

C++ code 8.15.8: A templated function implementing a generic Gauss-Newton iteration

```
2  template <typename VECTOR, typename FUNCTION, typename JACOBIAN,  
3          typename RECORDER =  
4          std::function<void(const VECTOR &, const VECTOR &, double)>>  
5  VECTOR gn(  
6      const VECTOR &init, FUNCTION &&F, JACOBIAN &&J, double rtol = 1.0E-6,  
7      double atol = 1.0E-8,  
8      RECORDER &&rec = [] (const VECTOR &, const VECTOR &, double) -> void {} ) {  
9      VECTOR x = init; // Vector for iterates  $x^{(k)}$   
10     // Vector for Gauss-Newton correction s  
11     auto f{F(x)}; // Auxiliary vector in the image space of F  
12     VECTOR s = J(x).householderQr().solve(f); //  
13     rec(x, s, f.norm());  
14     x = x - s; // Update of iterate  
15     // A posteriori termination based on absolute and relative tolerances  
16     while ((s.norm() > rtol * x.norm()) &&  
17           (s.norm() > atol)) { //  
18         f = F(x);  
19         s = J(x).householderQr().solve(f); //  
20         rec(x, s, f.norm()); // Possibly record iteration  
21         information  
22         x = x - s; // Update of iterate  
23     }  
24     return x;  
25 }
```

Get it on 🐱 GitLab ([locationestimation.hpp](#)).