Successive parabolic interpolation

1. Starting with the points 4, 4.5, 5, approximate a minimum of $f(x) \stackrel{\text{def}}{=} \frac{\sin(x)}{x} + e^{-x}$ with three steps of the method of successive parabolic interpolation.

Answer: The new value (to 10 significant digits) is indicated in bold, and the value rejected is the one with the largest value.

4.0, 4.5, 5.0

For this first step, we can either use the formula in the slides (the formula in the slides tries to mitigate the effects of subtractive cancellation), or more easily just find the interpolating polynomial:

```
f = @(x)( sin(x)/x + exp(-x) );
% Set up the Vandermonde matrix
p = [4.0^2 4.0 1; 4.5^2 4.5 1; 5.0^2 5.0 1] \ [f(4.0) f(4.5) f(5.0)]'
     p =
        0.1126158882502058
       -1.027704917247146
        2.138080472047045
-p(2)/(2*p(1))   % calculate -b/2a
     ans = 4.562877109151018
```

4.5, **4.562877109**, 5.0

4.5, **4.544108782**, 4.562877109

4.543030477, 4.544108782, 4.562877109

2. The actual minimum is at x = 4.542956187. How does the error in the x-value and the error in the f-value differ with the approximation x = 4.543030477?

Answer: The error in the x-value is 0.00007429, but the f(4.543030477) = –0.2063270787440699 and the value of the function at the actual minimum is –0.2063270793592269, and the error here is only 0.0000000006152. Thus, we do not need to be as close to a minimum to actually have an accurate approximation as to what the minimum is.

Note, normally we only display 10 digits as we are trying to convey the idea behind the algorithms, as opposed to an accounting of each digit; however, in this case, we require more digits to display the accuracy of the algorithm.

3. Starting with the points –2, –1.75, –1.5, approximate a minimum of $f(x) \overset{\text{def}}{=} x^4 - 6x^2 + 4x + 4$ with three steps of the method of successive parabolic interpolation.

Answer: The new value is indicated in bold, and the value rejected is the one with the largest value.

–2, –1.75, –1.5

For this first step, we can either use the formula in the slides (the formula in the slides tries to mitigate the effects of subtractive cancellation), or more easily just find the interpolating polynomial:

```
f = [1 0 -6 4 4]; % the polynomial x^4 - 6x^2 + 4x + 4
xs = [-2 -1.75 -1.5]';
p = vander( xs ) \ polyval( f, xs )
     p =
           12.43750000000002
           46.65625000000006
           31.56250000000006
-p(2)/(2*p(1))   % calculate -b/2a
     ans = -1.875628140703518
```

–2.0, –**1.875628141**, –1.75

–2.0, **–1.875516902**, –1.875628141

–1.875516902, –1.875628141, **–1.879170084**

4. The actual minimum is at x = –1.879385242. How does the error in the x-value and the error in the f-value differ with the approximation x = –1.881966011?

Answer: The error in the x-value is 0.0002151, but the f (–1.879170084) = –12.234421680201402733 and the value of the function at the actual minimum is –12.234422383429318516, and the error here is only 0.0000007032.

5. Implement a function in C++ that calculates the minimum using successive parabolic interpolation:

```
// Assume that the values are sorted so that
// x3 is the best approximation of the minimum
//   I.e., so f(x3) <= f(x2) <= f(x1)
std::pair<double, double> successive_parabolic_interpolation(
    std::function< double(double) > f,
    double x1, double x2, double x3
) {
    // Your implementation here, but not required for an examination
}
```

Answer:

```
std::pair<double, double> successive_parabolic_interpolation(
    std::function< double(double) > f,
    double x1, double x2, double x3
) {
    double f1{ f(x1) };
    double f2{ f(x2) };
    double f3{ f(x3) };

    assert( (f3 <= f2) && (f2 <= f1) );

    double x4{ 0.5*(
        x3 + x2 + (
            (f3 - f2)*(x2 - x1)*(x1 - x3)
        )/(
            (f3 - f2)*(x2 - x3) + (f1 - f2)*(x3 - x2)
        )
    ) };

    return std::make_pair( x4, f(x4) );
}
```