

Отчет по Лабораторному практикуму №1

Тема: “Алгоритмы «гладкой» однокритериальной оптимизации”

Студент группы ДМП-101уцп

Ерохин Никита

Условие: ЛПР выбирает адвоката для представления его интересов в суде. В качестве альтернатив у него имеются адвокаты А1, А2, А3 и А4. В качестве критериев выступают: Стоимость (К1), Авторитет (К2), Репутация (К3), Специализация (К4). Оценки показателей привлекательностей каждого адвоката (альтернативы) по каждому критерию, а также веса критериев по десятибалльной системе представлены матрицей 1 в прилагемом файле.

Решение:

В рис. 1 приведен исходный код программы и результат ее выполнения. После подсчета массива, являющимся результатом вычисления функции полезности можно сделать вывод, что в условиях поставленной задачи самым лучшим выбором для ЛПР является выбор адвоката А1.

```
[9]: import numpy as np
    from copy import deepcopy

[10]: #Support functions to normalize criteria
    def normalize_max_crit(matrix: np.array, crit_num: int) -> np.array:
        m = deepcopy(matrix).transpose()
        mini = min(m[crit_num])
        div = max(m[crit_num]) - min(m[crit_num])
        m[crit_num] = [(i - mini) / div for i in m[crit_num]]
        return m.transpose()

    def normalize_min_crit(matrix: np.array, crit_num: int) -> np.array:
        m = deepcopy(matrix).transpose()
        maxi = max(m[crit_num])
        div = max(m[crit_num]) - min(m[crit_num])
        m[crit_num] = [(maxi - i) / div for i in m[crit_num]]
        return m.transpose()

[11]: #input data
    matrix = np.array(
        [
            [3, 7, 2, 9],
            [8, 3, 6, 7],
            [4, 8, 3, 5],
            [9, 6, 5, 4]
        ]
    ).astype(float)
    weight = np.array([8, 9, 6, 7])

[12]: #normalizing criterias ad calculaiting payload function
    #(lowing cost, increasing authority, reputation and sepcialization)
    matrix = normalize_max_crit(normalize_max_crit(normalize_max_crit(normalize_min_crit(matrix, 0), 1), 2), 3)
    result = [sum(row * weight) for row in matrix]

[13]: result
[13]: [22.2, 11.533333333333333, 18.566666666666666, 9.899999999999999]
```

рис 1. Исходный код программы и результат выполнения