Изображение выглядит как эмблема, символ, герб, нашивка

Автоматически созданное описание

|  |
| --- |
| МИНОБРНАУКИ РОССИИ |
| Федеральное государственное бюджетное образовательное учреждение  высшего образования  **"МИРЭА - Российский технологический университет"**  **РТУ МИРЭА** |

**Институт** Информационных Технологий

**Кафедра** Вычислительной Техники

**ПРАКТИЧЕСКАЯ РАБОТА №2**

**по дисциплине**

**«Теория принятия решений»**

**Метод Электра II**

Студент группы: ИКБО-04-22 \_\_\_Кликушин В.И.\_\_\_\_\_ *(Ф. И.О. студента)*

Преподаватель \_\_Железняк Л.М.\_\_

*(Ф.И.О. преподавателя)*

|  |  |  |
| --- | --- | --- |
|  |  |  |

Москва 2024

**СОДЕРЖАНИЕ**

[ВВЕДЕНИЕ 3](#_Toc130799277)

[1 МЕТОД ЭЛЕКТРА II 4](#_Toc130799278)

[1.1 Выбор лучшего варианта 4](#_Toc130799279)

[1.2 Веса предпочтений 6](#_Toc130799280)

[1.3 Вывод 29](#_Toc130799281)

[1.4 Результат работы программы 29](#_Toc130799282)

[ЗАКЛЮЧЕНИЕ 31](#_Toc130799283)

[СПИСОК ИНФОРМАЦИОННЫХ ИСТОЧНИКОВ 32](#_Toc130799284)

[ПРИЛОЖЕНИЯ 33](#_Toc130799285)

ВВЕДЕНИЕ

Цель работы: изучить метод Электра II и научиться применять его в нахождении оптимального решения в выбранной предметной области.

Предметная область: выбор оптимального высшего учебного заведения.

Метод Электра II состоит из нескольких этапов. На первом этапе определяется множество решений и для каждого из N критериев определяется вес – число, характеризующее важность соответствующего критерия. На втором этапе для каждой пары альтернатив вычисляется P+ - сумма весов критериев, по которым одна альтернатива предпочтительнее другой, и P- - сумма весов критериев, по которым эта же альтернатива менее предпочтительна по сравнению с другой. На третьем этапе вычисляются отношения P+/ P-, и если полученное отношение больше 1, то оно сохраняется в матрицу, а если меньше или равно, то не сохраняется.

На основе полученной матрицы строится граф предпочтений, и если в нём обнаруживаются петли, то назначается порог, который отбрасывает слабые связи, т.е. те пары альтернатив, которые не сильно отличаются друг от друга. Если в графе не осталось петель и граф остался целостным, то выбирается та альтернатива, к которой не идёт ни одно ребро на графе. Она является оптимальной.

# 1 МЕТОД ЭЛЕКТРА II

## **1.1 Выбор лучшего варианта**

Составлена таблица критериев, по которым оцениваются университеты (Таблица 1).

*Таблица 1 – Таблица критериев для оценки альтернатив*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Критерии | Вес критерия | Шкала | Код | Стремление |
| Проходной балл (+) | 4 | Более 270  Более 250  Не более 250 | 15  10  5 | max |
| Количество бюджетных мест (-) | 5 | Более 100  Более 50  Не более 50 | 15  10  5 | min |
| Стоимость обучения (руб.) (+) | 2 | Более 350 тыс. рублей  Более 250 тыс. рублей  Не более 250 тыс. | 15  10  5 | max |
| Размер стипендии (руб.) (+) | 5 | Больше 2 тыс. рублей  Не более 2 тыс. рублей | 10  5 | max |
| Рейтинг университета (баллы) (+) | 2 | Больше 900 баллов  Больше 800 баллов  Не более 800 | 15  10  5 | max |
| Расстояние до общежития (км) (-) | 4 | Больше 10 км  Больше 5 км  Не более 5 км | 15  10  5 | min |

Составлена таблица оценок выбора оптимального технического университета. Для 10-ти альтернатив заполнена Таблица 2.

*Таблица 2 – Таблица оценок по критериям*

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| № | Варианты решений | Критерии | | | | | |
| Проходной балл (+) | Количество бюджетных мест (-) | Стоимость обучения (руб.) (+) | Размер стипендии (руб.) (+) | Рейтинг университета (баллы) (+) | Рассто-  яние до обще-жития (км) (-) |
| 1 | РТУ МИРЭА | 15 | 15 | 10 | 10 | 10 | 5 |
| 2 | МГТУ имени Н.Э. Баумана | 15 | 10 | 15 | 10 | 15 | 15 |
| 3 | ВШЭ | 15 | 15 | 15 | 10 | 10 | 5 |
| 4 | МАИ | 5 | 5 | 10 | 5 | 5 | 5 |
| 5 | ИТМО | 15 | 5 | 10 | 10 | 10 | 10 |
| 6 | СПбГУ | 15 | 5 | 10 | 10 | 15 | 15 |
| 7 | МТУСИ. | 10 | 10 | 10 | 5 | 5 | 5 |
| 8 | СГУ им. Чернышев-ского | 5 | 5 | 5 | 5 | 10 | 5 |
| 9 | НИЯУ МИФИ | 15 | 5 | 10 | 10 | 15 | 15 |
| 10 | МФТИ | 15 | 5 | 15 | 5 | 15 | 15 |
| Вес | | 4 | 5 | 2 | 5 | 2 | 4 |
| Стремление | | max | min | max | max | max | min |

**1.2 Веса предпочтений**

Рассмотрим альтернативы 1 и 2 (i = 1, j = 2):

P12 = 0 + 0 + 0 + 0 + 0 + 4 = 4;

N12 = 0 + 5 + 2 + 0 + 2 + 0 = 9;

D12 = P12 / N12 = 4/9 = 0.44 <= 1 - отбрасываем.

P21 = 0 + 5 + 2 + 0 + 2 + 0 = 9;

N21 = 0 + 0 + 0 + 0 + 0 + 4 = 4;

D21 = P21 / N21 = 9/4 = 2.25 > 1 - принимаем.

Рассмотрим альтернативы 1 и 3 (i = 1, j = 3):

P13 = 0 + 0 + 0 + 0 + 0 + 0 = 0;

N13 = 0 + 0 + 2 + 0 + 0 + 0 = 2;

D13 = P13 / N13 = 0/2 = 0 <= 1 - отбрасываем.

P31 = 0 + 0 + 2 + 0 + 0 + 0 = 2;

N31 = 0 + 0 + 0 + 0 + 0 + 0 = 0;

D31 = P31 / N31 = 2/0 = ∞ > 1 - принимаем.

Рассмотрим альтернативы 1 и 4 (i = 1, j = 4):

P14 = 4 + 0 + 0 + 5 + 2 + 0 = 11;

N14 = 0 + 5 + 0 + 0 + 0 + 0 = 5;

D14 = P14 / N14 = 11/5 = 2.2 > 1 - принимаем.

P41 = 0 + 5 + 0 + 0 + 0 + 0 = 5;

N41 = 4 + 0 + 0 + 5 + 2 + 0 = 11;

D41 = P41 / N41 = 5/11 = 0.45 <= 1 - отбрасываем.

Рассмотрим альтернативы 1 и 5 (i = 1, j = 5):

P15 = 0 + 0 + 0 + 0 + 0 + 4 = 4;

N15 = 0 + 5 + 0 + 0 + 0 + 0 = 5;

D15 = P15 / N15 = 4/5 = 0.8 <= 1 - отбрасываем.

P51 = 0 + 5 + 0 + 0 + 0 + 0 = 5;

N51 = 0 + 0 + 0 + 0 + 0 + 4 = 4;

D51 = P51 / N51 = 5/4 = 1.25 > 1 - принимаем.

Рассмотрим альтернативы 1 и 6 (i = 1, j = 6):

P16 = 0 + 0 + 0 + 0 + 0 + 4 = 4;

N16 = 0 + 5 + 0 + 0 + 2 + 0 = 7;

D16 = P16 / N16 = 4/7 = 0.57 <= 1 - отбрасываем.

P61 = 0 + 5 + 0 + 0 + 2 + 0 = 7;

N61 = 0 + 0 + 0 + 0 + 0 + 4 = 4;

D61 = P61 / N61 = 7/4 = 1.75 > 1 - принимаем.

Рассмотрим альтернативы 1 и 7 (i = 1, j = 7):

P17 = 4 + 0 + 0 + 5 + 2 + 0 = 11;

N17 = 0 + 5 + 0 + 0 + 0 + 0 = 5;

D17 = P17 / N17 = 11/5 = 2.2 > 1 - принимаем.

P71 = 0 + 5 + 0 + 0 + 0 + 0 = 5;

N71 = 4 + 0 + 0 + 5 + 2 + 0 = 11;

D71 = P71 / N71 = 5/11 = 0.45 <= 1 - отбрасываем.

Рассмотрим альтернативы 1 и 8 (i = 1, j = 8):

P18 = 4 + 0 + 2 + 5 + 0 + 0 = 11;

N18 = 0 + 5 + 0 + 0 + 0 + 0 = 5;

D18 = P18 / N18 = 11/5 = 2.2 > 1 - принимаем.

P81 = 0 + 5 + 0 + 0 + 0 + 0 = 5;

N81 = 4 + 0 + 2 + 5 + 0 + 0 = 11;

D81 = P81 / N81 = 5/11 = 0.45 <= 1 - отбрасываем.

Рассмотрим альтернативы 1 и 9 (i = 1, j = 9):

P19 = 0 + 0 + 0 + 0 + 0 + 4 = 4;

N19 = 0 + 5 + 0 + 0 + 2 + 0 = 7;

D19 = P19 / N19 = 4/7 = 0.57 <= 1 - отбрасываем.

P91 = 0 + 5 + 0 + 0 + 2 + 0 = 7;

N91 = 0 + 0 + 0 + 0 + 0 + 4 = 4;

D91 = P91 / N91 = 7/4 = 1.75 > 1 - принимаем.

Рассмотрим альтернативы 1 и 10 (i = 1, j = 10):

P110 = 0 + 0 + 0 + 5 + 0 + 4 = 9;

N110 = 0 + 5 + 2 + 0 + 2 + 0 = 9;

D110 = P110 / N110 = 9/9 = 1 <= 1 - отбрасываем.

P1010 = 0 + 5 + 2 + 0 + 2 + 0 = 9;

N1010 = 0 + 0 + 0 + 5 + 0 + 4 = 9;

D101 = P101 / N101 = 9/9 = 1 <= 1 - отбрасываем.

Рассмотрим альтернативы 2 и 3 (i = 2, j = 3):

P23 = 0 + 5 + 0 + 0 + 2 + 0 = 7;

N23 = 0 + 0 + 0 + 0 + 0 + 4 = 4;

D23 = P23 / N23 = 7/4 = 1.75 > 1 - принимаем.

P32 = 0 + 0 + 0 + 0 + 0 + 4 = 4;

N32 = 0 + 5 + 0 + 0 + 2 + 0 = 7;

D32 = P32 / N32 = 4/7 = 0.57 <= 1 - отбрасываем.

Рассмотрим альтернативы 2 и 4 (i = 2, j = 4):

P24 = 4 + 0 + 2 + 5 + 2 + 0 = 13;

N24 = 0 + 5 + 0 + 0 + 0 + 4 = 9;

D24 = P24 / N24 = 13/9 = 1.44 > 1 - принимаем.

P42 = 0 + 5 + 0 + 0 + 0 + 4 = 9;

N42 = 4 + 0 + 2 + 5 + 2 + 0 = 13;

D42 = P42 / N42 = 9/13 = 0.69 <= 1 - отбрасываем.

Рассмотрим альтернативы 2 и 5 (i = 2, j = 5):

P25 = 0 + 0 + 2 + 0 + 2 + 0 = 4;

N25 = 0 + 5 + 0 + 0 + 0 + 4 = 9;

D25 = P25 / N25 = 4/9 = 0.44 <= 1 - отбрасываем.

P52 = 0 + 5 + 0 + 0 + 0 + 4 = 9;

N52 = 0 + 0 + 2 + 0 + 2 + 0 = 4;

D52 = P52 / N52 = 9/4 = 2.25 > 1 - принимаем.

Рассмотрим альтернативы 2 и 6 (i = 2, j = 6):

P26 = 0 + 0 + 2 + 0 + 0 + 0 = 2;

N26 = 0 + 5 + 0 + 0 + 0 + 0 = 5;

D26 = P26 / N26 = 2/5 = 0.4 <= 1 - отбрасываем.

P62 = 0 + 5 + 0 + 0 + 0 + 0 = 5;

N62 = 0 + 0 + 2 + 0 + 0 + 0 = 2;

D62 = P62 / N62 = 5/2 = 2.5 > 1 - принимаем.

Рассмотрим альтернативы 2 и 7 (i = 2, j = 7):

P27 = 4 + 0 + 2 + 5 + 2 + 0 = 13;

N27 = 0 + 0 + 0 + 0 + 0 + 4 = 4;

D27 = P27 / N27 = 13/4 = 3.25 > 1 - принимаем.

P72 = 0 + 0 + 0 + 0 + 0 + 4 = 4;

N72 = 4 + 0 + 2 + 5 + 2 + 0 = 13;

D72 = P72 / N72 = 4/13 = 0.31 <= 1 - отбрасываем.

Рассмотрим альтернативы 2 и 8 (i = 2, j = 8):

P28 = 4 + 0 + 2 + 5 + 2 + 0 = 13;

N28 = 0 + 5 + 0 + 0 + 0 + 4 = 9;

D28 = P28 / N28 = 13/9 = 1.44 > 1 - принимаем.

P82 = 0 + 5 + 0 + 0 + 0 + 4 = 9;

N82 = 4 + 0 + 2 + 5 + 2 + 0 = 13;

D82 = P82 / N82 = 9/13 = 0.69 <= 1 - отбрасываем.

Рассмотрим альтернативы 2 и 9 (i = 2, j = 9):

P29 = 0 + 0 + 2 + 0 + 0 + 0 = 2;

N29 = 0 + 5 + 0 + 0 + 0 + 0 = 5;

D29 = P29 / N29 = 2/5 = 0.4 <= 1 - отбрасываем.

P92 = 0 + 5 + 0 + 0 + 0 + 0 = 5;

N92 = 0 + 0 + 2 + 0 + 0 + 0 = 2;

D92 = P92 / N92 = 5/2 = 2.5 > 1 - принимаем.

Рассмотрим альтернативы 2 и 10 (i = 2, j = 10):

P210 = 0 + 0 + 0 + 5 + 0 + 0 = 5;

N210 = 0 + 5 + 0 + 0 + 0 + 0 = 5;

D210 = P210 / N210 = 5/5 = 1 <= 1 - отбрасываем.

P1020 = 0 + 5 + 0 + 0 + 0 + 0 = 5;

N1020 = 0 + 0 + 0 + 5 + 0 + 0 = 5;

D102 = P102 / N102 = 5/5 = 1 <= 1 - отбрасываем.

Рассмотрим альтернативы 3 и 4 (i = 3, j = 4):

P34 = 4 + 0 + 2 + 5 + 2 + 0 = 13;

N34 = 0 + 5 + 0 + 0 + 0 + 0 = 5;

D34 = P34 / N34 = 13/5 = 2.6 > 1 - принимаем.

P43 = 0 + 5 + 0 + 0 + 0 + 0 = 5;

N43 = 4 + 0 + 2 + 5 + 2 + 0 = 13;

D43 = P43 / N43 = 5/13 = 0.38 <= 1 - отбрасываем.

Рассмотрим альтернативы 3 и 5 (i = 3, j = 5):

P35 = 0 + 0 + 2 + 0 + 0 + 4 = 6;

N35 = 0 + 5 + 0 + 0 + 0 + 0 = 5;

D35 = P35 / N35 = 6/5 = 1.2 > 1 - принимаем.

P53 = 0 + 5 + 0 + 0 + 0 + 0 = 5;

N53 = 0 + 0 + 2 + 0 + 0 + 4 = 6;

D53 = P53 / N53 = 5/6 = 0.83 <= 1 - отбрасываем.

Рассмотрим альтернативы 3 и 6 (i = 3, j = 6):

P36 = 0 + 0 + 2 + 0 + 0 + 4 = 6;

N36 = 0 + 5 + 0 + 0 + 2 + 0 = 7;

D36 = P36 / N36 = 6/7 = 0.86 <= 1 - отбрасываем.

P63 = 0 + 5 + 0 + 0 + 2 + 0 = 7;

N63 = 0 + 0 + 2 + 0 + 0 + 4 = 6;

D63 = P63 / N63 = 7/6 = 1.17 > 1 - принимаем.

Рассмотрим альтернативы 3 и 7 (i = 3, j = 7):

P37 = 4 + 0 + 2 + 5 + 2 + 0 = 13;

N37 = 0 + 5 + 0 + 0 + 0 + 0 = 5;

D37 = P37 / N37 = 13/5 = 2.6 > 1 - принимаем.

P73 = 0 + 5 + 0 + 0 + 0 + 0 = 5;

N73 = 4 + 0 + 2 + 5 + 2 + 0 = 13;

D73 = P73 / N73 = 5/13 = 0.38 <= 1 - отбрасываем.

Рассмотрим альтернативы 3 и 8 (i = 3, j = 8):

P38 = 4 + 0 + 2 + 5 + 0 + 0 = 11;

N38 = 0 + 5 + 0 + 0 + 0 + 0 = 5;

D38 = P38 / N38 = 11/5 = 2.2 > 1 - принимаем.

P83 = 0 + 5 + 0 + 0 + 0 + 0 = 5;

N83 = 4 + 0 + 2 + 5 + 0 + 0 = 11;

D83 = P83 / N83 = 5/11 = 0.45 <= 1 - отбрасываем.

Рассмотрим альтернативы 3 и 9 (i = 3, j = 9):

P39 = 0 + 0 + 2 + 0 + 0 + 4 = 6;

N39 = 0 + 5 + 0 + 0 + 2 + 0 = 7;

D39 = P39 / N39 = 6/7 = 0.86 <= 1 - отбрасываем.

P93 = 0 + 5 + 0 + 0 + 2 + 0 = 7;

N93 = 0 + 0 + 2 + 0 + 0 + 4 = 6;

D93 = P93 / N93 = 7/6 = 1.17 > 1 - принимаем.

Рассмотрим альтернативы 3 и 10 (i = 3, j = 10):

P310 = 0 + 0 + 0 + 5 + 0 + 4 = 9;

N310 = 0 + 5 + 0 + 0 + 2 + 0 = 7;

D310 = P310 / N310 = 9/7 = 1.29 > 1 - принимаем.

P1030 = 0 + 5 + 0 + 0 + 2 + 0 = 7;

N1030 = 0 + 0 + 0 + 5 + 0 + 4 = 9;

D103 = P103 / N103 = 7/9 = 0.78 <= 1 - отбрасываем.

Рассмотрим альтернативы 4 и 5 (i = 4, j = 5):

P45 = 0 + 0 + 0 + 0 + 0 + 4 = 4;

N45 = 4 + 0 + 0 + 5 + 2 + 0 = 11;

D45 = P45 / N45 = 4/11 = 0.36 <= 1 - отбрасываем.

P54 = 4 + 0 + 0 + 5 + 2 + 0 = 11;

N54 = 0 + 0 + 0 + 0 + 0 + 4 = 4;

D54 = P54 / N54 = 11/4 = 2.75 > 1 - принимаем.

Рассмотрим альтернативы 4 и 6 (i = 4, j = 6):

P46 = 0 + 0 + 0 + 0 + 0 + 4 = 4;

N46 = 4 + 0 + 0 + 5 + 2 + 0 = 11;

D46 = P46 / N46 = 4/11 = 0.36 <= 1 - отбрасываем.

P64 = 4 + 0 + 0 + 5 + 2 + 0 = 11;

N64 = 0 + 0 + 0 + 0 + 0 + 4 = 4;

D64 = P64 / N64 = 11/4 = 2.75 > 1 - принимаем.

Рассмотрим альтернативы 4 и 7 (i = 4, j = 7):

P47 = 0 + 5 + 0 + 0 + 0 + 0 = 5;

N47 = 4 + 0 + 0 + 0 + 0 + 0 = 4;

D47 = P47 / N47 = 5/4 = 1.25 > 1 - принимаем.

P74 = 4 + 0 + 0 + 0 + 0 + 0 = 4;

N74 = 0 + 5 + 0 + 0 + 0 + 0 = 5;

D74 = P74 / N74 = 4/5 = 0.8 <= 1 - отбрасываем.

Рассмотрим альтернативы 4 и 8 (i = 4, j = 8):

P48 = 0 + 0 + 2 + 0 + 0 + 0 = 2;

N48 = 0 + 0 + 0 + 0 + 2 + 0 = 2;

D48 = P48 / N48 = 2/2 = 1 <= 1 - отбрасываем.

P84 = 0 + 0 + 0 + 0 + 2 + 0 = 2;

N84 = 0 + 0 + 2 + 0 + 0 + 0 = 2;

D84 = P84 / N84 = 2/2 = 1 <= 1 - отбрасываем.

Рассмотрим альтернативы 4 и 9 (i = 4, j = 9):

P49 = 0 + 0 + 0 + 0 + 0 + 4 = 4;

N49 = 4 + 0 + 0 + 5 + 2 + 0 = 11;

D49 = P49 / N49 = 4/11 = 0.36 <= 1 - отбрасываем.

P94 = 4 + 0 + 0 + 5 + 2 + 0 = 11;

N94 = 0 + 0 + 0 + 0 + 0 + 4 = 4;

D94 = P94 / N94 = 11/4 = 2.75 > 1 - принимаем.

Рассмотрим альтернативы 4 и 10 (i = 4, j = 10):

P410 = 0 + 0 + 0 + 0 + 0 + 4 = 4;

N410 = 4 + 0 + 2 + 0 + 2 + 0 = 8;

D410 = P410 / N410 = 4/8 = 0.5 <= 1 - отбрасываем.

P1040 = 4 + 0 + 2 + 0 + 2 + 0 = 8;

N1040 = 0 + 0 + 0 + 0 + 0 + 4 = 4;

D104 = P104 / N104 = 8/4 = 2 > 1 - принимаем.

Рассмотрим альтернативы 5 и 6 (i = 5, j = 6):

P56 = 0 + 0 + 0 + 0 + 0 + 4 = 4;

N56 = 0 + 0 + 0 + 0 + 2 + 0 = 2;

D56 = P56 / N56 = 4/2 = 2 > 1 - принимаем.

P65 = 0 + 0 + 0 + 0 + 2 + 0 = 2;

N65 = 0 + 0 + 0 + 0 + 0 + 4 = 4;

D65 = P65 / N65 = 2/4 = 0.5 <= 1 - отбрасываем.

Рассмотрим альтернативы 5 и 7 (i = 5, j = 7):

P57 = 4 + 5 + 0 + 5 + 2 + 0 = 16;

N57 = 0 + 0 + 0 + 0 + 0 + 4 = 4;

D57 = P57 / N57 = 16/4 = 4 > 1 - принимаем.

P75 = 0 + 0 + 0 + 0 + 0 + 4 = 4;

N75 = 4 + 5 + 0 + 5 + 2 + 0 = 16;

D75 = P75 / N75 = 4/16 = 0.25 <= 1 - отбрасываем.

Рассмотрим альтернативы 5 и 8 (i = 5, j = 8):

P58 = 4 + 0 + 2 + 5 + 0 + 0 = 11;

N58 = 0 + 0 + 0 + 0 + 0 + 4 = 4;

D58 = P58 / N58 = 11/4 = 2.75 > 1 - принимаем.

P85 = 0 + 0 + 0 + 0 + 0 + 4 = 4;

N85 = 4 + 0 + 2 + 5 + 0 + 0 = 11;

D85 = P85 / N85 = 4/11 = 0.36 <= 1 - отбрасываем.

Рассмотрим альтернативы 5 и 9 (i = 5, j = 9):

P59 = 0 + 0 + 0 + 0 + 0 + 4 = 4;

N59 = 0 + 0 + 0 + 0 + 2 + 0 = 2;

D59 = P59 / N59 = 4/2 = 2 > 1 - принимаем.

P95 = 0 + 0 + 0 + 0 + 2 + 0 = 2;

N95 = 0 + 0 + 0 + 0 + 0 + 4 = 4;

D95 = P95 / N95 = 2/4 = 0.5 <= 1 - отбрасываем.

Рассмотрим альтернативы 5 и 10 (i = 5, j = 10):

P510 = 0 + 0 + 0 + 5 + 0 + 4 = 9;

N510 = 0 + 0 + 2 + 0 + 2 + 0 = 4;

D510 = P510 / N510 = 9/4 = 2.25 > 1 - принимаем.

P1050 = 0 + 0 + 2 + 0 + 2 + 0 = 4;

N1050 = 0 + 0 + 0 + 5 + 0 + 4 = 9;

D105 = P105 / N105 = 4/9 = 0.44 <= 1 - отбрасываем.

Рассмотрим альтернативы 6 и 7 (i = 6, j = 7):

P67 = 4 + 5 + 0 + 5 + 2 + 0 = 16;

N67 = 0 + 0 + 0 + 0 + 0 + 4 = 4;

D67 = P67 / N67 = 16/4 = 4 > 1 - принимаем.

P76 = 0 + 0 + 0 + 0 + 0 + 4 = 4;

N76 = 4 + 5 + 0 + 5 + 2 + 0 = 16;

D76 = P76 / N76 = 4/16 = 0.25 <= 1 - отбрасываем.

Рассмотрим альтернативы 6 и 8 (i = 6, j = 8):

P68 = 4 + 0 + 2 + 5 + 2 + 0 = 13;

N68 = 0 + 0 + 0 + 0 + 0 + 4 = 4;

D68 = P68 / N68 = 13/4 = 3.25 > 1 - принимаем.

P86 = 0 + 0 + 0 + 0 + 0 + 4 = 4;

N86 = 4 + 0 + 2 + 5 + 2 + 0 = 13;

D86 = P86 / N86 = 4/13 = 0.31 <= 1 - отбрасываем.

Рассмотрим альтернативы 6 и 9 (i = 6, j = 9):

P69 = 0 + 0 + 0 + 0 + 0 + 0 = 0;

N69 = 0 + 0 + 0 + 0 + 0 + 0 = 0;

D69 = P69 / N69 = 0/0 = 1 <= 1 - отбрасываем.

P96 = 0 + 0 + 0 + 0 + 0 + 0 = 0;

N96 = 0 + 0 + 0 + 0 + 0 + 0 = 0;

D96 = P96 / N96 = 0/0 = 1 <= 1 - отбрасываем.

Рассмотрим альтернативы 6 и 10 (i = 6, j = 10):

P610 = 0 + 0 + 0 + 5 + 0 + 0 = 5;

N610 = 0 + 0 + 2 + 0 + 0 + 0 = 2;

D610 = P610 / N610 = 5/2 = 2.5 > 1 - принимаем.

P1060 = 0 + 0 + 2 + 0 + 0 + 0 = 2;

N1060 = 0 + 0 + 0 + 5 + 0 + 0 = 5;

D106 = P106 / N106 = 2/5 = 0.4 <= 1 - отбрасываем.

Рассмотрим альтернативы 7 и 8 (i = 7, j = 8):

P78 = 4 + 0 + 2 + 0 + 0 + 0 = 6;

N78 = 0 + 5 + 0 + 0 + 2 + 0 = 7;

D78 = P78 / N78 = 6/7 = 0.86 <= 1 - отбрасываем.

P87 = 0 + 5 + 0 + 0 + 2 + 0 = 7;

N87 = 4 + 0 + 2 + 0 + 0 + 0 = 6;

D87 = P87 / N87 = 7/6 = 1.17 > 1 - принимаем.

Рассмотрим альтернативы 7 и 9 (i = 7, j = 9):

P79 = 0 + 0 + 0 + 0 + 0 + 4 = 4;

N79 = 4 + 5 + 0 + 5 + 2 + 0 = 16;

D79 = P79 / N79 = 4/16 = 0.25 <= 1 - отбрасываем.

P97 = 4 + 5 + 0 + 5 + 2 + 0 = 16;

N97 = 0 + 0 + 0 + 0 + 0 + 4 = 4;

D97 = P97 / N97 = 16/4 = 4 > 1 - принимаем.

Рассмотрим альтернативы 7 и 10 (i = 7, j = 10):

P710 = 0 + 0 + 0 + 0 + 0 + 4 = 4;

N710 = 4 + 5 + 2 + 0 + 2 + 0 = 13;

D710 = P710 / N710 = 4/13 = 0.31 <= 1 - отбрасываем.

P1070 = 4 + 5 + 2 + 0 + 2 + 0 = 13;

N1070 = 0 + 0 + 0 + 0 + 0 + 4 = 4;

D107 = P107 / N107 = 13/4 = 3.25 > 1 - принимаем.

Рассмотрим альтернативы 8 и 9 (i = 8, j = 9):

P89 = 0 + 0 + 0 + 0 + 0 + 4 = 4;

N89 = 4 + 0 + 2 + 5 + 2 + 0 = 13;

D89 = P89 / N89 = 4/13 = 0.31 <= 1 - отбрасываем.

P98 = 4 + 0 + 2 + 5 + 2 + 0 = 13;

N98 = 0 + 0 + 0 + 0 + 0 + 4 = 4;

D98 = P98 / N98 = 13/4 = 3.25 > 1 - принимаем.

Рассмотрим альтернативы 8 и 10 (i = 8, j = 10):

P810 = 0 + 0 + 0 + 0 + 0 + 4 = 4;

N810 = 4 + 0 + 2 + 0 + 2 + 0 = 8;

D810 = P810 / N810 = 4/8 = 0.5 <= 1 - отбрасываем.

P1080 = 4 + 0 + 2 + 0 + 2 + 0 = 8;

N1080 = 0 + 0 + 0 + 0 + 0 + 4 = 4;

D108 = P108 / N108 = 8/4 = 2 > 1 - принимаем.

Рассмотрим альтернативы 9 и 10 (i = 9, j = 10):

P910 = 0 + 0 + 0 + 5 + 0 + 0 = 5;

N910 = 0 + 0 + 2 + 0 + 0 + 0 = 2;

D910 = P910 / N910 = 5/2 = 2.5 > 1 - принимаем.

P1090 = 0 + 0 + 2 + 0 + 0 + 0 = 2;

N1090 = 0 + 0 + 0 + 5 + 0 + 0 = 5;

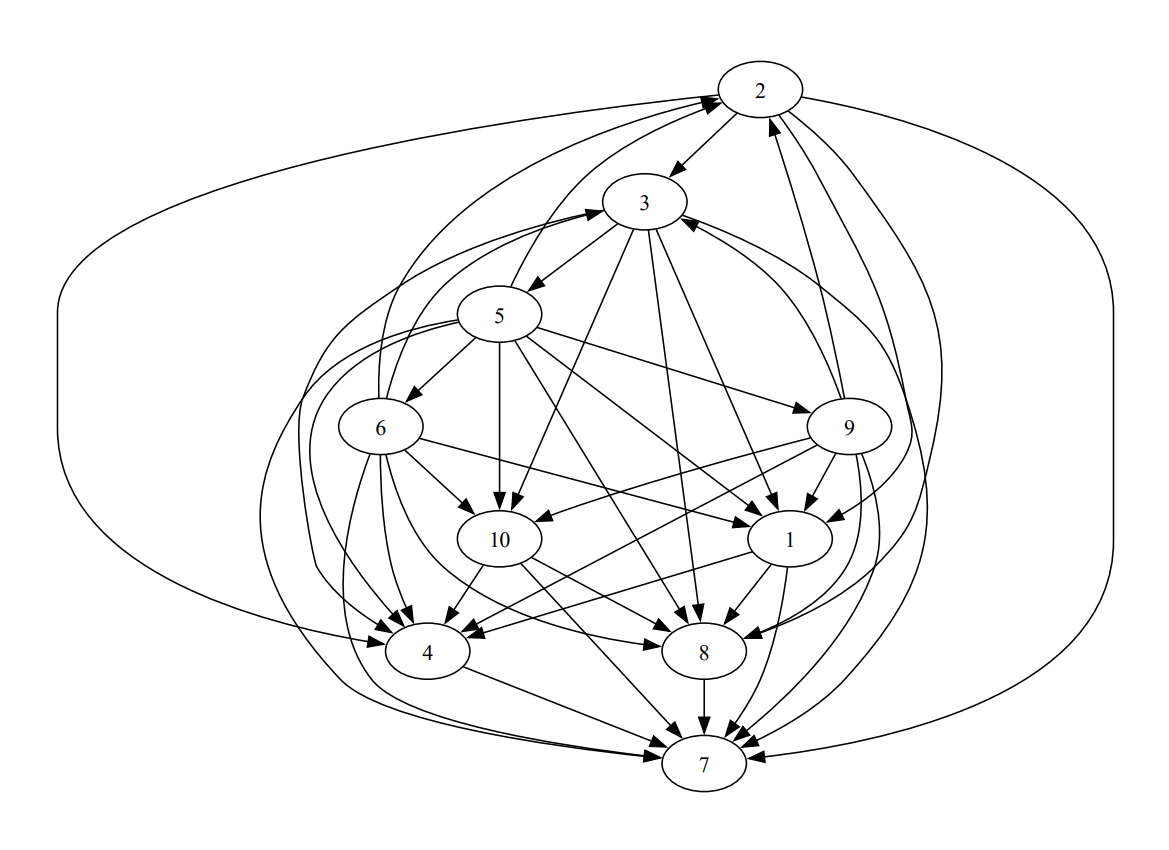
D109 = P109 / N109 = 2/5 = 0.4 <= 1 - отбрасываем.

Составлена матрица предпочтений с внесенными и принятыми значениями D (Таблица 3).

*Таблица 3 – Полная матрица предпочтений альтернатив.*

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| 1 | - | - | - | 2.2 | - | - | 2.2 | 2.2 | - | - |
| 2 | 2.25 | - | 1.75 | 1.44 | - | - | 3.25 | 1.44 | - | - |
| 3 | ∞ | - | - | 2.6 | 1.2 | - | 2.6 | 2.2 | - | 1.29 |
| 4 | - | - | - | - | - | - | 1.25 | - | - | - |
| 5 | 1.25 | 2.25 | - | 2.75 | - | 2 | 4 | 2.75 | 2 | 2.25 |
| 6 | 1.75 | 2.5 | 1.17 | 2.75 | - | - | 4 | 3.25 | - | 2.5 |
| 7 | - | - | - | - | - | - | - | - | - | - |
| 8 | - | - | - | - | - | - | 1.17 | - | - | - |
| 9 | 1.75 | 2.5 | 1.17 | 2.75 | - | - | 4 | 3.25 | - | 2.5 |
| 10 | - | - | - | 2 | - | - | 3.25 | 2 | - | - |

По матрице построен граф предпочтений (Рисунок 1).



**Рисунок 1 – Вид графа предпочтений**

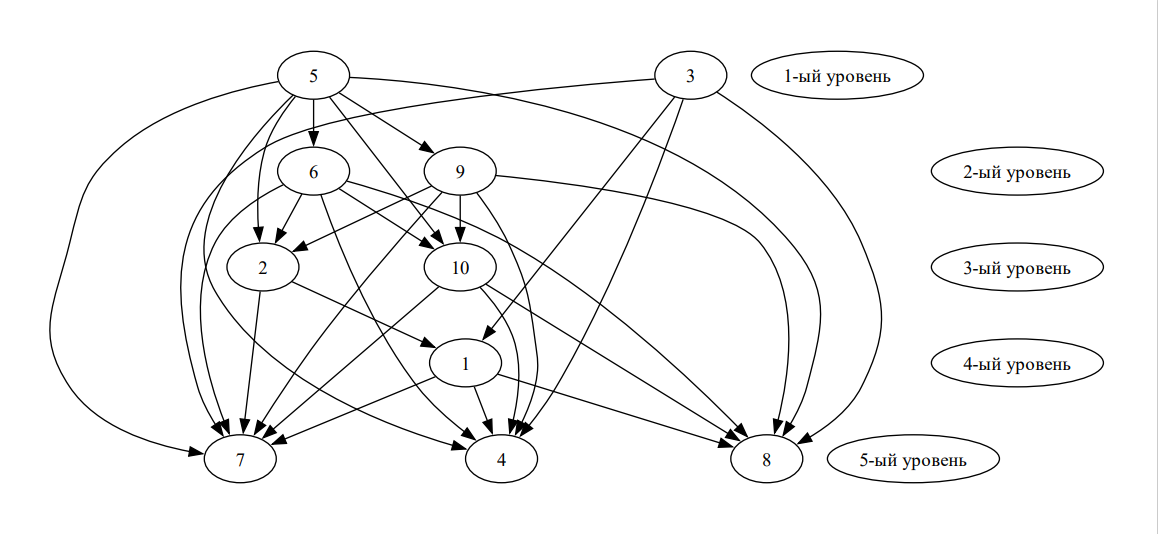
Назначен порог отбора предпочтений C = 1.76 (это соответствует тому, что учитываются только более сильные связи в графе).

Таким образом, матрица разрежается. В ней остаются только самые сильные связи (Таблица 4).

*Таблица 4* **–** *Матрица предпочтений проектов, при пороге С=1.76*

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| 1 | - | - | - | 2.2 | - | - | 2.2 | 2.2 | - | - |
| 2 | 2.25 | - | - | - | - | - | 3.25 | - | - | - |
| 3 | ∞ | - | - | 2.6 | - | - | 2.6 | 2.2 | - | - |
| 4 | - | - | - | - | - | - | - | - | - | - |
| 5 | - | 2.25 | - | 2.75 | - | 2 | 4 | 2.75 | 2 | 2.25 |
| 6 | - | 2.5 | - | 2.75 | - | - | 4 | 3.25 | - | 2.5 |
| 7 | - | - | - | - | - | - | - | - | - | - |
| 8 | - | - | - | - | - | - | - | - | - | - |
| 9 | - | 2.5 | - | 2.75 | - | - | 4 | 3.25 | - | 2.5 |
| 10 | - | - | - | 2 | - | - | 3.25 | 2 | - | - |

По этой матрице построен граф предпочтений (Рисунок 2).



**Рисунок 2 – Вид графа предпочтений для случая порога принятия решений C = 1.76**

Циклов в графе нет, при этом граф остался целостным. Оптимальным решением является альтернатива А5 и A3.

## **1.3 Вывод**

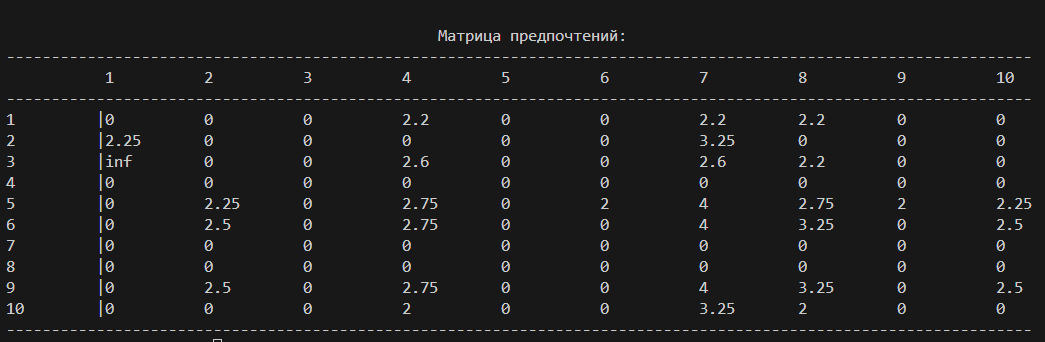
Метод Электра II позволяет определить оптимальное решение, уменьшив субъективный фактор, который был у метода Парето и у методов сужения, однако если ставить порог равным 1, то в графе могут появляться циклы, из-за которых невозможно определить оптимальное решение. Поэтому нужно экспериментально определять подходящее значение порога.

## **1.4 Результат работы программы**

**Изображение выглядит как снимок экрана, текст

Автоматически созданное описание**

**Рисунок 3 – Вывод матрицы предпочтений**

****

**Рисунок 4 – Вывод матрицы предпочтений с порогом = 1.76**

**ЗАКЛЮЧЕНИЕ**

В ходе данной практической работы мной был изучен метод Электра II из семейства Электра и применён для нахождения оптимального высшего технического заведения. Преимуществами метода является большая объективность по сравнению с методом Парето и его методами сужения, однако, чтобы получить единственное решение, необходимо дополнительно устанавливать порог стремления, чтобы на графе предпочтений не образовывалось петель и чтобы он оставался целостным.

**СПИСОК ИНФОРМАЦИОННЫХ ИСТОЧНИКОВ**

1. Болотова Л. С. Многокритериальная оптимизация. Болотова Л. С., Сорокин А. Б. [Электронный ресурс] / Метод. указания по вып. курсовой работы — М.: МИРЭА, 2015.
2. Сорокин А. Б. Методы оптимизации: гибридные генетические алгоритмы. Сорокин А. Б. [Электронный ресурс] / Учебно-метод. пособие — М.: МИРЭА, 2016.
3. Сорокин А. Б. Линейное программирование: практикум. Сорокин А. Б., Бражникова Е. В., Платонова О. В. [Электронный ресурс] / Учебно-метод. пособие — М.: МИРЭА, 2017.

**ПРИЛОЖЕНИЯ**

Приложение А – Код реализации метода Электра II на языке Python.

**Приложение А**

Код реализации метода Электра II на языке Python.

Листинг А.1. Реализация метода Электра II.

import csv

import math

from graphviz import Digraph

def print\_matrix(c = 1):

'''Функция для вывода матрицы предпочтений с порогом'''

print('-' \* (10 \* (len(matrix)+1) + 4))

print(11 \* ' ', end='')

for i in range(1, len(matrix) + 1):

print(f'{i}'.ljust(11), end='')

print('\n' + '-' \* (10 \* (len(matrix)+1) + 4))

for i in range(len(matrix)):

print(str(i+1).ljust(10), end='|')

for j in range(len(matrix)):

if matrix[i][j] < c:

matrix[i][j] = 0

print(str(matrix[i][j]).ljust(10), end=' ')

print()

print('-' \* (10 \* (len(matrix)+1) + 4))

def compare\_alternatives(i, j, alt\_i, alt\_j, criteria):

'''Функция для сравнения альтернатив по кодам'''

P, N = 0, 0

P\_STR, N\_STR = f'P{i}{j} =', f'N{i}{j} ='

for i in range(len(criteria)):

counter\_i, counter\_j = 0, 0

code = criteria[i]['Код'].split(';')

for border in criteria[i]['Шкала'].split(';')[1:]:

if alt\_i[i] < float(border):

counter\_i += 1

if alt\_j[i] < float(border):

counter\_j += 1

alt\_i[i] = float(code[counter\_i])

alt\_j[i] = float(code[counter\_j])

if criteria[i]['Стремление'] == '-':

alt\_i[i] \*= (-1)

alt\_j[i] \*= (-1)

if alt\_i[i] > alt\_j[i]:

P += int(criteria[i]['Вес критерия'])

P\_STR += (' ' + criteria[i]['Вес критерия'] + ' +')

N\_STR += (' ' + str(0) + ' +')

elif alt\_i[i] < alt\_j[i]:

N += int(criteria[i]['Вес критерия'])

N\_STR += (' ' + criteria[i]['Вес критерия'] + ' +')

P\_STR += (' ' + str(0) + ' +')

else:

N\_STR += (' ' + str(0) + ' +')

P\_STR += (' ' + str(0) + ' +')

return P\_STR.rstrip(' +') + f' = {P}', N\_STR.rstrip(' +') + f' = {N}', P, N

def generate\_matrix():

'''Функция для генерации матрицы предпочтений'''

*Продолжение Листинга А.1*

def get\_D(P, N):

'''Функция для расчёта D-стремления'''

if N == 0 and P == 0:

return 1

elif N == 0 and P != 0:

return math.inf

value = P/N

if math.floor(value) == math.ceil(value):

value = int(value)

else:

value = round(value, 2)

return value

def generate\_D\_STR(i, j, P, N):

'''Функция для генерации D-стремления'''

value = get\_D(P, N)

if value <= 1:

return f'D{i}{j} = P{i}{j} / N{i}{j} = {P}/{N} = {value} <= 1 - отбрасываем.'

else:

if value == math.inf:

value = '\u221e'

return f'D{i}{j} = P{i}{j} / N{i}{j} = {P}/{N} = {value} > 1 - принимаем.'

for i in range(1, len(data)+1):

for j in range(i+1, len(data)+1):

print(f'Рассмотрим альтернативы {i} и {j} (i = {i}, j = {j}):')

alt\_i, alt\_j = data[i-1].copy(), data[j-1].copy()

P\_STR, N\_STR, P, N = compare\_alternatives(

i, j, alt\_i, alt\_j, criteria)

print(P\_STR+';', N\_STR+';', sep='\n')

print(generate\_D\_STR(i, j, P, N))

D = get\_D(P, N)

if D > 1:

matrix[i-1][j-1] = D

print(f'P{j}{i}{N\_STR[3:]};', f'N{j}{i}{P\_STR[3:]};', sep='\n')

print(generate\_D\_STR(j, i, N, P))

D = get\_D(N, P)

if D > 1:

matrix[j-1][i-1] = D

def draw\_graph(c=1):

'''Функция для рисования хаотичного графа с порогом'''

dot = Digraph(f'Хаотичный Граф с порогом = {c}')

for i in range(len(matrix)):

dot.node(str(i+1))

for i in range(len(matrix)):

for j in range(len(matrix)):

if matrix[i][j] >= c:

dot.edge(str(i+1), str(j+1))

dot.render(view=True)

def smart\_draw\_graph(levels, c=1):

'''Функция для рисования графа по уровням с порогом'''

dot1 = Digraph(f"Граф с порогом = {c}")

for i in range(len(matrix)):

for j in range(len(matrix)):

if matrix[i][j] >= c:

*Продолжение Листинга А.1*

dot1.edge(str(i+1), str(j+1))

for i in range(len(levels)):

sub = Digraph(name='Подграф'+str(i))

sub.attr(rank='same')

sub.node(f'{i+1}-ый уровень')

for j in levels[i]:

sub.node(f'{j+1}')

dot1.subgraph(sub)

dot1.render(view=True)

def get\_levels(c=1):

'''Вспомогательная функция для определения уровня вершин'''

levels = [] # массив всех вершин

visited = [] # массив посещённых вершин

while len(visited) < len(matrix):

level = []

for i in range(len(matrix)):

if i in visited:

continue

flag = True

for j in range(len(matrix)):

if matrix[j][i] >= c:

flag = any(j in lev for lev in levels)

if not flag:

break

if flag:

level.append(i)

visited.append(i)

levels.append(level)

print(f'{len(levels)}-ый уровень: ' +

', '.join(map(lambda x: str(x+1), level)))

return levels

with open('TPR\_PRACT2\_LIST.csv', encoding='utf-8') as file, \

open('codes.csv', encoding='utf-8') as criteria\_file:

criteria = [i for i in csv.DictReader(

criteria\_file)] # Информация о критериях

data = list(map(lambda x: [float(i) for i in x], [

i[1:] for i in csv.reader(file)][1:])) # Значения критериев для рассматриваемых альтернатив

matrix = [[0]\*len(data) for \_ in range(len(data))] # Матрица предпочтений

generate\_matrix()

print("Матрица предпочтений:".center(201))

print\_matrix()

draw\_graph()

arg = 1.76

smart\_draw\_graph(get\_levels(c=arg), c=arg)