**Міністерство освіти і науки України**

**Чернівецький національний університет імені Юрія Федьковича**

**Факультет математики та інформатики**

**Кафедра математичного моделювання**

**Звіт**

**до лабораторної роботи №4**

**з «Комп'ютерного моделювання жорстких процесів та систем»**

Виконав:

cтудент 301-Б групи

Мар’янчук О.О.

Викладач:

кандидат фіз.-мат. наук, асистент

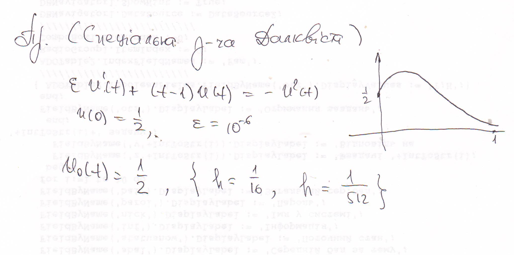
Іліка С.А.

**Чернівці 2023**

**Лабораторна робота № 4**

**Завдання:**

Розв’язати спеціальну задачу Далквіста неявною різницева схемою Ейлера з припасованим коєфіцієнтом:



Загальний вид неявної різницевої схема Ейлера з припасованим коєфіцієнтом:

Задача Далквіста

Заміна:

Різницева схема:

**Код до завдання:**

#include <iostream>

#include <vector>

#include <iomanip>

const double eps = 1e-6;

const size\_t maxIteration = 100;

const double tolerance = 1e-2;

class DifferenceScheme

{

public:

    DifferenceScheme(double \_t0, double \_t1, double \_h, double u0) :

        h(\_h),

        t0(\_t0),

        t1(\_t1),

        node\_count((t1 - t0) / h),

        u(node\_count)

    {

        u[0] = 1.0 / u0;

        implicitEilerWithFittedCoef();

        for (size\_t i = 0; i < node\_count; i++)

            u[i] = 1.0 / u[i];

    }

    double coef(size\_t i)

    {

        const double E = 2.71828182846;

        double t = h \* i;

        return ro \* (1 - t) / (1 - std::pow(E,-ro \* (1 - t)));

    }

    void implicitEilerWithFittedCoef() {

        for (size\_t i = 1; i < node\_count; i++)

        {

            double uil;

            u[i] = u[i - 1];

            size\_t k = 0;

            do

            {

                uil = u[i];

                double t = h \* i;

                u[i] = u[i-1] + ro / coef(i) \* (1 + (t - 1) \* u[i]);

            } while (std::abs(uil - u[i]) > tolerance && k++ < maxIteration);

        }

    }

    void printTable(size\_t width = 20) {

        std::cout << '|' << std::setfill('-') << std::setw(width) << 't'

            << '|' << std::setw(width) << 'u' << "|\n";

        std::cout << std::setfill(' ');

        for (size\_t i = 0; i < u.size(); i++)

        {

            std::cout << '|' << std::setw(width) << i \* h

                << '|' << std::setw(width) << u[i] << "|\n";

        }

    }

private:

    double h;

    double t0;

    double t1;

    size\_t node\_count;

    std::vector<double> u;

    double ro = h / eps;

};

int main() {

    std::cout << "h = 1/16" << std::endl;

    DifferenceScheme scheme1(0, 1, 1.0 / 16, 1.0 / 2);

    scheme1.printTable();

    std::cout << "h = 1/512" << std::endl;

    DifferenceScheme scheme2(0, 1, 1.0 / 512, 1.0 / 2);

    scheme2.printTable();

    return 0;

**Скріншоти виконання:**

