintervals. In other words, n of part I must be equal to *numints* of part II.

Remember that testing is a very important component of your solution. You should provide test results for the following cases:

- definite integrals that can be computed exactly over the given interval
- definite integrals that cannot be computed exactly over the given interval
- $f_1(x) = 1/(1 + 100x^4)$ over $[0, 1]$
- $f_2(x) = x^{2.5}$ over $[0, 1]$
- $f_3(x) = 250e^{-250x}$ over $[0, 1]$
- $f_4(x) = 500/\pi(1 + 25000x^2)$ over $[0, 1]$

For each integrand, the program should be tested with several values of *tolerance*. Your results should be illustrated by graphs that show the relationship between *tolerance* and *numints*.