

```

#include <stdio.h>
#include <stdlib.h>
#include <math.h>

double u_0 (double x) {
    if ( 0<x && x<2 )
        return 1-fabs(x-1);
    else
        return 0;
}

int main (int argc, char **args) {
    int i, j;
    FILE *file;
    char filename[512];

    if ( argc != 3 ) {
        fprintf(stderr, "USAGE: %s N M\n", args[0]);
        exit(3);
    }

    int N, M;
    if ( sscanf(args[1], "%d", &N) != 1 || sscanf(args[2], "%d", &M) != 1 ) {
        fprintf(stderr, "USAGE: %s N M\n", args[0]);
        exit(3);
    }

    double a = 1;
    double X = 4;
    double T = 1;

    double h = X/N;
    double tau = T/M;

    double lambda = a*tau/h;

    double * u_alt = (double*) calloc(N+1, sizeof(double));
    double * u_neu = (double*) calloc(N+1, sizeof(double));
    if ( u_alt == NULL || u_neu == NULL ) {
        fprintf(stderr, "ERROR: Fehler beim allokieren des Speichers.\n");
        exit(1);
    }

    printf("tau <= h/a: \t\t %f <= %f\n", tau, h/a);

    // u_alt mit den Anfangswerten initialisieren
    for (i=0; i<=N; i++)
        u_alt[i] = u_0(i*h);

    for (j=1; j<=M; j++) {
        // fuehre einen Schritt des Verfahrens aus

```

```

// zuerst setze den Randwert von u_neu fuer i=0
u_neu[0] = 0;

// berechne nun die Werte von u_neu fuer i=1..N-1 aus den Werten von u_alt
for (i=1; i<=N; i++) {
    u_neu[i] = u_alt[i] - lambda * ( u_alt[i] - u_alt[i-1] );
}

// kopiere fuer den naechsten Schleifendurchlauf u_neu in u_alt
for (i=0; i<=N; i++)
    u_alt[i] = u_neu[i];

// Schreibe die Approximation zum Zeitpunkt t_j = j*tau in eine Datei
sprintf(filename, "erg_1_j%d.dat", j);
file = fopen(filename, "w");
if ( file == NULL ) {
    fprintf(stderr, "ERROR: Fehler beim oeffnen der Datei.\n");
    exit(2);
}

for (i=0; i<=N; i++)
    fprintf(file, "%5.4f\t%5.4f\n", i*h, u_neu[i]);

fclose(file);
}

// Schreibe die exakte Loesung in eine Datei
file = fopen("exakt_1.dat", "w");
if ( file == NULL ) {
    fprintf(stderr, "ERROR: Fehler beim oeffnen der Datei.\n");
    exit(2);
}

N = 500; h = X/N;
for (i=0; i<=N; i++) {
    fprintf(file, "%5.4f\t", i*h);
    for (j=0; j<=M; j++)
        fprintf(file, "%5.4f\t", u_0(i*h-a*j*tau));

    fprintf(file, "\n");
}

fclose(file);

return 0;
}

```