

Міністерство освіти і науки України
Національний технічний університет України «Київський
політехнічний інститут імені Ігоря Сікорського» Інститут
прикладного системного аналізу
Кафедра математичних методів системного аналізу

ЗВІТ

про виконання Лабораторна робота № 1

з дисципліни «Проектування та аналіз обчислювальних
алгоритмів»

Варіант №23

Виконав:

Студент II курсу

групи КА-86

Фролкін В. Ю.

Варіант № 23

Перевірив:

Севастьянов В.

Завдання:

№ 841

Провести площину через перпендикуляри, проведені із точки $(-3, 2, 5)$ до площин

$$4x + y - 3z + 13 = 0 \text{ та } x - 2y + z - 11 = 0$$

Розв'язок:

Handwritten solution for finding the plane perpendicular to two given planes:

$$4x + y - 3z + 13 = 0 \quad \text{и} \quad x - 2y + z - 11 = 0$$

Point $A(-3, 2, 5)$

Normal vectors of the planes: $n_1(4, 1, -3)$ and $n_2(1, -2, 1)$ - норм. вектори площин

$$\vec{n} = \begin{vmatrix} i & j & k \\ 4 & 1 & -3 \\ 1 & -2 & 1 \end{vmatrix} = i(1-6) - j(4+3) + k(-8-1) = -5i - 7j - 9k$$

Уравнение плоскости, которая проходит через \perp точки

$$-5(x+3) - 7(y-2) - 9(z-5) = 0$$

или

$$5x + 7y + 9z - 44 = 0$$

Множиною вхідних даних для даної задачі є 2 рівняння площин та координати точки. Коректним форматом даних є площина у вигляді: $Ax + By + Cz + D = 0$ та координати точки.

Множиною вихідних даних є рівняння площини.

Лістинг:

```
import numpy as np

import re

def take_norm_vector(data):

    pat = r'([+-]?\d*)x([+-]?\d*)y([+-]?\d*)z[+-]?\d*=0'

    flat = re.search(pat, data)

    if not flat:
```

```

        raise ValueError('Incorrect input, check equation')
norm_vector = []
# get coordinates of normal vector from regex
for i in flat.groups():
    if i == '+' or i == '-':
        norm_vector.append(1.0)
    elif i == '-':
        norm_vector.append(-1.0)
    else:
        norm_vector.append(float(i))
return norm_vector

def solve(flat1, flat2, dot):
    # get two normal vectors
    n1 = take_norm_vector(flat1)
    n2 = take_norm_vector(flat2)
    # multiply them to get normal vector for our flat
    norm_vector = list(np.cross(n1, n2))
    result = '{}(x - {}) + {}(y - {}) + {}(z - {}) = 0'.format(
        norm_vector[0], dot[0], norm_vector[1], dot[1], norm_vector[2], dot[2])
    # make result looks better
    result = result.replace('-', '+ -')
    result = result.replace('+ -', '- ')
    return result

```

Результат роботи програми:

- 3 коректними даними

```

>>> from alg1 import solve
>>> flat1 = '4x+y-3z+13=0'
>>> flat2 = 'x-2y+z-11=0'
>>> dot = [-3, 2, 5]
>>> result = solve(flat1, flat2, dot)

```

```
>>> print(result)
```

```
-5.0(x + 3) - 7.0(y - 2) - 9.0(z - 5) = 0
```

- 3 некоректными данными

```
>>> flat1 = 'xyz'
```

```
>>> flat2 = 'x-2y+z-11=0'
```

```
>>> dot = [-3, 2, 5]
```

```
>>> result = solve(flat1, flat2, dot)
```

```
ValueError: Incorrect input, check equation
```