

C++ code 6.10.12: Sub-problem (6-10.e): Symplectic timestepting for (6.10.11) → [GITHUB](#)

```
2 Eigen::MatrixXd simulateHamiltonianDynamics(const Eigen::VectorXd &p0,
3                                             const Eigen::VectorXd &q0, double T,
4                                             unsigned int M) {
5     int n = p0.size();
6     Eigen::MatrixXd PQ(2 * n, M + 1);
7
8     // Coefficients of the method
9     Eigen::VectorXd a(3);
10    a << 2. / 3., -2. / 3., 1.;
11    Eigen::VectorXd b(3);
12    b << 7. / 24., 3. / 4., -1. / 24.;
13
14    double tau = T / M;
15    Eigen::VectorXd pj(p0), qj(q0);
16    PQ.col(0) << pj, qj;
17    for (int j = 1; j <= M; j++) { // integrate
18        for (int k = 0; k < 3; k++) { // one step
19            // f(q) = - 4 * |q|^2 * q
20            pj -= tau * b(k) * 4. * qj.squaredNorm() * qj;
21            // g(p) = p
22            qj += tau * a(k) * pj;
23            PQ.col(j) << pj, qj;
24        }
25    }
26    return PQ;
27 }
```