

C++ code 11.11.6: Implementation of `rhoInverse()` → [GitLab](#)

```
2 template <typename RHOFUNCTOR, typename DRHOFUNCTOR>
3 double rhoInverse(double u, double z0, RHOFUNCTOR &&rho, DRHOFUNCTOR &&drho,
4                 double atol = 1.0E-10, double rtol = 1.0E-5) {
5     // Reduce to problem of finding zeros: phi(z)=0 <==> rho(z)=u
6     auto F = [rho, u](double z) -> double { return rho(z) - u; };
7     auto dF = [drho](double z) -> double { return drho(z); };
8
9     // Newton iteration for finding a zero of F
10    double s;
11    do {
12        s = F(z0) / dF(z0);
13        z0 = z0 - s;
14    } while (std::abs(s) > rtol * std::abs(z0) && std::abs(s) > atol);
15    return z0;
16 }
```