

CISC271
Fall 2006
Homework for week 6
in preparation for quiz 3
Solutions

This homework is about interpolating polynomials, with the monomial, Lagrange and Newton bases.

Recktenwald Chapter 10. questions 8, 9 , 11, and 18.

10-8 The Lagrange basis functions for the cubic interpolating polynomial passing through $(x_j, y_j), (x_{j+1}, y_{j+1}), (x_{j+2}, y_{j+2}), (x_{j+3}, y_{j+3})$, can be concisely written as:

$$L_{j+k} = \prod_{i=0, i \neq k}^3 \frac{x - x_{j+i}}{x_{j+k} - x_{j+i}} \text{ for } k = 0, 1, 2, 3$$

10-9 The denominator for the Lagrange polynomial L_j is

$$L_j = \prod_{i=1, i \neq j}^n x_j - x_i$$

Thus with n Lagrange polynomials and $n - 1$ terms in the denominator it appears that we have $n^2 - n$ terms. This can be viewed as all terms in an $n \times n$ matrix except those on the main diagonal, that is, when $i = j$. We can get away with storing half of these values by observing that $x_j - x_i = (-1)x_i - x_j$. Thus we can compute and store half the terms and simply do a sign change for the others. As is mentioned in the text there is little practical benefit from doing this.

10-11 Here is a table with the results of my hand calculations. Are your's the same?

x_i	$y_i = f[x_i]$	$f[x_i, x_{i+1}]$	$f[x_1, x_2, x_3]$
10	3.810		
20	1.492	-0.2318	
30	0.629	-0.08	0.0076

Table 1: Newton's divided differences.

10-18 Here is my m-file for quadinterp

```
function yhat = quadinterpDR(x,y,xhat)
% Solves question 10-18 from Recktenwald. Given
% newtint Interpolation with Newton polynomials of arbitrary degree
%
% Synopsis: yhat = quadinterpDR(x,y,xhat)
%
% Input: x,y = vectors containing the tabulated y = f(x) data
% xhat = scalar x value where interpolant is evaluated
%
% Output: yhat = value of y at xhat obtained via interpolation
%
% Note: Degree of interpolating polynomial is explicitly quadratic
% Find best subset of samples by locating x using binSearch.
% Then calls newtint for actual interpolation.
%
% test data is
% x = [ 0 10 20 30 40 50 ]'
% y = [10.60 3.81 1.492 .629 .2754 .1867]'
```

```
n = length(y); if length(x)~=n, error('x and y are not compatible'); end

%Never search past x(n-2)
if xhat >= x(n-2) ibeg = n-2; else ibeg = binSearch(x,xhat); end;
yhat = newtint(x(ibeg:ibeg+2),y(ibeg:ibeg+2),xhat);
```

After I created the vectors x and y This is how I generated data for plotting.

```
yhat= zeros(100,1);xhat = linspace(0,50);
for i = 1:100 yhat(i) = QuadInterpDR(x,y,xhat(i)); end
```

And here is the plot I got.

