1. The error in Hermite interpolation is given by

\[ f(x) - H_{2n}(x) = \frac{f^{(2n)}(z)}{(2n)!} \psi(x)^2, \]

where \( \psi(x) = (x - x_1)(x - x_2) \cdots (x - x_n) \). The proof that is given in Atkinson (pages 160-161) uses a limit argument of points pairwise coinciding, and properties of the expressions \( f[x_1, x_2, \ldots, x_n] \) from Newton's interpolation formula. Maybe this can be viewed as 'elegant', but it is definitely not the most straightforward approach. We can in fact proceed almost identically to what we did for 'plain' polynomial interpolation. Thus, choose some arbitrary location \( t \) and verify that (1) holds at this location. For this purpose, consider the function

\[ G(x) = f(x) - H_{2n}(x) - R \psi(x)^2 \]

and choose the factor \( R = R(t) \) so that \( G(t) = 0 \), etc. Complete this argument to arrive at (1).

2. On the notes "Spline illustrations" on the class web page, the decay rates for cardinal cubic- and quadratic splines on equispaced grids were given as \( (2 - \sqrt{3})^k \) and 1 (no decay) respectively. Demonstrate these results theoretically.

Hint: No great rigor is required, but give plausible arguments that lead to these specific factors. Useful ingredients for the cubic spline case are (i) The linear system that is used for determining spline interpolants, and (ii) The notes "Linear recursion relations" (also posted on the class web site). With regard to the quadratic case, there is a very brief (2-line) argument that leads to the desired result.

3. When using \( B \)-splines, one often needs to calculate the value of these basis functions at certain locations \( z \). There turns out to exist a delightfully simple algorithm that achieves this for \( B \)-splines of any order (i.e. not just cubic ones). It goes as follows (in the cubic case; say the nodes are at \( x_1, x_2, x_3, x_4, x_5 \)):

i. Write down the five \( x \)-coordinates for the \( B \)-spline in a first column,
ii. Check in which subinterval \( z \) falls; say between \( x_i \) and \( x_{i+1} \), then enter \( B_i^0 = 1/(x_{i+1} - x_i) \) while leaving the other \( B_i^0 \)-entries zero,
iii. Fill in the rest of the \( B \)-entries recursively, very much like the Newton divided difference table:

\[ B_k^p(z) = \frac{(z - x_p)B_p^{p-1}(z) + (x_{p+k+1} - z)B_{p+k}^{k-1}(z)}{x_{p+k+1} - x_p} \]

Graphically, in the cubic case and assuming the \( z \)-value fell in the second subinterval, the chart will look like

\[
\begin{align*}
x_1 &= \ldots & B_1^0 &= 0 \\
x_2 &= \ldots & B_2^0 &= \frac{1}{x_3 - x_2} B_1^1 &= \ldots \\
x_3 &= \ldots & B_3^0 &= 0 & B_1^2 &= \ldots \\
x_4 &= \ldots & B_4^0 &= 0 & B_2^2 &= \ldots \\
x_5 &= \ldots & B_5^0 &= 0 & B_3^2 &= \ldots \\
\end{align*}
\]
The value of the $B$-spline at $z$ will be the rightmost entry in this table (however, with a normalization that may not match the standard custom of integral evaluating to one).

By means of this approach, produce a graph of the cubic $B$-spline based on the five nodes $x = [-2 -1 0 3 6]$. On the curve, be sure to mark the node values.

Notes (not needed for solving the homework problem):

i. For $B$-splines of higher order, we simply need more start $x$-values and then run out the chart beyond level 3 to whatever the $B$-spline order was.

ii. For each $z$-value, several $B$-splines are non-zero. Their values at this location $z$ can be obtained from a single table as above by again extending to more $x$-values (but now not needing to run it out further to the right).