

C++11 code 2.8.15: Sub-problem (2-8.j): Implementation of `Eval()` for `MyLinearLoadVector`

→ [GitLab](#)

```
2 Eigen::Vector4d computeLoadVector(  
3     const Eigen::MatrixXd &vertices ,  
4     std::function<double(const Eigen::Vector2d &)> f) {  
5     // Number of nodes of the element:  triangles = 3,  rectangles = 4  
6     const int num_nodes = vertices.cols();  
7     // Vector for returning element vector  
8     Eigen::Vector4d elem_vec = Eigen::Vector4d::Zero();  
9     // Area of the cell  
10    double area;  
11    // Midpoints of edges in the reference cell  
12    Eigen::MatrixXd midpoints(2, num_nodes);  
13    switch (num_nodes) {  
14        case 3: {  
15            // Compute cell area for triangles  
16            area = 0.5 * ((vertices(0, 1) - vertices(0, 0)) *  
17                          (vertices(1, 2) - vertices(1, 0)) -  
18                          (vertices(1, 1) - vertices(1, 0)) *  
19                          (vertices(0, 2) - vertices(0, 0)));  
20            // clang-format off  
21            midpoints << vertices(0, 0) + vertices(0, 1),  
22            vertices(0, 1) + vertices(0, 2),  
23            vertices(0, 2) + vertices(0, 0),  
24            vertices(1, 0) + vertices(1, 1),  
25            vertices(1, 1) + vertices(1, 2),  
26            vertices(1, 2) + vertices(1, 0);  
27            // clang-format on  
28            break;  
29        }  
30        case 4: {  
31            // Compute cell area for rectangles  
32            area =  
33                (vertices(0, 1) - vertices(0, 0)) * (vertices(1, 3) - vertices(1, 0));  
34            // clang-format off  
35            midpoints << vertices(0, 0) + vertices(0, 1),  
36            vertices(0, 1) + vertices(0, 2),  
37            vertices(0, 2) + vertices(0, 3),  
38            vertices(0, 3) + vertices(0, 0),  
39            vertices(1, 0) + vertices(1, 1),  
40            vertices(1, 1) + vertices(1, 2),  
41            vertices(1, 2) + vertices(1, 3),  
42            vertices(1, 3) + vertices(1, 0);  
43            // clang-format on  
44            break;  
45        }  
46        default: {  
47            LF_ASSERT_MSG(false, "Illegal entity type!");  
48            break;  
49        }  
50    } // end switch  
51    midpoints *= 0.5; // The factor 1/2  
52    // Evaluate f(x) at the quadrature points, i.e. the midpoints of the  
53    // edges  
54    Eigen::VectorXd fvals = Eigen::VectorXd::Zero(4);  
55    for (int i = 0; i < num_nodes; ++i) {  
56        fvals(i) = f(midpoints.col(i));  
57    }
```