Brent's Method

input : function f, bracket $[a, b]$ with $f(a) \cdot f(b) < 0$, tolerance ε_s
output: estimate b of the root

```
// b is the best estimate, c is on the other side of the root, a
   is the previous b
c ← a
// d is the signed step b will move by, e is the previous step;
   both start as the bracket width
d ← b − a;   e ← d
repeat
    if f(b) · f(c) > 0 then
        // b and c no longer bracket the root; a does
        c ← a;   d ← b − a;   e ← d
    end
    if |f(c)| < |f(b)| then
        // make b the point with the smallest |f|
        a ← b;   b ← c;   c ← a
    end
    m ← (c − b) / 2                                  // bisection step
    δ ← ε_s |b|                                  // absolute tolerance
    if |m| ≤ δ or f(b) = 0 then
        return b
    end
    if |e| < δ or |f(a)| ≤ |f(b)| then
        d ← m;   e ← m                                        // bisect
    else
        s ← f(b) / f(a)
        if a = c then
            // only two distinct points:  secant step
            p ← 2ms;   q ← 1 − s
        else
            // three distinct points:  inverse quadratic
               interpolation
            q ← f(a) / f(c);   r ← f(b) / f(c)
            p ← s (2mq (q − r) − (b − a)(r − 1))
            q ← (q − 1)(r − 1)(s − 1)
        end
        if p > 0 then q ← −q
        else p ← −p
        if 2p < 3mq − |δq| and p < |eq/2| then
            // interpolated step stays in the bracket and shrinks
               it fast enough
            e ← d;   d ← p / q
        else
            d ← m;   e ← m                                    // bisect
        end
    end
    a ← b
    if |d| > δ then b ← b + d
    else b ← b + δ sign(m)
until converged
return b
```