C++ code 9.10.25: `Eval()` method of BElemMatProvider ↬ GitLab

```cpp
Eigen::Matrix<double, 2, 3> BElemMatProvider::Eval(
    const lf::mesh::Entity &tria) {
  // Obtain vertex coordinates of the triangle in a 2x3 matrix
  const auto vertices = lf::geometry::Corners(*(tria.Geometry()));
  LF_ASSERT_MSG((vertices.cols() == 3) && (vertices.rows() == 2),
                "Invalid vertex coordinate " << vertices.rows() << "x"
                                             << vertices.cols() << " matrix");
  // Matrix for returning result
  Eigen::Matrix<double, 2, 3> elmat;
  // Compute the gradients of the barycentric coordinate functions
  // and store them in the columns of a 2x3 matrix grad_bary_coords
  Eigen::Matrix<double, 3, 3> X;  // temporary matrix
  X.block<3, 1>(0, 0) = Eigen::Vector3d::Ones();
  X.block<3, 2>(0, 1) = vertices.transpose();  // area of triangular cell
  const double area = 0.5 * std::abs(X.determinant());
  // This matrix contains the barycentric coordinate functions in its
     columns
  auto inv = X.inverse().eval();
  const Eigen::Matrix<double, 2, 3> grad_bary_coords{inv.block<2, 3>(1, 0)};
  // Since the local shape function for the finite element space Q_h are
  // Cartesian coordinate vectors, we just need to scale the components
     of the
  // gradients of the barycentric coordinate functions with the area of
     the
  // triangle.
  elmat = area * grad_bary_coords;
  return elmat;
}
```