

C++ code 11.4.18: Sub-problem (11-4.k): function `numexpGodunovSmoothU0()`
→ [GITHUB](#)

```
2 Eigen::VectorXd numexpGodunovSmoothU0(const Eigen::VectorXi &M) {
3     const double T = 1.0;
4     const int M_size = M.size();
5     Eigen::VectorXd error(M_size);
6
7     for (int i = 0; i < M_size; ++i) {
8         Eigen::VectorXd x = getXValues(T, M(i));
9         Eigen::VectorXd u0 = x.unaryExpr(&smoothU0);
10        Eigen::VectorXd uT = solveGodunov(u0, T, M(i));
11
12        double tau = T / M(i);
13        double h = Constant::e * tau;
14        error(i) = h * (referenceSolution(x) - uT).lpNorm<1>();
15    }
16    return error;
17 }
```