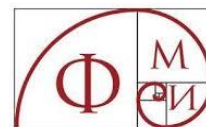


Софийски университет “Св. Климент
Охридски”
Факултет по математика и информатика



**Курсов проект по Небесна Механика
летен семестър 2022/2023 г.**

*Изготвил: Кристиян Войнски
Специалност: Информационни системи
Курс: 3, Група: 2
Факултетен номер: 72035*

Юни 2023
гр. София

Задача 1:

Пресметнете координатите и скоростите на планетите в деня, в който сте родени.

Орбитата на планетата зависи от 6 елемента в задачата на Кеплер, като те са:

- a** - дължина на голямата полуос
- e** - ексцентрицитет
- i** - наклонение на плоскостта на орбитата
- l** - средна аномалия за даден момент **t**
- g + θ** - дължина на перихелия
- θ** - дължина на възела

Отбелязваме, че 5 от тези елементи са константни величини, като единствено средната аномалия **l** е линейна функция, зависеща от времето **t**.

Допълнителен елемент е ексцентричната аномалия **u**, като в сила е уравнението на Кеплер: **l = u - e * sin u**

Ексцентрицитетът е характеризира сплеснатостта на елипсата, като формулата е следната:

$$e = \sqrt{1 - \frac{b^2}{a^2}}$$

и където **b** е дължината на малката полуос.

Връзката на елиптичните елементи с декартовите координати е следната:

$$\begin{pmatrix} x \\ y \\ z \end{pmatrix} = \begin{pmatrix} \cos \theta & -\sin \theta & 0 \\ \sin \theta & \cos \theta & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos i & -\sin i \\ 0 & \sin i & \cos i \end{pmatrix} \begin{pmatrix} \cos g & -\sin g & 0 \\ \sin g & \cos g & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} z_1 \\ z_2 \\ 0 \end{pmatrix}$$

планета	$g + \theta$	θ	i	e	a
Меркурий	1.3511847	0.8431191	0.1221960	0.2056306	0.3870989
Венера	2.2945197	1.3376520	0.0592186	0.0067732	0.7233199
Земя	1.7958565	-0.1964356	0.0000008	0.0167102	1.0000001
Марс	5.8620457	0.8648700	0.0322828	0.0934123	1.5236623
Юпитер	0.2573727	1.7541461	0.0227702	0.0483926	5.2033630
Сатурн	1.6124238	1.9836956	0.0433400	0.0541506	9.5370703
Уран	2.9823761	1.2948990	0.0134297	0.0471677	19.191263
Нептун	0.7845002	2.2978117	0.0308621	0.0085858	30.068963
Плутон	3.9087201	1.9241827	0.2990283	0.2488076	39.481686

(използвани данни от материалите по Небесна Механика)

Следващата стъпка е да обърнем θ (дължината на възела), $g + \theta$ (дължината на перихелия) и i (наклонението на плоскостта на орбитата) в радиани.

Стойностите на μ за планетите са следните:

Планета	Меркурий	Венера	Земя	Марс
μ	1 / 6023600	1 / 408523	1 / 328900.5	1 / 3098708

Юпитер	Сатурн	Уран	Нептун	Плутон
1 / 1047.34	1 / 3497.8	1 / 22902.9	1 / 19402	1 / 135000000

$\gamma = 1 + \mu$, където γ е гравитационна константа.

Величината n наричаме средно движение, като то представлява момента на преминаване през перихелия на планетата.

$$n = \sqrt{\frac{\gamma}{a^3}}$$

Връзката между средната и ексцентричната аномалия наричаме уравнение на Кеплер.

$$l = u - e * \sin u$$

Изчисляваме променливата t , която пази времето от рождената ни дата до 2000 г.
 Рождена дата: 27.09.2001 г., като разликата между 27.09.2001 и 01.01.2000 г. в дни е 635 дни. Променливата t приема стойността **635 / 365.25**.

$$Q = \begin{pmatrix} \cos \theta & -\sin \theta & 0 \\ \sin \theta & \cos \theta & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos i & -\sin i \\ 0 & \sin i & \cos i \end{pmatrix} \begin{pmatrix} \cos g & -\sin g & 0 \\ \sin g & \cos g & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

$$Q = \begin{pmatrix} \cos \theta \cos g - \sin \theta \sin g \cos i & -\cos \theta \sin g - \sin \theta \cos g \cos i & \sin \theta \sin i \\ \sin \theta \cos g + \cos \theta \sin g \cos i & -\sin \theta \sin g + \cos \theta \cos g \cos i & -\cos \theta \sin i \\ \sin g \sin i & \cos g \sin i & \cos i \end{pmatrix}$$

където $\theta, g \in [0, 2\pi)$ и $i \in [0, \pi)$

Таблица с резултати от приложения алгоритъм за всяка планета

	Меркурий	Венера	Земя	Марс	Юпитер	Сатурн	Уран	Нептун	Плутон
r	0.2756	-0.3576	0.9990	1.1572	-0.2248	3.2665	16.0093	18.4333	-8.0126
	-0.3154	0.6228	0.0790	-0.7558	5.1329	8.4560	-11.9483	-23.7918	-28.8840
	-0.0510	0.0291	0.0000	-0.0443	-0.0162	-0.2771	-0.2517	0.0651	5.4085
v	0.9076	-1.0232	-0.0944	0.4758	-0.4440	-0.3194	0.1350	0.1431	0.1795
	1.1545	-0.5915	0.9934	0.7505	0.0012	0.1163	0.1726	0.1127	-0.0789
	0.0110	0.0510	0.0000	0.0040	0.0099	0.0107	-0.0011	-0.0056	-0.0435
 r 	0.2756	0.3576	0.9990	1.1572	0.2248	3.2665	16.0093	18.4333	8.0126
 v 	0.9076	1.0232	0.0944	0.4758	0.4440	0.3194	0.1350	0.1431	0.1795

Задача 2: Пресметнете елементите на Делоне и Поанкаре (от първи и втори вид) в деня, в който сте родени.

Елементите на Делоне (**L**, **G**, **Θ**, **I**, **g**, **θ**) се изразяват чрез орбиталните елементи, дефинирани в предходната задача.

Имаме следното, че съответно двойките (**I**, **L**), (**G**, **g**), (**Θ**, **θ**) са спрегнати канонични променливи, като те приемат стойностите:

$$L = \mu \sqrt{\gamma a}$$

$$G = \mu \sqrt{\gamma a (1 - e^2)} = L (1 - e^2)$$

$$\Theta = \mu \sqrt{\gamma a (1 - e^2)} \cos i = G \cos i$$

Като при това и в двата случая стойностите на **I**, **g** и **θ** съвпадат.

Елементите на Делоне (**L**, **G**, **Θ**, **I**, **g**, **θ**) са константни величини с хамилтони:

$$\hat{H} = \frac{\mu^3 \gamma^2}{2L^2}$$

Нужно е да обърнем **θ** и **i** в радиани.

Дефинираме променливата **To**, като тя представлява момента на преминаване на перихелия на планетата, което дава информация за началото на епохата.

$$l = \sqrt{\gamma} a^{-\frac{3}{2}} (t - T_o)$$

$$u = l + e \sin(l + e \sin(l + e \sin(l)))$$

$$n = \sqrt{\frac{\gamma}{a^3}}$$

Изчисляваме променливата **t**, която пази времето от рождената ни дата до 2000 г.
 Рождена дата: 27.09.2001 г., като разликата между 27.09.2001 и 01.01.2000 г. в дни е 635 дни. Променливата **t** приема стойността **635 / 365.25**.

Чрез следното равенство ще можем да изразим елементите от двете системи на Поанкаре:
 $\lambda = l + g + \theta$

Първата система от шест елемента, характеризираща орбитите на планетите:

$$\begin{pmatrix} L & L - G & G - \Theta \\ l + g + \theta & -g - \theta & -\theta \end{pmatrix}$$

Втората система от шест елемента:

$$\begin{pmatrix} L & \sqrt{2(L - G)} \cos(g + \theta) & \sqrt{2(G - \Theta)} \cos \theta \\ l + g + \theta & -\sqrt{2(L - G)} \sin(g + \theta) & -\sqrt{2(G - \Theta)} \sin \theta \end{pmatrix}$$

Елементи на Делоне

	L	G	Θ	l	g	θ	H
Меркурий	1.0328e-7	1.0108e-7	1.0033e-7	4.8424e+1	5.0836e-1	8.4352e-1	-2.1449e-7
Венера	2.0814e-6	2.0814e-6	2.0777e-6	1.8648e+1	9.5859e-1	1.3383e+0	-1.6928e-6
Земя	3.0404e-6	3.0400e-6	3.0400e-6	1.0880e+1	1.7966e+0	0.0000e+0	-1.5202e-6
Марс	3.9826e-7	3.9654e-7	3.9633e-7	6.1503e+0	-1.2829e+0	8.6497e-1	-1.0595e-7
Юпитер	2.1787e-3	2.1762e-3	2.1757e-3	1.2639e+0	-1.4965e+0	1.7536e+0	-9.1860e-5
Сатурн	8.8298e-4	8.8174e-4	8.8091e-4	-3.7333e-1	-3.6764e-1	1.9838e+0	-1.4995e-5
Уран	1.9127e-4	1.9106e-4	1.9104e-4	2.6133e+0	1.6919e+0	1.2918e+0	-1.1377e-6
Нептун	2.8263e-4	2.8262e-4	2.8249e-4	-1.6805e+0	-1.5153e+0	2.3001e+0	-8.5709e-7
Плутон	4.6544e-8	4.5090e-8	4.3088e-8	3.0340e-1	1.9856e+0	1.9252e+0	-9.3807e-11

Първа система на Поанкаре

	L	L - G	G - Θ	l + g + θ	- g - θ	- θ
Меркурий	1.0328e-7	2.1934e-9	7.5431e-10	8.8040e+6	-1.3519e+0	-8.4352e-1
Венера	2.0814e-6	3.7465e-11	3.6506e-9	3.4478e+6	-2.2969e+0	-1.3383e+0
Земя	3.0404e-6	3.8920e-10	0.0000e+0	1.2677e+1	-1.7966e+0	0.0000e+0
Марс	3.9826e-7	1.7260e-9	2.0646e-0	5.7324e+0	4.1788e-1	-8.6497e-1
Юпитер	2.1787e-3	2.5114e-6	5.6359e-7	1.5210e+0	-2.5705e-1	-1.7536e+0
Сатурн	8.8298e-4	1.2410e-6	8.2918e-7	1.2428e+0	-1.6161e+0	-1.9838e+0
Уран	1.9127e-4	2.1137e-7	1.7343e-8	5.5970e+0	-2.9837e+0	-1.2918e+0
Нептун	2.8263e-4	9.0444e-9	1.3485e-7	-8.9578e-1	-7.8477e-1	-2.3001e+0
Плутон	4.6544e-8	1.4540e-9	2.0026e-9	4.2141e+0	-3.9107e+0	-1.9252e+0

Втора система на Поанкаре

	L	ε	p	λ = l + g + θ	η	q
Меркурий	1.0328e-7	1.4384e-5	2.5823e-5	8.8040e+6	-6.4652e-5	-2.9014e-5
Венера	2.0814e-6	-5.7473e-6	1.9688e-5	3.4478e+6	-6.4729e-6	-8.3148e-5
Земя	3.0404e-6	-6.2462e-6	0.0000e+0	1.2677e+1	-2.7192e-5	0.0000e+0
Марс	3.9826e-7	5.3698e-5	1.3181e-5	5.7324e+0	2.3844e-5	-1.5466e-5
Юпитер	2.1787e-3	2.1675e-3	-1.9299e-4	1.5210e+0	-5.6977e-4	-1.0440e-3
Сатурн	8.8298e-4	-7.1412e-5	-5.1684e-4	1.2428e+0	-1.5738e-3	-1.1795e-3
Уран	1.9127e-4	-6.4210e-4	5.1285e-5	5.5970e+0	-1.0223e-4	-1.7904e-4
Нептун	2.8263e-4	9.5162e-5	-3.4604e-4	-8.9578e-1	-9.5042e-5	-3.8724e-4
Плутон	4.6544e-8	-3.8747e-5	-2.1959e-5	4.2141e+0	3.7507e-5	-5.9354e-5

Решение на задача 1 като код:

```
#include <iostream>
#include <cmath>
#include <vector>
#include <iomanip>

using namespace std;

vector<vector<double>> multiplyMatrices(const
vector<vector<double>>& matrix, const vector<vector<double>>&
other) {
    vector<vector<double>> result(matrix.size(),
vector<double>(other[0].size(), 0));

    for (int i = 0; i < matrix.size(); i++) {
        for (int j = 0; j < other[0].size(); j++) {
            double sum = 0;
            for (int k = 0; k < matrix[0].size(); k++) {
                sum += matrix[i][k] * other[k][j];
            }
            result[i][j] = sum;
        }
    }

    return result;
}

vector<vector<double>> multiplyMatrixByScalar(const
vector<vector<double>>& matrix, double scalar) {
    vector<vector<double>> result(matrix.size(),
vector<double>(matrix[0].size(), 0));

    for (int i = 0; i < matrix.size(); i++) {
        for (int j = 0; j < matrix[i].size(); j++) {
            result[i][j] = matrix[i][j] * scalar;
        }
    }

    return result;
}

double vectorNorm(const vector<double>& vec) {
    double sum = 0;

    for (int i = 0; i < vec.size(); i++) {
        sum += vec[i] * vec[i];
    }
}
```

```

    return sqrt(sum);
}

void result(double a, double e, double i, double L, double w,
double Omega, double miu, double t) {
    double tita = (Omega * M_PI) / 180;
    double g = ((w - Omega) * M_PI) / 180;
    double iCopy = (i * M_PI) / 180;

    vector<vector<double>> Tita = {
        {cos(tita), -sin(tita), 0},
        {sin(tita), cos(tita), 0},
        {0, 0, 1}
    };

    vector<vector<double>> I = {
        {1, 0, 0},
        {0, cos(iCopy), -sin(iCopy)},
        {0, sin(iCopy), cos(iCopy)}
    };

    vector<vector<double>> G = {
        {cos(g), -sin(g), 0},
        {sin(g), cos(g), 0},
        {0, 0, 1}
    };

    vector<vector<double>> Q =
multiplyMatrices(multiplyMatrices(Tita, I), G);

    double gama = 1 + miu;
    double n = sqrt(gama / pow(a, 3));
    double to = (((w - L) / n) * M_PI) / 180;
    double l = n * (t * 2 * M_PI - to);
    double u = l + e * sin(l + e * sin(l + e * sin(l)));

    vector<vector<double>> r =
multiplyMatrices(multiplyMatrixByScalar(Q, a), {
        {cos(u) - e},
        {sin(u) * sqrt(1 - e * e)},
        {0}
    });

    vector<vector<double>> v =
multiplyMatrixByScalar(multiplyMatrices(Q, {
        {-sin(u)},
        {cos(u) * sqrt(1 - e * e)},

```



```

        {0}
    )), (a * n) / (1 - e * cos(u)));

    cout << "r:" << endl;
    for (const auto& row : r) {
        for (const auto& val : row) {
            cout << val << " ";
        }
        cout << endl;
    }

    cout << "v:" << endl;
    for (const auto& row : v) {
        for (const auto& val : row) {
            cout << val << " ";
        }
        cout << endl;
    }

    cout << "Norm of r: " << vectorNorm(r[0]) << endl;
    cout << "Norm of v: " << vectorNorm(v[0]) << endl;
}

int main() {

    cout << fixed << setprecision(4);
    double time = 635.0 / 365.25;

    vector<vector<double>> planets = {
        {0.387, 0.205, 7.004, 252.250, 77.457, 48.330, 1 /
6023600.0},
        {0.723, 0.006, 3.394, 181.979, 131.602, 76.679, 1 /
408523.0},
        {1, 0.016, 0, 100.464, 102.937, 0, 1 / 328900.5},
        {1.523, 0.093, 1.849, -4.553, -23.943, 49.559, 1 /
3098708.0},
        {5.202, 0.048, 1.304, 34.396, 14.728, 100.473, 1 / 1047.34},
        {9.536, 0.053, 2.485, 49.954, 92.598, 113.662, 1 / 3497.8},
        {19.189, 0.047, 0.772, 313.238, 170.954, 74.016, 1 /
22902.9},
        {30.069, 0.008, 1.770, -55.120, 44.964, 131.784, 1 /
19402.0},
        {39.482, 0.248, 17.140, 238.929, 224.068, 110.303, 1 /
135000000.0}
    };

    for (const auto& planet : planets) {

```

```

        result(planet[0], planet[1], planet[2], planet[3], planet[4],
planet[5], planet[6], time);
    }

    return 0;
}

```

Решение на задача 2 като код:

```

#include <iostream>
#include <cmath>
#include <iomanip>
using namespace std;

void task02(double a, double e, double i, double L, double w, double
Omega, double miu, double t) {
    double iCopy = (i * M_PI) / 180;
    double n = sqrt(1 / pow(a, 3));
    double to = ((w - L) / n) * M_PI / 180;

    double gamma = 1 + miu;
    double capL = miu * sqrt(gamma * a);
    double capG = capL * sqrt(1 - e * e);
    double capTheta = capG * cos(iCopy);

    double l = n * (t * 2 * M_PI - to);
    double g = ((w - Omega) * M_PI) / 180;
    double theta = (Omega * M_PI) / 180;
    double H = (-miu * gamma) / (2 * a);

    double first11 = capL;
    double first12 = capL - capG;
    double first13 = capG - capTheta;
    double first21 = l + g + theta;
    double first22 = -g - theta;
    double first23 = -theta;

    double second11 = first11;
    double second12 = sqrt(2 * (capL - capG)) * cos(g + theta);
    double second13 = sqrt(2 * (capG - capTheta)) * cos(theta);
    double second21 = first21;
    double second22 = -sqrt(2 * (capL - capG)) * sin(g + theta);
    double second23 = -sqrt(2 * (capG - capTheta)) * sin(theta);
}

```

```

    cout << first11 << " " << first12 << " " << first13 << " " << first21
<< " " << first22 << " " << first23 << endl;
    cout << second11 << " " << second12 << " " << second13 << " " <<
second21 << " " << second22 << " " << second23 << endl;
}

int main() {
    double time = 635.0 / 365.25;

    cout << scientific << setprecision(4);

    double planets[9][7] = {
        {0.387, 0.205, 7.004, 252.250, 77.457, 48.330, 1 / 6023600.0},
        {0.723, 0.006, 3.394, 181.979, 131.602, 76.679, 1 / 408523.0},
        {1, 0.016, 0, 100.464, 102.937, 0, 1 / 328900.5},
        {1.523, 0.093, 1.849, -4.553, -23.943, 49.559, 1 / 3098708.0},
        {5.202, 0.048, 1.304, 34.396, 14.728, 100.473, 1 / 1047.34},
        {9.536, 0.053, 2.485, 49.954, 92.598, 113.662, 1 / 3497.8},
        {19.189, 0.047, 0.772, 313.238, 170.954, 74.016, 1 / 22902.9},
        {30.069, 0.008, 1.770, -55.120, 44.964, 131.784, 1 / 19402.0},
        {39.482, 0.248, 17.140, 238.929, 224.068, 110.303, 1 /
135000000.0}
    };

    for (auto & planet : planets) {
        task02(planet[0], planet[1], planet[2], planet[3], planet[4],
planet[5], planet[6], time);
    }

    return 0;
}

```