

## C++ code 5.2.36: Implementation of `sourcefn2()` → [GitLab](#)

```
2 Eigen::Matrix<double, Eigen::Dynamic, 2> sourcefn2(  
3     const Eigen::Matrix<double, 2, Eigen::Dynamic> &knots) {  
4     const std::size_t M = knots.cols() - 1;  
5     Eigen::Matrix<double, Eigen::Dynamic, 2> f(M, 2);  
6     // Definition quadrature points (2-point Gauss)  
7     const double rh = .5 + .5 / std::sqrt(3.);  
8     const double lh = .5 - .5 / std::sqrt(3.);  
9     const double h = 1. / static_cast<double>(M);  
10  
11     // Loop over all segments of the curve  
12     for (int i = 0; i < M; ++i) {  
13         // Compute the derivative of uh in the current cell  
14         const Eigen::VectorXd uhp = 1. / h * (knots.col(i + 1) - knots.col(i));  
15         const double uhp_norm = uhp.norm();  
16         // Evaluate the y-component of uh on both quadrature points  
17         const double uh2l = (1 - lh) * knots(1, i) + lh * knots(1, i + 1);  
18         const double uh2r = (1 - rh) * knots(1, i) + rh * knots(1, i + 1);  
19         // Compute the second components of the function f  
20         f(i, 0) = uhp_norm / (2.0 * std::sqrt(-uh2l) * uh2l);  
21         f(i, 1) = uhp_norm / (2.0 * std::sqrt(-uh2r) * uh2r);  
22     }  
23     return f;  
24 }
```