MAT128a: Numerical Analysis, Fall 2015
Programming Project Three
Due: November 20, 2015

In this project, you will write three MATLAB functions which perform tasks involving numerical integration. There is a template file available on the website which defines each of these functions. Follow these templates exactly — we will grade the project by calling your functions and testing to see that they perform as expected.

Each of the functions you write should be placed in an ".m" file whose name is the same as the name of the function. You will submit your project by sending an email to the following address:

`mat128a_fall2015@math.ucdavis.edu`

Please send one email with three attachments, one for each of the ".m" files which comprise this project. You will get a reply with your score within two weeks of submission (probably much sooner). Please try to avoid making multiple submissions — wait to submit your project until you are completely satisfied with it. If you do submit more than once, only the last submission will be graded. Moreover, only submissions *received* before 11:59 PM on the due date will be considered.

Project description

1. Write a MATLAB function called "clenshaw" which takes as input a positive integer n and returns the nodes and weights of the (n+1) point Clenshaw-Curtis quadrature rule we derived in class. That is the quadrature rule with nodes

$$x_k = \cos\left(\frac{2k+1}{2n+1}\pi\right), \quad k = 0, 1, \ldots, n, \tag{1}$$

and weights $w_0, \ldots, w_n$ chosen so that the formula

$$\int_{-1}^{1} f(x)\,dx = \sum_{k=0}^{n} f(x_k) w_k \tag{2}$$

is exact for polynomials of degree less than or equal to n.

2. Write a MATLAB function called "oneint" which takes as an input real numbers a and b such that $a < b$, a function defined on $[a, b]$, and a nonnegative integer n and returns as output the sum

$$\left(\frac{b-a}{2}\right) \sum_{k=0}^{n} f\left(\frac{b-a}{2}x_k + \frac{b+a}{2}\right) w_k, \tag{3}$$

where $x_0, \ldots, x_n$ and $w_0, \ldots, w_n$ are the nodes and weights, respectively, of the $(n+1)$-point

Clenshaw-Curtis quadrature on the interval $[a, b]$. Of course, this sum approximates the integral

$$\int_a^b f(x)\,dx. \tag{4}$$

3. Write a MATLAB functions called "adapint" which takes as input real numbers a and b such that $a < b$, a real number $\epsilon > 0$, and a function f defined on the interval $[a, b]$ and returns as ouput the approximation of the integral

$$\int_a^b f(x)\,dx \tag{5}$$

obtained using the standard adaptive integration procedure described in class. In particular, recursively subdivide the interval $[a, b]$ into a collection of intervals

$$[a_i, b_i] \tag{6}$$

such that the approximations of the integral

$$\int_{a_i}^{b_i} f(x)dx \tag{7}$$

obtained by using the $(n+1)$-point Clenshaw-Curtis rule on the interval $[a_i, b_i]$ and by applying the same quadrature rule to each of the intervals

$$\left[a, \frac{a+b}{2}\right], \left[\frac{a+b}{2}, b\right] \tag{8}$$

differ by less than ϵ is magnitude. Then return the approximation of (5) obtained by using the $(n+1)$-point Clenshaw-Curtis rule to approximate each of the integrals

$$\int_{a_i}^{b_i} f(x)dx. \tag{9}$$