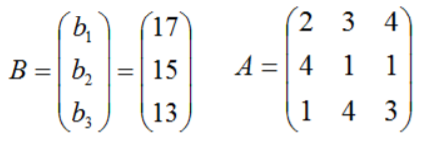
Лабораторная работа 3

Компьютерное моделирование на основе решения систем линейных уравнений.

1.Из порта необходимо вывезти 3 вида груза в количестве b1, b2, b3. В распоряжении имеется 3 типа судов, причем j-е судно может взять на борт aij единиц i-го груза. определить количество судов каждого типа x1, x2, x3, которое необходимо привлечь для вывоза груза при следующих исходных данных:



Код программы:

#include <stdio.h>

#define \_i 3

#define \_j 4

void solve(float \*a, float \*b){

    //создание одной матрицы array

    float array[\_i][\_j];

    int count\_a = 0, count\_b = 0;

    for(int i = 0; i < \_i; i++){

        for(int j = 0; j < \_j-1; j++){

            array[i][j] = a[count\_a];

            count\_a++;

        }

        array[i][\_j-1]=b[count\_b];

        count\_b++;

    }

    //решение задачи

    //прямой ход получаем треугольную матрицу

    float a\_ii, kf;

    for(int i = 0; i < \_i-1; i++)

    {

        a\_ii = array[i][i];

        for(int j = i; j < \_j; j++)

        {

            array[i][j] = array[i][j] / a\_ii;

        }

        for(int k = i + 1; k <\_i; k++)

        {

            kf = array[k][i];

            for(int j = 0; j < \_j; j++)

            {

                array[k][j] -= kf \* array[i][j];

            }

        }

    }

    kf = 1/array[\_i-1][\_j-2];

    for(int j = \_j-2; j<\_j;j++)

        array[\_i-1][j]\*=kf;

    //обратный ход получаем единичную матрицу матрицу

    for(int i = \_i-1; i > 0; i--)

    {

        for (int k = i-1; k > -1; k--)

        {

            kf = array[k][i]/array[i][i];

            for(int j = 0; j < \_j; j++)

            {

                array[k][j] -= kf \* array[i][j];

            }

        }

    }

    //вывод ответа

    for (int i=0; i<\_i; i++){

        printf("X%d = %f ",i+1, array[i][\_j-1]);

        printf("\n");

    }

}

int main(){

    float a[\_i][\_j-1]={{2, 3, 4},

                       {4, 1, 1},

                       {1, 4, 3}};

    float b[\_i][1]={{17}, {15}, {13}};

    solve(&a[0][0], &b[0][0]);

    return 0;

}

Результат работы программы:



Ответ: x1=3, x2=1, x3=2.

2. Имеется система 4-х грузов, связанных нитью (рис 1). Три груза расположены на горизонтальной плоскости. Система приводится в движение грузом, который движется в вертикальном направлении и подвешен на нити, перекинутой через неподвижный блок. Определить ускорение движения системы и силы натяжения каждой нити, если m1 = m2 = m3 = m4 = 1 кг. Коэффициент трения kтр = 0,01, ускорение свободного падения g = 9,8 м / с^2.

Код программы:

#include <stdio.h>

#define \_i 4

#define \_j 5

void solve(float \*a, float \*b){

    //создание одной матрицы array

    float array[\_i][\_j];

    int count\_a = 0, count\_b = 0;

    for(int i = 0; i < \_i; i++){

        for(int j = 0; j < \_j-1; j++){

            array[i][j] = a[count\_a];

            count\_a++;

        }

        array[i][\_j-1]=b[count\_b];

        count\_b++;

    }

    //решение задачи

    //прямой ход получаем треугольную матрицу

    float a\_ii, kf;

    for(int i = 0; i < \_i-1; i++)

    {

        a\_ii = array[i][i];

        for(int j = i; j < \_j; j++)

        {

            array[i][j] = array[i][j] / a\_ii;

        }

        for(int k = i + 1; k <\_i; k++)

        {

            kf = array[k][i];

            for(int j = 0; j < \_j; j++)

            {

                array[k][j] -= kf \* array[i][j];

            }

        }

    }

    kf = 1/array[\_i-1][\_j-2];

    for(int j = \_j-2; j<\_j;j++)

        array[\_i-1][j]\*=kf;

    //обратный ход получаем единичную матрицу матрицу

    for(int i = \_i-1; i > 0; i--)

    {

        for (int k = i-1; k > -1; k--)

        {

            kf = array[k][i]/array[i][i];

            for(int j = 0; j < \_j; j++)

            {

                array[k][j] -= kf \* array[i][j];

            }

        }

    }

    //вывод ответа

    for (int i=0; i<\_i; i++){

        printf("X%d = %f ",i+1, array[i][\_j-1]);

        printf("\n");

    }

}

int main(){

    float a[\_i][\_j-1]={{1, 0, 0, 1},

                       {-1, 1, 0, 1},

                       {0, -1, 1, 1},

                       {0, 0, -1, 1}};

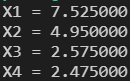
    float b[\_i][1]={{10}, {-0.1}, {0.1}, {-0.1}};

    solve(&a[0][0], &b[0][0]);

    return 0;

}

Результат работы программы:



Вывод: x4 – искомое ускорение системы, следовательно а = 2,475 м/с.