C++ code 11.4.14: Sub-problem (11-4.h): Computation of reference solutuion ➔ GITHUB

```
double smoothU0(double x) {
  return (x < 0.0)
             ? 0.0
             : ((1.0 < x) ? 1.0 : Square(std::sin(0.5 * Constant::pi * x)));
}
Eigen::VectorXd referenceSolution(const Eigen::VectorXd &x) {
  double T = 1.0;
  // Reference solution on a very fine mesh
  unsigned int M = 3200;

  Eigen::VectorXd y = getXValues(T, M);
  Eigen::VectorXd u0 = y.unaryExpr(&smoothU0);
  Eigen::VectorXd u = solveLaxWendroff(u0, T, M);
  int N = x.size();
  Eigen::VectorXd u_ref(N);
  // The vector u is larger than u_ref. Use eval() from above the "evaluate" u
  // at the positions x(i) and thus obtain the reference solution u_ref.
  for (int i = 0; i < N; ++i) {
    u_ref(i) = eval(u, y, x(i));
  }
  return u_ref;
}
```