НАЦІОНАЛЬНИЙ АВІАЦІЙНИЙ УНІВЕРСИТЕТ

Факультет кібербезпеки, комп’ютерної та програмної інженерії

Кафедра комп'ютерних інформаційних технологій

**Лабораторна робота № 4.1**

**з навчальної дисципліни**

**"Математичні моделі динамічних систем"**

Тема: ДОСЛІДЖЕННЯ НАВІГАЦІЙНИХ ЕЛЕМЕНТІВ ПОЛЬОТУ НА ПЕОМ

Виконавець: студент групи УС-311Дзиговський В.І.

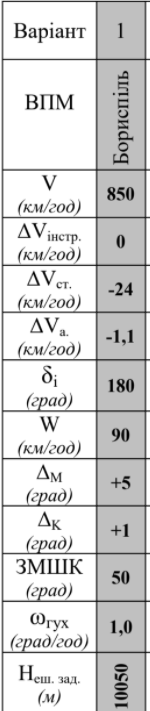
Київ 2020

**Мета лабораторної роботи:**

1. Вивчити навігаційні елементи польоту та методи їх визначення.

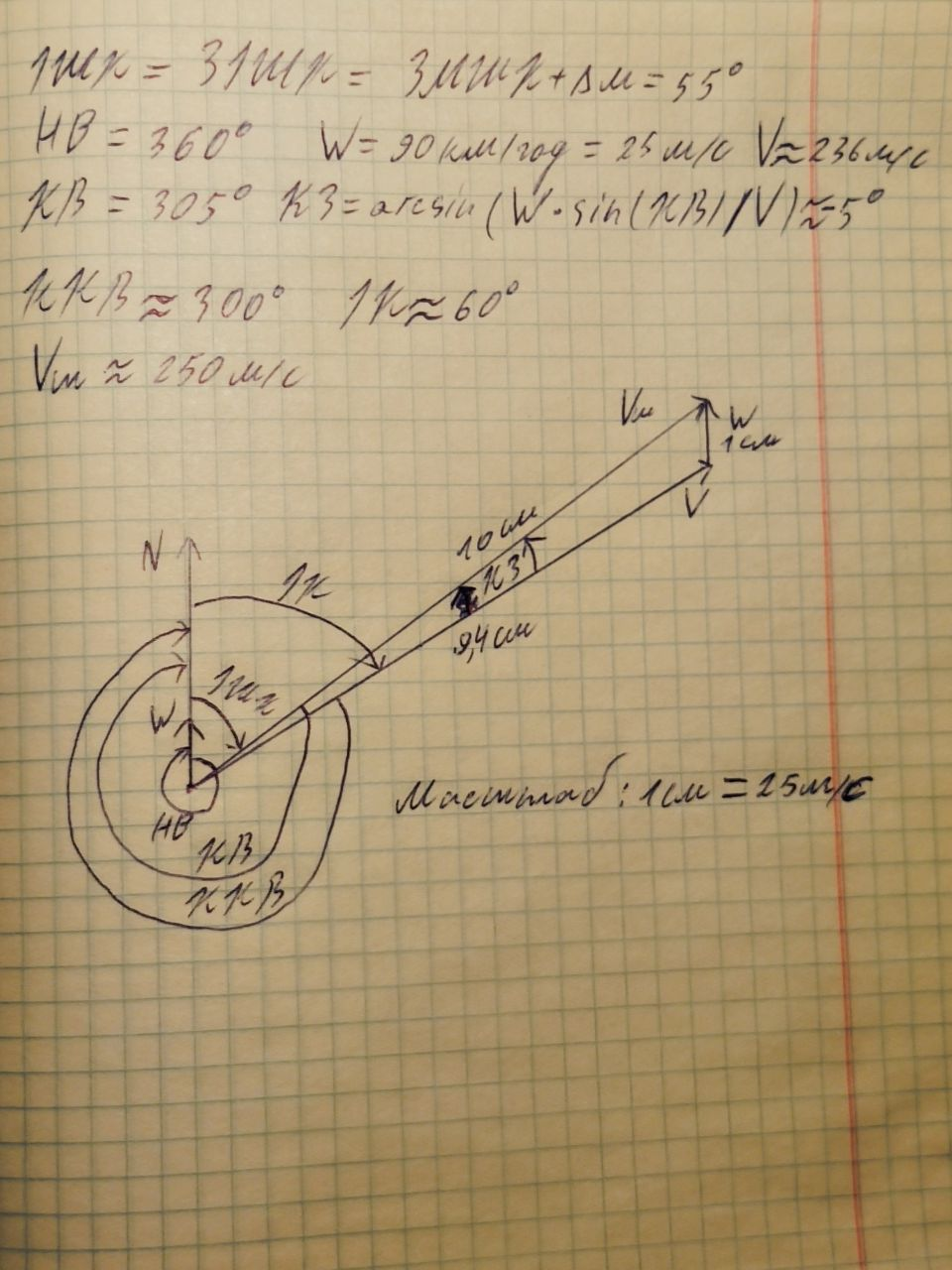
2. Зробити дослідження навігаційних елементів польоту та здійснити зчислення та прокладку шляху.

**Варіант завдання**

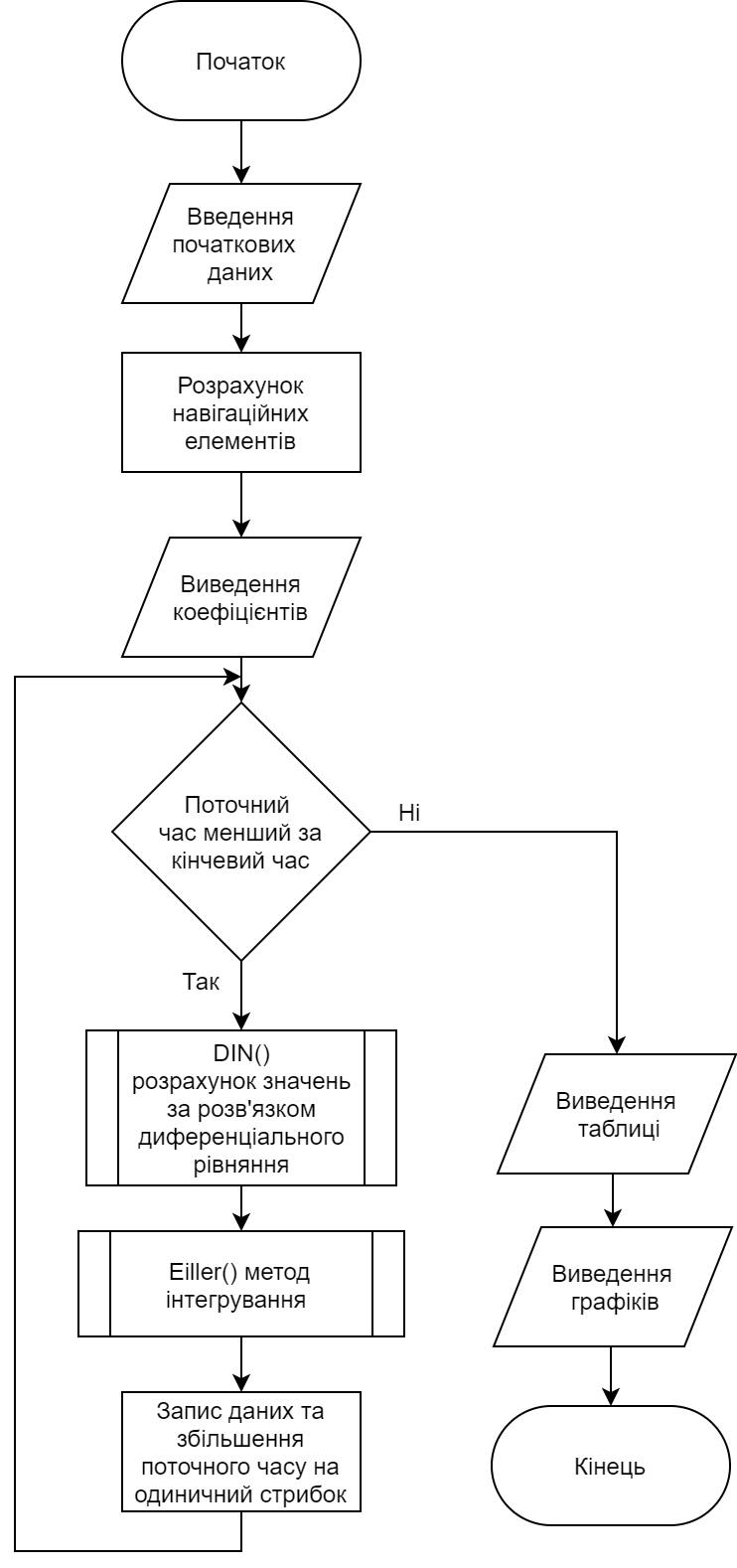
****

**Навігаційні елементи польоту у вихідному**

**пункті маршруту визначені графічним шляхом**

****

**Алгоритм роботи**



**Код програми**

public class Rozrakhunok

{

public double T = 0, DT = 1, TD = 0, TF = 1800, DD = 60;

public double V = 850/3.6, dVinstr = 0, dVst = -24/3.6, dVa = -1.1/3.6, di = 180, W = 90/3.6,

dm = 5, dk = 1, ZMSK = 50, Heszad = 10050;

public double p0 = 0.1249, pn, R = 6373000, Vi, Vpr;

public double wgyx = 1 / 3600;

public double ZISK, HBi, ISKvpm, KV, KZ, KKV, IK, MK, KK, psig, IK2, psigo, Vs, fiso, alphaso;

double[] X = new double[9];

double[] Y = new double[9];

public List<double> Time = new List<double>();

public List<double> massAlpha = new List<double>();

public List<double> massFi = new List<double>();

public List<double> massPsi = new List<double>();

public List<double> massIK = new List<double>();

public List<double> massVsx = new List<double>();

public List<double> massVsz = new List<double>();

public List<double> massX = new List<double>();

public List<double> graphTime = new List<double>();

public List<double> graphFi = new List<double>();

public List<double> graphAlpha = new List<double>();

public Rozrakhunok()

{

pn = (0.1249 - 0.0117\*Heszad + 0.000343 \* Heszad \* Heszad) / 1000000;

Vi = V \* Math.Sqrt(pn / p0);

Vpr = Vi - dVinstr - dVa - dVst;

ZISK = ZMSK + dm;

HBi = di + 180;

ISKvpm = ZISK;

KV = (HBi - ZISK);

KZ = Math.Asin(W \* Math.Sin(KV / 57.29577951) / V) \* 57.29577951;

KKV = KV + KZ;

IK = HBi - KKV;

MK = IK - dm;

KK = MK - dk;

psigo = ZISK - KZ;

Vs = V \* Math.Cos(KZ / 57.29577951) + W \* Math.Cos(KV / 57.29577951);

fiso = 50.3521148;

alphaso = 30.8960442;

Y[0] = fiso;

Y[1] = alphaso;

while (T <= TF+DT)

{

DIN();

Eiller();

if (T >= TD-T)

{

Time.Add(T);

massAlpha.Add(Y[0]);

massFi.Add(Y[1]);

massPsi.Add(psig);

massIK.Add(IK2);

massX.Add(Y[4]);

massVsx.Add(Y[2]);

massVsz.Add(Y[3]);

TD = TD + DD;

}

graphTime.Add(T);

graphFi.Add(Y[2]);

graphAlpha.Add(Y[3]);

T = T + DT;

}

}

public void DIN()

{

psig = psigo + Y[5];

IK2 = psig - Y[4];

double ISHK = IK2 + KZ;

X[0] = 57.3 \* X[2] / (R + Heszad);

X[1] = 57.3 \* X[3] / ((R + Heszad) \* Math.Cos(Y[0] / 57.29577951));

X[2] = Vs \* Math.Cos(ISHK/ 57.29577951);

X[3] = Vs \* Math.Sin(ISHK/ 57.29577951);

X[4] = -57.3 \* (X[3] / (R + Heszad)) \* Math.Tan(Y[0] / 57.29577951f);

X[5] = wgyx;

}

public void Eiller()

{

for (int i = 0; i < 6; i++)

{

Y[i] = Y[i] + X[i] \* DT;

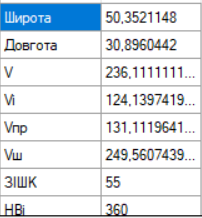
}

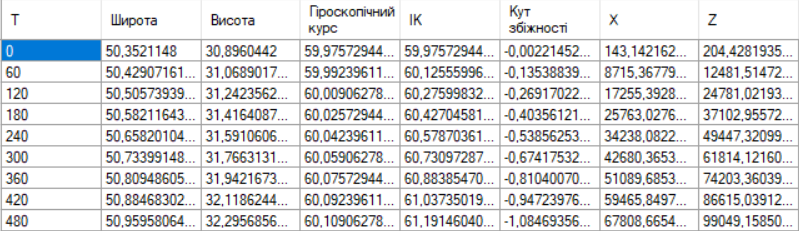
}

}

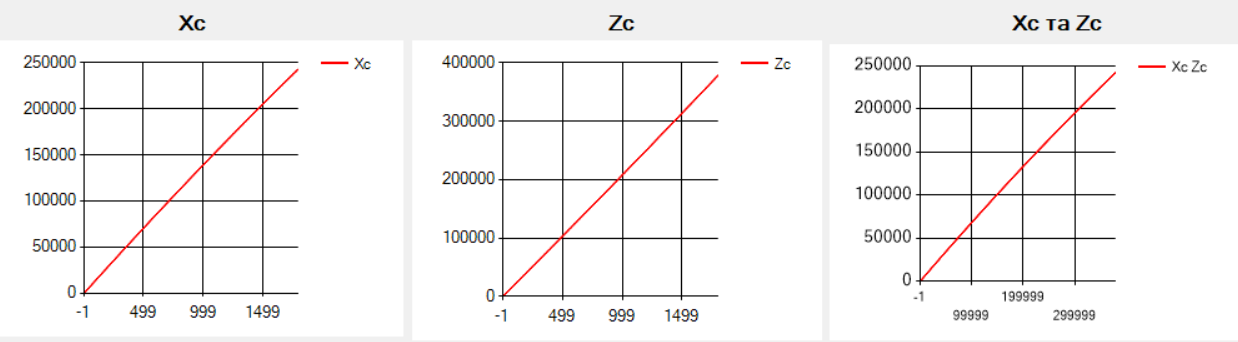
}

**Навігаційні елементи**

****

****

**Дослідження з уходом гіроскопа**



**Абсолютні і відносні похибки визначення навігаційних елементів польоту графічним шляхом**

Для КВ:

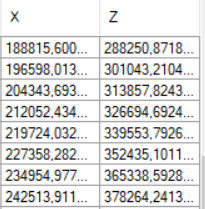
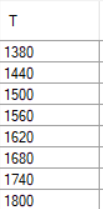
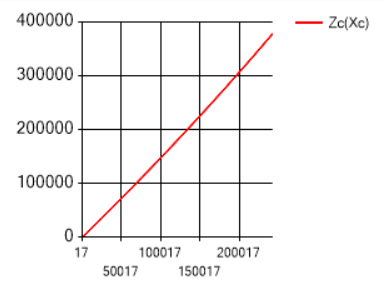
Для КЗ:

Для ККВ:

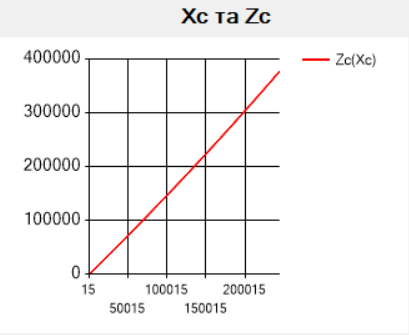
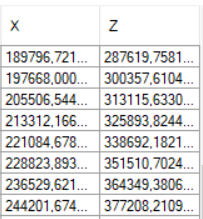
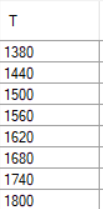
Для ІК:

Для Vш:

**Фактичний шлях руху літака**



**Заданий шлях руху літака (без уходу гіроскопу)**

**Абсолютне відхилення фактичного місця знаходження літака наприкінці маршруту від заданого**

**Висновок про особливості визначення навігаційних елементів польоту різними методами і вплив похибки роботи курсової системи на точність літаководіння по маршруту**

За результатами досліджень можна стверджувати, що при моделюванні з ПЕОМ навігаційні елементи є більш точним, ніж при графічному розрахунку, що значно впливає на точність курсової системи. Похибка курсової системи є не значеною, але через певний час система потребує скидання деяких параметрів та гіроскопічного датчику для того, щоб уникнути великого накопичення похибки.