**RESPONSI PRAKTIKUM SCPK**

NAMA : Tsaniah Munfidah

NIM : 12319095

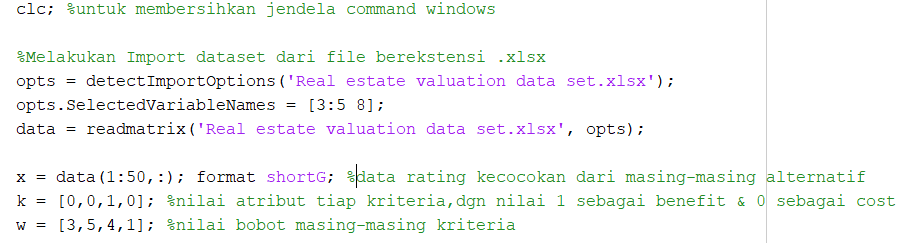
PLUG : D

(Jum’at, 25 Juni 2021 – Sabtu, 26 Juni 2021)

1. **Weighted Product (WP)**
2. Link Github WP : <https://github.com/tsaniahm/D_123190095_ResponsiSCPK.git>
3. Langkah pembuatan program dan penjelasan (yang penting-penting saja)

Contoh:

1. Import data dan inisialisasi nilai atribut dan bobot



1. Tahap berbaikan bobot dan menghitung hasil vektor

|  |
| --- |
|  |

1. Tahap perankingan dan pengurutan berdasarkan nilai vektor tertinggi

|  |
| --- |
|  |

1. Tahap menampilkan hasil urutan nilai vektor 5 terbesar

|  |
| --- |
|  |

1. Pembuktian (dibuktikan dengan perhitungan manual, 5 ranking teratas)

* Ada 5 Data ranking teratas, yaitu:

A1 = Dataset baris ke 20

A2 = Dataset baris ke 12

A3 = Dataset baris ke 17

A4 = Dataset baris ke 1

A5 = Dataset baris ke 27

* Pada kasus ini terdapat 5 kriteria yang dijadikan sebagai acuan dalam pengambilan keputusan yaitu:

C1 = house age

C2 = distance to the nearest MRTC3 station

C3= number of convenience store

C4= house price of unit area

* Selain itu, Tingkat kepentingan (W/bobot) setiap kriteria menggunakan nilai dengan range 1 sampai dengan 5 sebagai berikut:

1 = Sangat rendah,

2 = Rendah,

3 = Cukup,

4 = Tinggi,

5 = Sangat Tinggi.

W = [3,5,4,1]

K = [0,0,0,1,0] kriteria tiap bobot

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Alternatif | Kriteria | | | |
| C1 | C2 | C3 | C4 |
| A1 | 1,5 | 23,38284 | 7 | 47,7 |
| A2 | 6,3 | 90,45606 | 9 | 58,1 |
| A3 | 1 | 292,9978 | 6 | 70,1 |
| A4 | 32 | 84,87882 | 10 | 37,9 |
| A5 | 31 | 383,8624 | 5 | 56,2 |

W1 = 3/(3+5+4+1) = 3/13 = 0,23 = **-0.23**

W2 = 5/(3+5+4+1) = 5/13 = 0,38 = **-0,38**

W3 = 4/(3+5+4+1) = 4/13 = 0,31= **0.31**

W4 = 1/(3+5+4+1) = 1/13 = 0,08= **0.08**

W1 + W2 + W3 + W4 + W5 = 1

S1 = (1,5^(-0,23))\*(23.38284^(-0,38))\*(7^0,31)\*(47.7^(-0,08)) = **0.37**

S2 = (6.3^(-0,23))\*(90.45606^(-0,38))\*(9^0,31)\*(58.1^(-0,08)) = **0.17**

S3 = (1^(-0,23))\*(292.9978^(-0,38))\*(6^0,31)\*(70.1^(-0,08)) = **0.14**

S4 = (32^(-0,23))\*(84.87882^(-0,38))\*(10^0,31)\*(37.9^(-0,08)) = **0.13**

S5 = (31^(-0,23))\*(383.8624^(-0,38))\*(5^0,31)\*(56.2^(-0,08)) = **0.09**

**St = S1 + S2 + S3+S4+S5+….+S50 = 3.13** (pada matlab digunakan 50 data sehingga harus memakai S total dari 50 data, karena beda data hasil S totalnya juga akan beda dan akan mempengaruhi hasil akhir)

V1 = S1 / St

**V1 = 0,37/ 3,13 = 0,11701 (TERBUKTI)**

V2 = S2 / St

**V2 = 0,17/ 3,13 = 0, 05314 (TERBUKTI)**

V3 = S3 / St

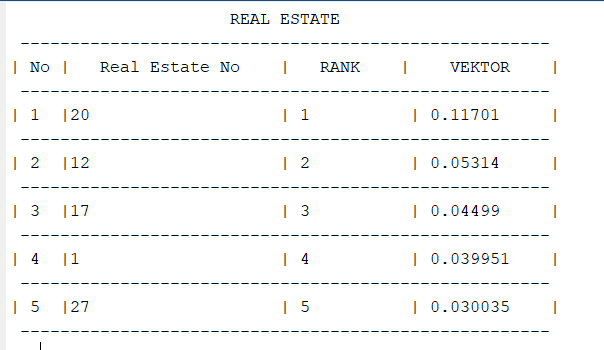
**V3 = 0,14/ 3,13 = 0,04499 (TERBUKTI)**

V4 = S4 / St

**V4 = 0,13/ 3,13 = 0,0399 (TERBUKTI)**

V5 = S5 / St

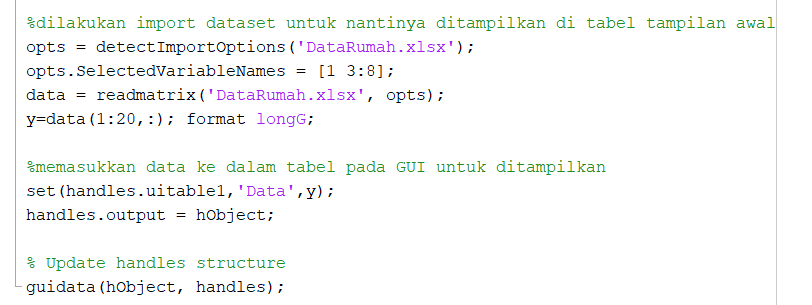
**V5 = 0,9/ 3,13 = 0,0300 (TERBUKTI)**



1. **Simple Additive Wighting (SAW)**
2. Link Github SAW :
3. Langkah pembuatan program dan penjelasan (yang penting-penting saja)

Contoh:

1. Memunculkan dataset ke dalam tabel GUI dengan menuliskan kode pada OpeningFcn sehingga saat GUI di jalankan isi tabel dapat langsung muncul juga di dalam tampilan



1. Melakukan penulisan kode di callback pushbutton1 yang dimana digunakan untuk melakukan proses kalkulasi data mulai dari di import, diolah, hingga ditampilkan pada tabel hasil

|  |
| --- |
|  |

1. Screenshot GUI



1. Pembuktian (dibuktikan dengan perhitungan manual, 5 ranking rumah teratas)

Ada 5 Data ranking teratas, yaitu:

A1 = Dataset baris ke 14

A2 = Dataset baris ke 5

A3 = Dataset baris ke 20

A4 = Dataset baris ke 3

A5 = Dataset baris ke 12

Pada kasus ini terdapat 6 kriteria yang dijadikan sebagai acuan dalam pengambilan keputusan yaitu:

C1 = Harga (BOBOT : 0.3)

C2 = Luas Tanah (BOBOT : 0.2 )

C3 = Luas Bangunan (BOBOT : 0.23 )

C4 = Jumlah Kamar Tidur (BOBOT : 0.1 )

C5 = Jumlah Kamar Mandi (BOBOT : 0.7)

C6 = Jumlah Garasi (BOBOT : 0.1)

k=[0,1,1,1,1,1] Kriteria tiap bobot

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Alternatif | Kriteria | | | | | |
| C1 | C2 | C3 | C4 | C5 | C6 |
| A1 | 12,5 M | 200 | 979 | 4 | 2 | 6 |
| A2 | 9M | 400 | 355 | 6 | 5 | 3 |
| A3 | 9,7M | 450 | 248 | 5 | 5 | 4 |
| A4 | 3M | 267 | 250 | 4 | 4 | 4 |
| A5 | 9,5M | 450 | 240 | 4 | 4 | 1 |

**Normalisasi Kriteria C1:**

C1 = cost, maka dari 20 data yg diolah diperoleh nilai minimal dari C1 yaitu 430000000

A1 = 430000000/12500000000 = **0.0344**

A2 = 430000000/9000000000 = **0.047777778**

A3 = 430000000/9700000000 = **0.044329897**

A4 = 430000000/3000000000 = **0.143333333**

A5 = 430000000/9500000000 = **0.045263158**

**Normalisasi Kriteria C2**

C2 = benefit, maka dari 20 data yang diolah diperoleh nilai maximum dari C2 yaitu

450

A1 = 200/450 = **0.444444444**

A2 = 400/450 = **0.888888889**

A3 = 450/450 = **1**

A4 = 267/450 = **0.593333333**

A5 = 450/450 = **1**

**Normalisasi Kriteria C3**

C3 = benefit, maka dari 20 data yang diolah diperoleh nilai maximum dari C3 yaitu 979

A1 = 979/979 = **1**

A2 = 355/979 = **0.362614913**

A3 = 248/979 = **0.253319714**

A4 = 250/979 = **0.255362615**

A5 = 240/979 = **0.24514811**

**Normalisasi Kriteria C4**

C4 = benefit, maka dari 20 data yang diolah diperoleh nilai maximum dari C4 yaitu 6

A1 =4 /6 = **0.66666667**

A2 = 6/6 = **1**

A3 = 5/6 = **0.83333333**

A4 = 4/6 = **0.66666667**

A5 = 4/6 = **0.66666667**

**Normalisasi Kriteria C5**

C5 = benefit, maka dari 20 data yang diolah diperoleh nilai maximum dari C5 yaitu 5

A1 =2/5 = **0.4**

A2 = 5/5 = **1**

A3 = 5/5 = **1**

A4 = 4/5 = **0.8**

A5 = 4/5 = **0.8**

**Normalisasi Kriteria C6**

C6 = benefit, maka dari 20 data yang diolah diperoleh nilai maximum dari C6 yaitu 5

A1 =6/6 = **1**

A2 = 3/6 = **0.5**

A3 = 4/6 = **0.66666667**

A4 = 4/6 = **0.66666667**

A5 = 1/6 = **0.16666667**

W = [0.3, 0.2, 0.23, 0.10, 0.07, 0.10] (bobot)

* **V1** = ( 0.0344)(0.3) + ( 0.444444444)(0.2 )+ (1)(0.23) + (0.66666667)(0.10) + (0.4)(0.07)

+ (1)(0.10)

= **0.5239 (TERBUKTI)**

* **V2** = ( 0.047777778)(0.3) + (0.888888889)(0.2) + (0.362614913)(0.23) + (1)(0.10) + (1)(0.07)

+ (0.5)(0.10)

=  **0.4955 (TERBUKTI)**

* **V3** = ( 0.044329897)(0.3) + (1)(0.2) + (0.253319714)(0.23) + (0.83333333)(0.10) + (1)(0.07)

+ (0.66666667)(0.10)

=  **0.4916 (TERBUKTI)**

* **V4** = (0.143333333)(0.3) + (0.593333333)(0.2) + (0.255362615)(0.23) + (0.66666667)(0.10)

+ (0.8)(0.07) + ( 0.66666667)(0.10)

= **0.4097 (TERBUKTI)**

* **V5**= (0.045263158)(0.3) + 1()(0.2) + ( 0.24514811)(0.23) + (0.66666667)(0.10) + (0.8)(0.07)

+ (0.16666667)(0.10)

= **0.4093 (TERBUKTI)**