

C++11 code 11.7.18: Code for `interpolate()` → [GitLab](#)

```
2  template <typename VECSOURCE, typename VECDEST>
3  void interpolate(const VECSOURCE &s, VECDEST &d) {
4      // Determine number of cells
5      const std::size_t n = s.size();
6      const std::size_t N = d.size();
7      // Mesh widths/cell sizes
8      const double H = 1.0 / n;
9      const double h = 1.0 / N;
10     // Loop over nodes of destination mesh
11     for (int j = 0; j < N; ++j) {
12         // position of destination node
13         const double xd = h * (j + 0.5);
14         // Index of the source node, potentially < 0
15         int k = static_cast<int>(std::floor(xd / H - 0.5));
16         // Position of nodes of source cell, in which xd is contained
17         const double xsl = H * (k + 0.5); // left node
18         const double xsr = xsl + H;      // right node
19         // Values at source nodes taking into account periodic continuation
20         auto svl = s[(k < 0) ? k + n : k];
21         auto svr = s[(k + 1 >= n) ? k + 1 - n : k + 1];
22         // Linear interpolation inside source cell
23         d[j] = ((xsr - xd) * svl + (xd - xsl) * svr) / H;
24     }
25 }
```