

### C++11 code 2.8.3: Sub-problem (2-8.d): Implementation of `solvePoissonBVP()` → [GitLab](#)

```
2 Eigen::VectorXd solvePoissonBVP() {
3     // Convert the globally defined function f to a LehrFEM++ mesh
      function object
4     If::mesh::utils::MeshFunctionGlobal mf_f{f};
5
6     // The basis expansion coefficient vector for the finite-element
      solution
7     Eigen::VectorXd solution = Eigen::VectorXd::Zero(1);
8
9     // Define the element matrix and element vector builders and solve the
      system
10    If::uscalfe::LinearFELaplaceElementMatrix elmat_builder;
11    If::uscalfe::LinearFELocalLoadVector<double, decltype(mf_f)> elvec_builder(
12        mf_f);
13
14    std::cout << "=====" << std::endl;
15    std::cout << "solvePoissonBVP" << std::endl;
16    // Call the function that solves the Galerkin linear system of
      equations
17    solution = solve(elmat_builder, elvec_builder);
18
19    return solution;
20 }
```