

Міністерство освіти і науки України
Національний технічний університет України «Київський політехнічний інститут
імені Ігоря Сікорського»
Факультет інформатики та обчислювальної техніки
Кафедра інформатики та програмної інженерії

Звіт

з лабораторної роботи № 5 з
дисципліни «Основи програмування 2.
Модульне програмування»

«Успадкування»

Варіант 14

Виконав студент ІП-13, Замковий Дмитро Володимирович

(шифр, прізвище, ім'я, по батькові)

Перевірів Вечерковська Анастасія Сергіївна

(прізвище, ім'я, по батькові)

Лабораторна робота 5

Успадкування

Мета заняття:

Завдання: Спроекувати клас "Рухома матеріальна точка", який представляє точку, що рухається в певному напрямку і містить методи для визначення координат точки в заданий момент часу t та обчислити відстань від неї до іншої точки. На основі цього класу створити класи-нащадки "Рухома матеріальна точка (x, y, z)", яка рухається у просторі і її координати визначаються як $x=x_0+a_1*\sin(t)$, $y=y_0+a_2*\cos(t)$, $z=z_0+a_3*t^2$ та "Рухома матеріальна точка (x, y)", яка рухається по площині і її координати визначаються як $x=x_0+a_1*\sin(t)$, $y=y_0+a_2*\cos(t)$, $z=0$. Створити q об'єктів класу "Рухома матеріальна точка (x, y, z)" та k об'єктів класу "Рухома матеріальна точка (x, y)" (дані згенерувати випадковим чином). Визначити найменшу відстань між рухомими матеріальними точками у заданий користувачем момент часу t

Хід роботи:

Виконання програми на мові програмування C++:

Файл Lab5.cpp:

```
#include "TPoint.h"
#include <iostream>
#include <vector>
#include <time.h>
#include <stdlib.h>
using namespace std;

/*
 * Спроекувати клас "Рухома матеріальна точка", який представляє точку, що
 * рухається в певному напрямку і містить методи для визначення координат точки
 * в заданий момент часу t та обчислити відстань від неї до іншої точки. На основі
 * цього класу створити класи-нащадки "Рухома матеріальна точка (x, y, z)", яка ру-
 * хається у просторі і її координати визначаються як  $x=x_0+a_1*\sin(t)$ ,  $y=y_0+a_2*\cos(t)$ ,
 *  $z=z_0+a_3*t^2$  та "Рухома матеріальна точка (x, y)", яка рухається по площині і її ко-
 * ординати визначаються як  $x=x_0+a_1*\sin(t)$ ,  $y=y_0+a_2*\cos(t)$ ,  $z=0$ . Створити q об'єктів
 * класу "Рухома матеріальна точка (x, y, z)" та k об'єктів класу "Рухома матеріальна
 * точка (x, y)" (дані згенерувати випадковим чином). Визначити найменшу відстань
 * між рухомими матеріальними точками у заданий користувачем момент часу t
 */

int main()
{
    char mode;
    int q, k, t;
    cout << "Рухома матеріальна точка (x, y, z)" << endl;
    do {
        cout << "Генерувати випадкові числа? (+/-): "; cin >> mode;
    } while (mode != '+' && mode != '-');
    do {
```

```

        cout << "Введіть q: "; cin >> q;
    } while (q < 2);
    do {
        cout << "Введіть t: "; cin >> t;
    } while (t < 0);
    vector<TPoint_xyz> xyz;
    if (mode == '+')
    {
        srand(time(NULL));
        for (int i = 0; i < q; i++)
        {
            xyz.push_back(TPoint_xyz(t));
        }
    }
    else
    {
        double x0, ax, y0, ay, z0, az;
        for (int i = 0; i < q; i++)
        {
            cout << i + 1 << ": Введіть значення (x0, ax, y0, ay, z0, az): ";
            cin >> x0 >> ax >> y0 >> ay >> z0 >> az;
            xyz.push_back(TPoint_xyz(x0, ax, y0, ay, z0, az, t));
        }
    }
    double min_val = INFINITY, tmp;
    int min_t1 = 0,
        min_t2 = 0;

    for (int i = 0; i < q - 1; i++)
    {
        for (int j = i + 1; j < q; j++)
        {
            tmp = xyz[i].count_distace(xyz[j].get_coord());
            if (tmp < min_val)
            {
                min_t1 = i;
                min_t2 = j;
                min_val = tmp;
            }
        }
    }
    for (int i = 0; i < q; i++)
    {
        xyz[i].output(to_string(i));
    }
    cout << "Найменше значення між точками " << min_t1 << " i " << min_t2 << " : " <<
min_val;

    cout << endl << endl << "=====" << endl << endl;

    cout << "Рухом матеріальна точка (x, y)" << endl;
    do {
        cout << "Генерувати випадкові числа? (+/-): "; cin >> mode;
    } while (mode != '+' && mode != '-');
    do {
        cout << "Введіть k: "; cin >> k;
    } while (k < 2);
    do {
        cout << "Введіть t: "; cin >> t;
    } while (t < 0);
    vector<TPoint_xy> xy;
    if (mode == '+')

```

```

    {
        for (int i = 0; i < k; i++)
        {
            xy.push_back(TPoint_xy(t));
        }
    }
    else
    {
        double x0, ax, y0, ay;
        for (int i = 0; i < k; i++)
        {
            cout << i + 1 << ": Введіть значення (x0, ax, y0, ay): ";
            cin >> x0 >> ax >> y0 >> ay;
            xy.push_back(TPoint_xy(x0, ax, y0, ay, t));
        }
    }
    min_val = INFINITY;
    min_t1 = 0;
    min_t2 = 0;

    for (int i = 0; i < k - 1; i++)
    {
        for (int j = i + 1; j < k; j++)
        {
            tmp = xy[i].count_distace(xy[j].get_coord());
            if (tmp < min_val)
            {
                min_t1 = i;
                min_t2 = j;
                min_val = tmp;
            }
        }
    }
    for (int i = 0; i < k; i++)
    {
        xy[i].output(to_string(i));
    }
    cout << "Найменше значення між точками " << min_t1 << " i " << min_t2 << " : " <<
    min_val;
}

```

Файл TPoint.h:

```

#pragma once
#include <string>

double random_val();

class TCoordinates {
private:
    double x, y, z;
public:
    TCoordinates() { x = 0; y = 0; z = 0; };
    TCoordinates(double tx, double ty, double tz) : x(tx), y(ty), z(tz) {};
    double get_x() { return x; };
    double get_y() { return y; };
    double get_z() { return z; };
    void set_x(double val) { x = val; };
    void set_y(double val) { y = val; };
    void set_z(double val) { z = val; };
};

class TPoint
{
protected:

```

```

        TCoordinates coord;
public:
    void output(std::string str);
    TCoordinates get_coord() { return coord; };

    virtual double count_distance(TCoordinates t) = 0;
};

class TPoint_xy :
    public TPoint
{
public:
    TPoint_xy(double t);
    TPoint_xy(double x0, double ax, double y0, double ay, double t);
    double count_distance(TCoordinates t);
};

class TPoint_xyz :
    public TPoint
{
public:
    TPoint_xyz(double t);
    TPoint_xyz(double x0, double ax, double y0, double ay, double z0, double az, double
t);
    double count_distance(TCoordinates t);
};

```

Файл TPoint.cpp:

```

#include "TPoint.h"
#include <time.h>
#include <stdlib.h>
#include <math.h>
#include <iostream>
#include <string>
using namespace std;

double random_val()
{
    return (-10 + rand() % 20 + rand() / 10000.0);
}

void TPoint::output(string str)
{
    cout << "=====" << str << "=====" << endl;
    cout << "\tx: " << coord.get_x() << endl
        << "\ty: " << coord.get_y() << endl
        << "\tz: " << coord.get_z() << endl;
}

TPoint_xy::TPoint_xy(double t)
{
    double x0 = random_val();
    double ax = random_val();
    double y0 = random_val();
    double ay = random_val();
    coord.set_x(x0 + ax * sin(t));
    coord.set_y(y0 + ay * cos(t));
    coord.set_z(0);
}

```

```

TPoint_xy::TPoint_xy(double x0, double ax, double y0, double ay, double t)
{
    coord.set_x(x0 + ax * sin(t));
    coord.set_y(y0 + ay * cos(t));
    coord.set_z(0);
}

double TPoint_xy::count_distance(TCoordinates t)
{
    return sqrt(pow(t.get_x() - coord.get_x(), 2) + pow(t.get_y() - coord.get_y(), 2));
}

TPoint_xyz::TPoint_xyz(double t)
{
    double x0 = random_val(),
           ax = random_val(),
           y0 = random_val(),
           ay = random_val(),
           z0 = random_val(),
           az = random_val();
    coord.set_x(x0 + ax * sin(t));
    coord.set_y(y0 + ay * cos(t));
    coord.set_z(z0 + az * t * t);
}

TPoint_xyz::TPoint_xyz(double x0, double ax, double y0, double ay, double z0, double az,
double t)
{
    coord.set_x(x0 + ax * sin(t));
    coord.set_y(y0 + ay * cos(t));
    coord.set_z(z0 + az * t * t);
}

double TPoint_xyz::count_distance(TCoordinates t)
{
    return sqrt(pow(t.get_x() - coord.get_x(), 2) + pow(t.get_y() - coord.get_y(), 2) +
pow(t.get_z() - coord.get_z(), 2));
}

```

Виконання програми на мові програмування Python:

Файл main.py:

```

from TPoint import *
from random import random

mode = ''
q = 0
k = 0
t = -1
xyz = {}

print('Рухома матеріальна точка (x, y, z)')
while mode != '+' and mode != '-':
    mode = input('Генерувати випадкові числа? (+/-): ')

while q < 2:
    q = int(input('Введіть q: '))

while t < 0:
    t = int(input('Введіть t: '))

```

```
if mode == '+':
    for i in range(q):
        xyz[i] = TPoint_xyz(random() * 100, random() * 100, random() * 100,
random() * 100, random() * 100, random() * 100, t)
else:
    for i in range(q):
        print(f'{i}: Введіть значення (x0, ax, y0, ay, z0, az): ')
        x0 = int(input())
        ax = int(input())
        y0 = int(input())
        ay = int(input())
        z0 = int(input())
        az = int(input())
        xyz[i] = TPoint_xyz(x0, ax, y0, ay, z0, az, t)

min_val = float('inf')
min_t1 = 0
min_t2 = 0

for i in range(q-1):
    for j in range(i + 1, q):
        tmp = xyz[i].count_distace(xyz[j].coord)
        if tmp < min_val:
            min_val = tmp
            min_t1 = i
            min_t2 = j

for i in range(q):
    xyz[i].output(i)

print(f'Найменше значення між точками {min_t1} і {min_t2}: {min_val}')

print('\n\n\n')

mode = ''
t = -1
xy = {}

print('Рухома матеріальна точка (x, y)')
while mode != '+' and mode != '-':
    mode = input('Генерувати випадкові числа? (+/-): ')

while k < 2:
    k = int(input('Введіть k: '))

while t < 0:
    t = int(input('Введіть t: '))

if mode == '+':
    for i in range(q):
        xy[i] = TPoint_xy(random() * 100, random() * 100, random() * 100,
random() * 100, t)
else:
    for i in range(q):
        print(f'{i}: Введіть значення (x0, ax, y0, ay): ')
        x0 = int(input())
        ax = int(input())
        y0 = int(input())
        ay = int(input())
        xy[i] = TPoint_xy(x0, ax, y0, ay, t)
```

```

min_val = float('inf')
min_t1 = 0
min_t2 = 0

for i in range(k-1):
    for j in range(i + 1, k):
        tmp = xy[i].count_distace(xy[j].coord)
        if tmp < min_val:
            min_val = tmp
            min_t1 = i
            min_t2 = j

for i in range(k):
    xy[i].output(i)

print(f'Найменше значення між точками {min_t1} і {min_t2}: {min_val}')

```

Файл TPoint.py:

```

from TPoint import *
from random import random

mode = ''
q = 0
k = 0
t = -1
xyz = {}

print('Рухома матеріальна точка (x, y, z)')
while mode != '+' and mode != '-':
    mode = input('Генерувати випадкові числа? (+/-): ')

while q < 2:
    q = int(input('Введіть q: '))

while t < 0:
    t = int(input('Введіть t: '))

if mode == '+':
    for i in range(q):
        xyz[i] = TPoint_xyz(random() * 100, random() * 100, random() * 100,
random() * 100, random() * 100, random() * 100, t)
else:
    for i in range(q):
        print(f'{i}: Введіть значення (x0, ax, y0, ay, z0, az): ')
        x0 = int(input())
        ax = int(input())
        y0 = int(input())
        ay = int(input())
        z0 = int(input())
        az = int(input())
        xyz[i] = TPoint_xyz(x0, ax, y0, ay, z0, az, t)

min_val = float('inf')
min_t1 = 0
min_t2 = 0

for i in range(q-1):
    for j in range(i + 1, q):
        tmp = xyz[i].count_distace(xyz[j].coord)

```



```

        if tmp < min_val:
            min_val = tmp
            min_t1 = i
            min_t2 = j

for i in range(q):
    xyz[i].output(i)

print(f'Найменше значення між точками {min_t1} і {min_t2}: {min_val}')

print('\n\n\n')

mode = ''
t = -1
xy = {}

print('Рухома матеріальна точка (x, y)')
while mode != '+' and mode != '-':
    mode = input('Генерувати випадкові числа? (+/-): ')

while k < 2:
    k = int(input('Введіть k: '))

while t < 0:
    t = int(input('Введіть t: '))

if mode == '+':
    for i in range(q):
        xy[i] = TPoint_xy(random() * 100, random() * 100, random() * 100,
random() * 100, t)
else:
    for i in range(q):
        print(f'{i}: Введіть значення (x0, ax, y0, ay): ')
        x0 = int(input())
        ax = int(input())
        y0 = int(input())
        ay = int(input())
        xy[i] = TPoint_xy(x0, ax, y0, ay, t)

min_val = float('inf')
min_t1 = 0
min_t2 = 0

for i in range(k-1):
    for j in range(i + 1, k):
        tmp = xy[i].count_distance(xy[j].coord)
        if tmp < min_val:
            min_val = tmp
            min_t1 = i
            min_t2 = j

for i in range(k):
    xy[i].output(i)

print(f'Найменше значення між точками {min_t1} і {min_t2}: {min_val}')

```

Результат:

На мові програмування C++:

```

Консоль отладки Microsoft Visual Studio
Рухома матеріальна точка (x, y, z)
Генерувати випадкові числа? (+/-): +
Введіть q: 3
Введіть t: 5
=====0=====
      x: -9.27342
      y: -5.49729
      z: -194.977
=====1=====
      x: -13.9843
      y: -3.79233
      z: 40.4582
=====2=====
      x: 7.68512
      y: -3.2154
      z: -99.6447
Найменше значення між точками 0 і 2 : 96.8559

=====
Рухома матеріальна точка (x, y)
Генерувати випадкові числа? (+/-): 3
Генерувати випадкові числа? (+/-): +
Введіть k: 3
Введіть t: 5
=====0=====
      x: -0.995369
      y: -1.79051
      z: 0
=====1=====
      x: 6.0705
      y: 7.32855
      z: 0
=====2=====
      x: 5.26236
      y: 11.44
      z: 0
Найменше значення між точками 1 і 2 : 4.19008
C:\Users\Dima\source\VS\OP_labs\Lab5\Debug\Lab5.exe (процесс 11804) завершил работу с кодом 0.
Нажмите любую клавишу, чтобы закрыть это окно...

```

На мові програмування Python:

```
C:\Users\Dima\source\PC\OP\venv\Scripts\python.exe C:/Users/Dima/source/PC/OP/Lab5/main.py
Рухома матеріальна точка (x, y, z)
Генерувати випадкові числа? (+/-): +
Введіть q: 3
Введіть t: 5
===== 0 =====
      x: 61.67316116027443
      y: 42.84810219931731
      z: 2384.9637172719104

===== 1 =====
      x: -58.57808131959263
      y: 35.09042534196163
      z: 48.19332033688549

===== 2 =====
      x: 96.27006258831491
      y: 100.64671179991554
      z: 1928.2303085647713

Найменше значення між точками 0 і 2: 461.67416160039363


Рухома матеріальна точка (x, y)
Генерувати випадкові числа? (+/-): +
Введіть k: 2
Введіть t: 7
===== 0 =====
      x: 74.75378551787409
      y: 62.521208503268056
      z: 0

===== 1 =====
      x: 134.72281996570393
      y: 115.19701423227747
      z: 0

Найменше значення між точками 0 і 1: 79.81870458611404
```