



Pemrograman Komputer

oleh

Djoko Luknanto

Departemen Teknik Sipil dan Lingkungan
Fakultas Teknik Universitas Gadjah Mada

Evaluasi Polinomial

Bentuk umum

$$p_n(x) = \sum_{i=0}^n a_i x^i = a_0 + a_1 x + a_2 x^2 + \dots + a_n x^n$$

Contoh:

$$p_n(2) = \sum_{i=0}^n a_i 2^i = a_0 + a_1 2 + a_2 2^2 + \dots + a_n 2^n$$

$$p_2(x) = 6 + 5x + x^2$$

$$p_2(2) = 6 + 5(2) + (2)^2 = 6 + 10 + 4 = 20$$

Secara numeris evaluasi di atas tidak efisien!

Evaluasi Polinomial Efisien

Bentuk umum

$$p_n(x) = \sum_{i=0}^n a_i x^i = a_0 + a_1 x + a_2 x^2 + \dots + a_n x^n$$

diubah menjadi (cara Horner):

$$p_n(x) = a_0 + x(a_1 + x(a_2 + \dots + x(a_{n-1} + a_n x) \dots))$$

terdapat pola berulang $a_n x + a_{n-1}$

bentuk akhir menjadi:

$$p_n(x) = (((\dots(a_n x + a_{n-1})x + a_{n-2})x + \dots + a_1)x + a_0$$

Evaluasi Polinomial Efisien

Contoh $n = 4$

$$p_4(x) = (((a_4x + a_3)x + a_2)x + a_1)x + a_0$$

Dalam pemrograman bentuk umum

$$p_n(x) = (((... (a_nx + a_{n-1})x + a_{n-2})x + ... + a_1)x + a_0$$

$$p_n = (((a_nx + a_{n-1})x + a_{n-2})x + \dots)x + a_0$$

algoritmanya menjadi:

1. $P = A(n)$ Algoritmanya sangat sederhana dan langsung
2. For $i = n-1$ to 0
 $P = P * X + A(i)$
Next i

Evaluasi Polinomial Cara Biasa

Indeks polinomial disesuaikan

$$p_n(x) = \sum_{i=0}^n a_i x^i = \sum_{j=1}^m a_j x^{j-1} = a_1 + a_2 x + a_3 x^2 + \dots + a_m x^{m-1}$$

Function Polinomial (X As Single, A As Range) As Double

' Evaluasi polinomial dengan cara biasa

Dim j As Integer, Hasil As Double

```
Hasil = A(1, 1)
```

```
For j = 2 To A.Count
```

```
    Hasil = Hasil + A(1, j) * X ^ (j - 1)
```

```
Next j
```

```
Polinomial = Hasil
```

perkalian

perpangkatan

penambahan

```
End Function
```

Evaluasi Polinomial Cara Horner

Bentuk akhir cara Horner

$$p_m(x) = (((... (a_m x + a_{m-1}) x + a_{m-2}) x + \dots + a_2) x + a_1$$

```
Function PoliHorner(X As Single, A As Range) As Double
```

```
' Evaluasi polinomial dengan cara
```

Indeks dalam Range
artinya A(baris, kolom)

```
Dim i As Integer, Hasil As Double
```

```
Hasil = A(1, A.Count)
```

```
For i = A.Count - 1 To 1 Step -1
```

```
    Hasil = Hasil * X + A(1, i)
```

```
Next i
```

```
PoliHorner = Hasil
```

tanpa perpangkatan

penambahan

perkalian

```
End Function
```

VBA-Excel: Cara Evaluasi Polinomial

X sebagai input pertama

Koefisien polinomial Range (\$B\$3:\$K\$3) sebagai input kedua

Menggunakan Macro

Persamaan Polinomial

$x^i, i =$	0	1	2	3	4	5	6	7	8	9
$a_i =$	6,57	2,35	-7,8	4,56	-7,8	1,09	0	0,5	0,78	-1,2

Pers: $6,57 + 2,35X - 7,8X^2 + 4,56X^3 - 7,8X^4 + 1,09X^5 - 0X^6 + 0,5X^7 + 0,78X^8 - 1,2X^9$

Macro: $f_x = =\$B\$3+\$C\$3*A8+\$D\$3*A8*\$D\$2+\$E\$3*A8*\$E\$2+\$F\$3*A8*\$F\$2+\$G\$3*A8*\$G\$2+\$H\$3*A8*\$H\$2+\$I\$3*A8*\$I\$2+\$J\$3*A8*\$J\$2+\$K\$3*A8*\$K\2

X	Macro	Polinomial	Horner
0,500	5,9161718750000	5,9161718750000	5,9161718750000
1,600	-66,1896959744000	-66,1897035394164	-66,1897035394164
2,700	-6.637,0623509826000	-6.637,0634421613300	-6.637,0634421613300
-2,300	2.217,9569678974000	2.217,9565186524200	2.217,9565186524200
1,760	-141,5071308248280	-141,5071244885550	-141,5071244885550

VBA: Function Polinomial(X As Single, Grup As Range) As Double

```

Function Polinomial(X As Single, Grup As Range) As Double
    ' Evaluasi polinomial dengan cara biasa
    Dim j As Integer, Hasil As Double
    Hasil = Grup(1, 1)
    For j = 2 To Grup.Count
        Hasil = Hasil + Grup(1, j) * X ^ (j - 1)
    Next j
    Polinomial = Hasil
End Function
    
```

VBA: Function PoliHorner(X As Single, Data As Range) As Double

```

Function PoliHorner(X As Single, Data As Range) As Double
    ' Evaluasi polinomial dengan cara Horner
    Dim i As Integer, Hasil As Double
    Hasil = Data(1, Data.Count)
    For i = Data.Count - 1 To 1 Step -1
        Hasil = Hasil * X + Data(1, i)
    Next i
    PoliHorner = Hasil
End Function
    
```

Definisi fungsi dalam VBA dan cara memanggilya dalam worksheet yang sederhana, membuat pembelajaran Pemrograman Komputer menjadi lebih mudah.

Cara Biasa

Cara Horner

Hasil hitungan Cara Biasa dan Cara Horner identik, namun dari sisi efisiensi Cara Horner lebih bagus!

Syntax: For...Next Loop

```
For counter = start To end [ Step step ]  
[ statements ]  
[ Exit For ]  
[ statements ]  
Next [ counter ]
```

Perhatikan konvensi penulisan dalam manual Bahasa Pemrograman VBA

Penjelasan dalam [...] boleh tidak ada!

The **For...Next** statement syntax has these parts:

Part	Description
<i>counter</i>	Required. Numeric <u>variable</u> used as a loop counter. The variable can't be a <u>Boolean</u> or an <u>array</u> element.
<i>start</i>	Required. Initial value of <i>counter</i> .
<i>end</i>	Required. Final value of <i>counter</i> .
<i>step</i>	Optional. Amount <i>counter</i> is changed each time through the loop. If not specified, <i>step</i> defaults to one.
<i>statements</i>	Optional. One or more statements between For and Next that are executed the specified number of times.

oleh
Djoko Luknanto
Departemen Teknik Sipil dan Lingkungan
Fakultas Teknik Universitas Gadjah Mada



MENGGUNAKAN VBA- EXCEL INDEX UNTUK AKSES VARIABEL RANGE

Index pada variable Range A(\$C\$4:\$L\$13)

Index A(1,1) selalu mengacu kepada sel pertama dari Range, dalam hal ini \$C\$4

Jumlah baris (i) harus dihitung sebelumnya

Jumlah kolom (j) harus dihitung sebelumnya

		Data berupa Range, A(\$C\$4:\$L\$13)									
		Kolom (j)									
		1	2	3	4	5	6	7	8	9	10
Baris (i)	1	-4.930,636	8.252,899	7.403,666	-1.775,592	-5.737,632	-7.039,365	-1.131,156	8.511,449	-871,008	-986,999
	2	-1.014,508	-6.711,102	2.628,196	-2.032,442	-372,007	3.034,318	-7.735,076	9.117,633	1.424,572	5.713,734
	3	2.963,003	3.586,102	-7.160,540	-3.969,212	-6.748,008	2.609,882	5.555,378	7.180,102	7.606,991	-6.429,423
	4	5.813,748	-5.681,102	8.122,424	9.511,659	-8.377,821	-925,260	3.391,837	7.495,122	-6.002,378	-6.551,144
	5	-1.111,102	7.111,102	-1.111,102	7.111,102	-1.111,102	7.111,102	-1.111,102	7.111,102	-1.111,102	7.111,102
	6	-1.111,102	7.111,102	-1.111,102	7.111,102	-1.111,102	7.111,102	-1.111,102	7.111,102	-1.111,102	7.111,102
	7	-9.116,826	-2.336,102	1.233,472	-8.188,886	-5.287,350	4.346,124	-2.361,058	2.258,152	414,782	-2.961,683
	8	-7.797,252	-8.173,102	4.177,985	8.135,578	5.866,760	7.829,143	-2.536,848	1.650,543	-9.543,474	-4.052,065
	9	-2.939,039	-8.173,102	3.333,631	3.066,252	-530,397	-2.917,819	4.794,815	9.632,193	9.379,944	4.477,936
	10	6.393,997	-7.326,102	3.563,999	-1.118,547	-9.775,967	9.657,431	5.874,609	7.764,309	-993,313	9.413,132

Dalam menggunakan index untuk Range, perlu diperhatikan: (1) Sel pertama yang diacu, (2) jumlah baris dan kolom. Di luar Kawasan Range masih dapat diacu, namun nilainya berisi “sampah” yang mungkin tidak mempunyai arti.

Index pada variable Range Nilai(\$C\$4:\$L\$13)

Jumlah kolom (j) harus dihitung sebelumnya

Index Nilai (1,1) selalu mengacu kepada sel pertama dari Range, dalam hal ini \$C\$4

\$C\$4

Data berupa... A(\$C\$4:\$L\$13)

	1	2	3	4	5	6	7	8	9	10
1	-3.822,208	328,848	-2.452,478	4.266,679	3.236,408	8.887,459	-6.723,908	4.307,660	9.763,267	570,483
2	-4.093,162	-7.728,689	6.298,407	-2.802,802	-9.323,029	-5.538,953	-5.651,601	6.295,332	5.113,378	-3.019,816
3	-4.486,034	-9.459,020	-4.529,333	-2.202,252	-6.402,393	4.406,976	-8.280,821	6.670,928	311,425	5.132,787
4	3.403,259	-2.847,692	-6.838,810	-3.333,350	-349,489	4.039,855	8.929,821	9.965,648	1.639,676	-8.814,443
5	-1.103,859	-5.584,965	7.318,169	9.159,516	8.044,694	8.636,164	-6.700,637	8.268,158	3.082,112	-9.871,389
6	-3.566,595	9.775,949	-882,476	8.548,723	-8.518,554	50,769	-759,339	9.358,120	-1.024,175	-9.615,852
7	3.150,973	-6.996,558	-5.223,917	-3.613,761	-7.081,596	7.785,155	-6.190,563	-4.193,422	-2.237,632	7.949,026
8	1.305,317	4.936,138	-7.762,721	361,273	8.931,773	7.955,668	7.839,338	8.026,733	-7.629,425	399,971
9	8.509,989	-2.768,928	-3.665,709	-1.845,586	1.342,593	7.662,258	-7.512,193	1.140,751	-2.193,382	-908,151

Akses dengan index: Nilai (i; j; \$C\$4:\$L\$13)

	1	2	3	4	5	6	7	8	9	10
1	\$C\$4	328,848	-2.452,478	4.266,679	3.236,408	8.887,459	-6.723,908	4.307,660	9.763,267	570,483
2	-4.093,162	-7.728,689	6.298,407	-2.802,802	-9.323,029	-5.538,953	-5.651,601	6.295,332	5.113,378	-3.019,816
3	-4.486,034	-9.459,020	-4.529,333	-2.202,252	-6.402,393	4.406,976	-8.280,821	6.670,928	311,425	5.132,787
4	3.403,259	-2.847,692	-6.838,810	-3.333,350	-349,489	4.039,855	8.929,821	9.965,648	1.639,676	-8.814,443
5	-1.103,859	-5.584,965	7.318,169	9.159,516	8.044,694	8.636,164	-6.700,637	8.268,158	3.082,112	-9.871,389
6	549,320	-3.857,172	3.863,389	-7.769,149	7.586,499	-3.164,527	4.758,804	-9.998,239	-7.983,428	1.174,395
7	-3.566,595	9.775,949	-882,476	8.548,723	-8.518,554	50,769	-759,339	9.358,120	-1.024,175	-9.615,852
8	3.150,973	-6.996,558	-5.223,917	-3.613,761	-7.081,596	7.785,155	-6.190,563	-4.193,422	-2.237,632	7.949,026
9	1.305,317	4.936,138	-7.762,721	361,273	8.931,773	7.955,668	7.839,338	8.026,733	-7.629,425	399,971
10	8.509,989	-2.768,928	-3.665,709	-1.845,586	1.342,593	7.662,258	-7.512,193	1.140,751	-2.193,382	-908,151

Akses dengan index: Nilai (i; j; \$E\$4:\$E\$13)

	1	2	3	4	5	6	7	8	9	10
1	8.636,164	-6.700,637	8.268,158	3.082,112	-9.871,389	0,000	0,000	0,000	0,000	0,000
2	-3.164,527	4.758,804	-9.998,239	-7.983,428	1.174,395	0,000	0,000	0,000	0,000	0,000
3	50,769	-759,339	9.358,120	-1.024,175	-9.615,852	0,000	0,000	0,000	0,000	0,000
4	7.785,155	-6.190,563	-4.193,422	-2.237,632	7.949,026	0,000	0,000	0,000	0,000	0,000
5	7.955,668	7.839,338	8.026,733	-7.629,425	399,971	0,000	0,000	0,000	0,000	0,000
6	7.662,258	-7.512,193	1.140,751	-2.193,382	-908,151	0,000	0,000	0,000	0,000	0,000
7	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000
8	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000
9	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000
10	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000

Jumlah baris (i) harus dihitung sebelumnya

Dalam menggunakan index untuk Range, perlu diperhatikan: (1) Sel pertama yang diacu, (2) jumlah baris dan kolom. Di luar Kawasan Range masih dapat diacu, namun nilainya berisi "sampah" yang mungkin tidak mempunyai arti.

Index pada variable Range Nilai(\$H\$8:\$L\$13)

Index Nilai (1,1) selalu mengacu kepada sel pertama dari Range, dalam hal ini \$H\$8

Jumlah baris (i) harus dihitung sebelumnya

The screenshot shows an Excel spreadsheet with the following data tables:

Data berupa Range, A(\$C\$4:\$L\$13)

		2	3	4	5	6	7	8	9	10
1		10,294	-9,936,302	-2,798,207	1,502,2			5,836,308	450,663	156,002
2		36,244	-9,749,009	-8,991,200	-7,819,4			-517,533	-1,428,139	-6,332,814
3		447,328	-2,235,159	-9,848,293	2,742,9			-6,697,192	-7,475,933	5,591,883
4		809,335	5,831,059	3,784,675	-3,055,520	7,214,851	-2,936,171	2,144,491	-9,766,274	-9,064,296
5		718,849	-3,703,593	-9,564,509	-745,271	-980,928	-5,935,806	-3,979,301	-3,816	-3,099,424
6		4,930,858	4,438,945	-452,969	-7,000,044	-7,231,060	5,968,502	-3,772,188	3,873,188	-4,683,335
7		5,156,727	4,057,548	1,235,544	-7,052,038	2,384,819	-2,682,841	-9,186,269	2,882,395	-2,492,721
8		-4,713,794	-2,693,279	9,970,836	2,779,980	1,291,234	5,944,900	-7,328,412	-3,276,558	-5,793,310
9		-8,330,567	7,830,607	543,924	-5,539,043			216,667	7,216,175	7,650,624
10		-6,649,897	7,072,704	-6,575,435	8,393,848	9,262,852	-693,054	2,857,921	-9,447,708	-6,013,713

Akses dengan index: NilaiSel (i; j; \$H\$8:\$L\$13)

	1	2	3	4	5	6	7	8	9	10
1	-980,928	-5,935,806	3,979,301	5,235,816	-3,099,424	0,000	0,000	5,000	-7,615,241	-3,718,849
2	-7,231,060	5,968,502	-3,772,188	3,873,188	-4,683,335	0,000	0,000	6,000	-4,294,663	-8,930,858
3	2,384,819	-2,682,841	-9,186,269	2,882,395	-2,492,721	0,000	0,000	7,000	-4,603,915	-5,156,727
4	1,291,234	-5,944,900	-7,328,412	-3,276,558	-5,793,310	0,000	0,000	8,000	-7,937,990	-4,713,794
5	-8,648,296	216,667	7,216,175	7,650,624	1,748,898	0,000	0,000	9,000	-240,348	-8,330,567
6	-693,054	2,857,921	-9,447,708	-6,013,713	-1,871,006	0,000	0,000	10,000	-6,649,897	7,072,704
7	0,000	0	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000
8	0,000	0	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000
9	0,000	0	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000

Akses dengan index: NilaiSel (i; j; \$J\$4:\$J\$13)

	1	2	3	4	5	6	7	8	9	10
1	3,008	450,663	156,002	0,000	0,000	1,000	9,032,315	7,710,294	-9,936,302	-2,798,207
2	3,008	-1,428,139	-6,332,814	0,000	0,000	2,000	-3,697,412	236,244	-9,749,009	-8,991,200
3	3,008	-7,475,933	5,591,883	0,000	0,000	3,000	6,503,103	2,447,328	-2,235,159	-9,848,293
4	3,008	7,830,607	-9,064,296	0,000	0,000	4,000	5,840,794	8,809,335	5,831,059	3,784,675
5	3,008	3,979,301	-3,099,424	0,000	0,000	5,000	-7,615,241	-3,718,849	-3,703,593	-9,564,509
6	3,008	-3,772,188	-4,683,335	0,000	0,000	6,000	-4,294,663	-8,930,858	4,438,945	-452,969
7	3,008	-2,492,721	2,882,395	0,000	0,000	7,000	-4,603,915	-5,156,727	4,057,548	1,235,544
8	3,008	-3,276,558	-5,793,310	0,000	0,000	8,000	-7,937,990	-4,713,794	-2,693,279	9,970,836
9	3,008	7,216,175	7,650,624	0,000	0,000	9,000	-240,348	-8,330,567	7,830,607	543,924

Hasil yang valid hanya dalam kawasan ini (sesuai ukuran data), yang lainnya sampah!

Jumlah kolom (j) harus dihitung sebelumnya

Index pada variable Range Nilai(\$E\$4:\$L\$13)

Jumlah baris (*i*)
harus dihitung
sebelumnya

Index Nilai (1,1) selalu mengacu
kepada sel pertama dari Range,
dalam hal ini \$E\$4

Data berupa Range, A(\$C\$4:\$L\$13)											
		Kolom (j)									
		4	5	6	7	8	9	10			
Baris (i)	1	-3.002,074	-4.190,844	6.408,821	4.382,521	4.493,650	3.526,834	4.916,852	-5.173,179	7.263,431	-8.993,830
	2	-5.370,499	-8.697,036	4.090,901	-4.085,572	0,971	-3.635,526	-7.584,319	2.465,510	-9.010,549	6.260,559
	3	-7.297,024	-6.233,583	3.273,394	9.488,879	3.049,302	4.496,868	3.837,515	4.812,433	-3.963,116	6.513,767
	4	6.889,654	-6.330,674	9.379,961	7.884,185	-3.944,513	-408,361	-108,221	2.331,434	278,845	4.225,929
	5	886,016	8.850,712	6.131,085	9.478,906	5.569,710	-6.596,026	6.393,511	-5.053,901	-4.802,864	0,000
	6	8.566,012	1.319,916	4.090,901	8.594,143	5.171,779	-7.538,073	-2.145,957	8.453,633	-2.651,646	7.717,521
	7	-131,502	-8.097,675	-4.090,901	-3.387,691	4.655,061	1.622,945	5.339,291	-8.773,865	-3.663,421	-4.887,684
	8	-9.179,248	-571,115	-1.072,065	-2.538,843	-1.053,564	-1.137,874	-9.075,528	1.072,065	-2.174,751	-3.466,675
	9	3.262,993	-1.732,103	0,000	8.982,757	-690,760	7.458,535	7.335,957	4.157,558	9.701,575	3.034,676
	10	3.594,628	-9.538,305	0,000	7.715	3.341,643	5.198,087	-1.121,691	884,057	-144,487	-6.869,577

Akses dengan index: Nilai (i; j; \$C\$4:\$L\$13)										
		Kolom (j)								
		3	4	5	6	7	8	9	10	
Baris (i)	1	6.593,821	4.382,521	4.493,650	3.526,834	4.916,852	-5.173,179	7.263,431	-8.993,830	0,000
	2	4.035,901	-4.085,572	0,971	-3.635,526	-7.584,319	2.465,510	-9.010,549	6.260,559	0,000
	3	3.227,394	9.488,879	3.049,302	4.496,868	3.837,515	4.812,433	-3.963,116	6.513,767	0,000
	4	9.812,961	7.884,185	-3.944,513	-408,361	-108,221	2.331,434	278,845	4.225,929	0,000
	5	8.850,712	6.131,085	9.478,906	5.569,710	-6.596,026	6.393,511	-5.053,901	-4.802,864	0,000
	6	8.594,143	5.171,779	-7.538,073	-2.145,957	8.453,633	-2.651,646	7.717,521	0,000	0,000
	7	-8.097,675	-4.090,901	-3.387,691	4.655,061	1.622,945	5.339,291	-8.773,865	-3.663,421	-4.887,684
	8	-571,115	-1.072,065	-2.538,843	-1.053,564	-1.137,874	-9.075,528	1.072,065	-2.174,751	-3.466,675
	9	-1.732,103	0,000	8.982,757	-690,760	7.458,535	7.335,957	4.157,558	9.701,575	3.034,676
	10	-9.538,305	0,000	7.715	3.341,643	5.198,087	-1.121,691	884,057	-144,487	-6.869,577

Akses dengan index: Nilai (i; j; \$H\$5:\$L\$13)												
		Kolom (j)										
		1	2	3	4	5	6	7	8	9	10	
Baris (i)	1	5.569,710	-6.596,026	3.935,511	-5.053,901	-4.802,864	0,000	0,000	5,000	886,016	8.850,712	0,000
	2	-7.538,073	-2.145,957	4.536,333	-2.651,646	7.717,521	0,000	0,000	6,000	8.566,012	1.319,916	0,000
	3	1.622,945	5.339,291	7.733,865	-3.663,421	-4.887,684	0,000	0,000	7,000	-131,502	-8.097,675	0,000
	4	-1.137,874	-9.075,528	1.072,065	-2.174,751	-3.466,675	0,000	0,000	8,000	-9.179,248	-571,115	0,000
	5	7.458,535	7.335,957	4.157,558	9.701,575	3.034,676	0,000	0,000	9,000	3.262,993	-1.732,103	0,000
	6	-1.121,691	0,000	-144,487	-6.869,577	-5.548,750	0,000	0,000	10,000	3.594,628	-9.538,305	0,000
	7	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000
	8	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000
	9	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000
	10	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000

Akses dengan index: Nilai (i; j; \$E\$4:\$E\$13)											
		Kolom (j)									
		1	2	3	4	5	6	7	8	9	10
Baris (i)	1	\$E\$13	4.382,521	4.493,650	3.526,834	4.916,852	-5.173,179	7.263,431	-8.993,830	0,000	0,000
	2	4.035,901	4.085,572	0,971	-3.635,526	-7.584,319	2.465,510	-9.010,549	6.260,559	0,000	0,000
	3	3.227,394	9.488,879	3.049,302	4.496,868	3.837,515	4.812,433	-3.963,116	6.513,767	0,000	0,000
	4	9.812,961	7.884,185	-3.944,513	-408,361	-108,221	2.331,434	278,845	4.225,929	0,000	0,000
	5	3.892,876	6.131,085	9.478,906	5.569,710	-6.596,026	6.393,511	-5.053,901	-4.802,864	0,000	0,000
	6	-7.025,620	8.594,143	5.171,779	-7.538,073	-2.145,957	8.453,633	-2.651,646	7.717,521	0,000	0,000
	7	-4.771,198	-3.387,691	4.655,061	1.622,945	5.339,291	-8.773,865	-3.663,421	-4.887,684	0,000	0,000
	8	-1.759,536	-2.538,843	-1.053,564	-1.137,874	-9.075,528	1.072,065	-2.174,751	-3.466,675	0,000	0,000
	9	1.898,463	8.982,757	-690,760	7.458,535	7.335,957	4.157,558	9.701,575	3.034,676	0,000	0,000
	10	3.594,628	-9.538,305	0,000	7.715	3.341,643	5.198,087	-1.121,691	884,057	-144,487	-6.869,577

Jumlah kolom (*j*)
harus dihitung
sebelumnya

Hasil yang valid hanya dalam
kawasan ini (sesuai ukuran data),
yang lainnya sampah!

oleh
Djoko Luknanto
Departemen Teknik Sipil dan Lingkungan
Fakultas Teknik Universitas Gadjah Mada



BEBERAPA JENIS VBA- EXCEL DO LOOP

Do...Loop statement (1/2)

Syntax

- **Do** [{ **While** | **Until** } *condition*]
[*statements*]
[**Exit Do**]
[*statements*]
Loop

While | Until di depan

Or, you can use this syntax:

- **Do**
[*statements*]
[**Exit Do**]
[*statements*]
Loop [{ **While** | **Until** } *condition*]

While | Until di belakang

Do...Loop statement (2/2)

While | Until di depan

```
Public Sub LoopExample()  
    Dim Check As Boolean, Counter As Long, Total As Long  
    Check = True: Counter = 0: Total = 0 ' Initialize variables.  
    Do ' Outer loop  
        Do While Counter < 20 ' Inner Loop  
            Counter = Counter + 1 ' Increment Counter.  
            If Counter Mod 10 = 0 Then ' Check in with the user on every multiple of 10.  
                Check = (MsgBox("Keep going?", vbYesNo) = vbYes) ' Stop when user click's on No  
                If Not Check Then Exit Do ' Exit inner loop.  
            End If  
        Loop  
        Total = Total + Counter ' Exit Do Lands here.  
        Counter = 0  
    Loop Until Check = False ' Exit outer loop immediately.  
    MsgBox "Counted to: " & Total  
End Sub
```

While | Until di belakang