

```
// Q3.b.
```

```
#include <stdio.h>
#include <stdlib.h>
#include <math.h>
#include "nr.h"
#include "nrutil.h"
#include "math.h"

int main() {
    int N_time = 50001;
    int N_space = 101;
    int i = 0, j = 0;

    float x0 = -1.0, x1 = 1.0;
    float *a, *b, *c, *r, *u, *w;

    double dx = (x1 - x0) / (N_space - 1.);    // Step in x.
    double dt = 0.125 * dx * dx;               // Time step.
    double alpha = dt / (dx * dx);
    double theta = 1;

    a = vector(0, N_space);
    b = vector(0, N_space);
    c = vector(0, N_space);
    r = vector(0, N_space);
    u = vector(0, N_space);
    w = vector(0, N_space);

    // Initial condition.
    for (i = 0; i < N_space; i++) {
        u[i] = -1 + 2 * i / (N_space - 1.);
    }

    for (j = 1; j < N_time; j++) {
        // The tridiagonal matrix.
        for (i = 0; i < N_space; i++) {
            if (i == N_space - 1) a[i] = 0;    // Boundary condition.
            else a[i] = -3 * alpha * theta * pow(u[i - 1], 2);

            if (i == 0) c[i] = 0;
            else c[i] = -3 * alpha * theta * pow(u[i + 1], 2);

            if (i == 0 || i == N_space - 1) b[i] = 1.0;
            else b[i] = 1 + 6 * alpha * theta * pow(u[i], 2);
        }

        for (i = 0; i < N_space; i++) {
            if (i == 0) r[i] = 0;
            else if (i == N_space - 1) r[i] = 0;
            else r[i] = alpha * (pow(u[i + 1], 3) - 2 * pow(u[i], 3) + pow(u[i - 1], 3));
        }

        tridag(a, b, c, r, w, N_space - 1);

        // Update the values of u from w.
        for (i = 0; i < N_space; i++) {
            if (i == 0) u[i] = -1;
```

```
        else if (i == N_space - 1) u[i] = 1;
        else u[i] = w[i] + u[i];
    }

    if (j == 100) {
        for (i = 0; i < N_space; i++) {
            printf("%d\t%d\t%f\t%f\n", j, i, x0 + i * dx, u[i]);
        }
    }

    free_vector(a, 0, N_space);
    free_vector(b, 0, N_space);
    free_vector(c, 0, N_space);
    free_vector(r, 0, N_space);
    free_vector(u, 0, N_space);
    free_vector(w, 0, N_space);

    return 0;
}
```