

C++ code 9.2.17: Function `sdirk2SteppingLinScalODE()` → [GitLab](#)

```
2  std::vector<double> sdirk2SteppingLinScalODE(unsigned int m) {
3      std::vector<double> sol_vec;
4      double step_size = 2.0 / m;  // "tau" in this example
5      // Initial conditions
6      sol_vec.push_back(1.0);
7      // Discrete evolution operator. For the SDIRK-2 method applied to the
8      // scalar linear ODE  $(d/dt)y = -y$ , this turns out to be a scalar valued
9      // function depending on the step_size. Since we take equidistant step sizes
10     // in this example, the action of the evolution operator is simply
11     // multiplication by a constant double
12     double lambda = 1.0 - 0.5 * sqrt(2.0);
13     double evolution_op = 1.0 - step_size *
14         (1.0 + step_size * std::pow(lambda, 2)) /
15         std::pow(1.0 + step_size * lambda, 2);
16     // Compute discrete evolution by applying the evolution operator at each step
17     for (int i = 1; i < m + 1; i++) {
18         sol_vec.push_back(evolution_op * sol_vec.at(i - 1));
19     }
20     return sol_vec;
21 }
```