

C++ code 3.2.2: Sub-problem (3-2.a): function `interpolateOntoQuadFE()` → [GitLab](#)

```
2  template <typename FUNCTOR>
3  Eigen::VectorXd interpolateOntoQuadFE(const If::assemble::DofHandler &dofh,
4                                         FUNCTOR &&f) {
5      // Obtain a pointer to the mesh object
6      auto mesh = dofh.Mesh();
7      const size_type N_dofs(dofh.NumDofs());
8      // variable for returning result vector
9      Eigen::VectorXd result = Eigen::VectorXd::Zero(N_dofs);
10     // Reference coordinates of the interpolation nodes of the triangle
11     Eigen::Matrix<double, 2, 6> refnodes(2, 6);
12     refnodes << 0.0, 1.0, 0.0, 0.5, 0.5, 0.0, 0.0, 0.0, 1.0, 0.0, 0.5, 0.5;
13     // Loop over the cells of the mesh (codim=0 entities)
14     for (const If::mesh::Entity *cell : mesh->Entities(0)) {
15         LF_ASSERT_MSG(cell->RefEl() == If::base::RefEl::kTria(),
16                       "Implemented for triangles only");
17         // Fetch pointer to associated geometry object
18         const If::geometry::Geometry *geom = cell->Geometry();
19         // Obtain actual coordinates of interpolation nodes
20         Eigen::MatrixXd nodes{geom->Global(refnodes)};
21         // get local to global index map for the current cell
22         auto glob_ind = dofh.GlobalDofIndices(*cell);
23         // Loop over local interpolation nodes
24         for (int i = 0; i < 6; i++) {
25             // update the result vector
26             result(glob_ind[i]) = f(nodes.col(i));
27         }
28     }
29     return result;
30 }
```