

# 1 Problem Set I solving wave equation

$$\frac{\partial^2 \phi}{\partial^2 t} = c^2 \frac{\partial^2 \phi}{\partial^2 x} \quad (1)$$

## 1.1 fully first order formulation

$$\eta = \phi_{,t}, \quad \chi = \phi_{,x} \quad (2)$$

$$\eta(t,x)\chi(t,x)\vec{u}(\phi,\eta,\chi)$$

$$\vec{u}_{,t} + \mathbf{A}\vec{u}_{,x} = \vec{S} \quad (3)$$

## 1.2 initial condition

$$\phi(0, x) = e^{\sin^2(\frac{\pi x}{L})} - 1, \quad 0 \leq x \leq L \quad (4)$$

with periodic condition:

$$\phi(t, x) = \phi(t, x \pm L) \quad (5)$$

# 2 Program

```
1 // The C++ standard version:
2 #include <cstdio>
3 #include <cmath>
4 #include <fstream>
5 #include <iostream>
6 using namespace std ;
7
8 void output(int ti, int xi, double t[], double x[], double phi[]);
9 void init(double t[], double x[], double phi[], int xsteps, double L);
10 void output(int ti, int xi, double t[], double x[], double phi[]){
11     // Declaring 2D array
12     // t x phi
13     cout << t[ti] << ' ' << x[xi] << ' ' << phi[xi] << endl;
14 }
15 void init(double t[], double x[], double phi[], int xsteps, double dx, double L){
16     t[0]=0;
17     //x[0]=0;
18     for (int i = 0; i <= xsteps; i=i+1) {
19         phi[i] = exp(pow(sin(M_PI/L*(i*dx)),2))-1;
20         x[i]=i*dx;
21         output(0,i,t,x,phi);
22     }
23 }
24
25 const double CSpeed=1;
26 const double CMax = 1;
27 const double dx = 0.05;
28 const double L = 1; // gridSpace
29 const double timeLength = 1;
30 const double dt = CMax*dx/CSpeed;
31 const int xSteps = int( L / dx );
32 const int tSteps = int (timeLength / dt );
33
34 double // Kommentar
```

```

35     x[xSteps],
36     t[tSteps],
37     phi[xSteps]
38     ;
39
40 int main(int argc, char** argv)
41 {
42
43     init(t, x, phi, xSteps, dx, L);
44
45
46     // cases for solver
47     //{{solving method second order}}
48     //{{solving method forth order}}
49
50     // Initialize 2D array using loop
51     //for (int i = 0; i < 4; i++) {
52     //    for (int j = 0; j < 4; j++) {
53     //        arr[i][j] = i + j;
54     //    }
55     //}
56     return 0;
57 };
58
59
60
61 //example function in cpp
62 //void pred_corr(double x[],double h,int i,double dxdt[])
63 //{
64 //    double
65 //    gam=1.,
66 //    beta=2.
67 //    ;
68 //    double n =7;
69 //    //predictor step
70 //    dxdt[i]=beta*x[i-tau]/(1+pow(x[i-tau],n))-gam*x[i];
71 //    x[i+1]=x[i]+h/12.*(23.*dxdt[i]-16.*dxdt[i-1]+5.*dxdt[i-2]);
72 //    //corrector step
73 //    dxdt[i+1]=beta*x[i+1-tau]/(1+pow(x[i+1-tau],n))-gam*x[i+1];
74 //    ↪ x[i+1]=x[i]+h/12.*(5.*dxdt[i+1]+8.*dxdt[i]-dxdt[i-1]);
75 //};

```