

DAFTAR PUSTAKA

- Arfa, Y. (2017). Perhitungan Dana Pensiun Dengan Metode Projected Unit Credit Dan Individual Level Premium. *Jurnal Matematika UNAND*, 3, 124–132.
- Brigo, D., & Mercurio, F. (2006). *Interest Rate Models Theory and Practice*. Springer Verlag.
- Bua, F. T. (2022). Formulasi Hukum Mortalitas Makeham dan Aplikasinya dalam Perhitungan Premi Asuransi Jiwa Dwiguna Model Stokastik *Cox-Ingersoll-Ross*= *Formulation of Makeham's Law of Mortality and Its Application in The Calculation of Endowment Life Insurance Premium with The Cox-Ingersoll-Ross Stochastic Model* (Doctoral dissertation, Universitas Hasanuddin).
- Divani, B. T. (2022). *Perhitungan Aktuarial Manfaat Pensiun Normal Dengan Menggunakan Metode Individual Level Premium Dan Attained Age Normal*. Fakultas Sains dan Teknologi Universitas Islam negeri Walisongo.
- Futami, T. (1993). *Matematika Asuransi Jiwa I*. Oriental Life Insurance Cultural Development Center.
- Higham, D. J. (2001). An algorithmic introduction to numerical simulation of stochastic differential equations. *SIAM review*, 43(3), 525-546.
- Lawrence, K. D., Klimberg, R. K., & Lawrence, S. M. (2009). *Fundamentals of Forecasting Using Excel*. Industrial Press Inc.
- Medikasari, W. (2020). *Aplikasi Suku Bunga Konstan dan Suku Bunga Stokastik dalam Perhitungan Aktuarial Asuransi DwiGuna*. Program Studi Matematika Fakultas Sains dan Teknologi UIN Syarif Hidayatullah.
- Nastiti, Z. D. (2015). *Implementation Of Cir Interest Rate Models To Determine Normal Pension Contribution In Accrued Benefit Cost Method*. Departemen of Mathematics Faculty of Mathematics and Natural Sciences Sepuluh Nopember Institute of Technology Surabaya.
- Norman, E. (2021). Manajemen Dana Pensiun Syariah. *Reslaj: Religion Education Social Laa Roiba Journal*, 226–235. <https://doi.org/10.47476/reslaj.v3i2.349>

- PAI. (2019). *Standar Praktik Aktuaria Dana Pensiun SPA-DP*.
- Peraturan Presiden Nomor 16 Tahun 2019 *Tentang Penyesuaian Gaji Pegawai Negeri Sipil*. (2019).
- PP No. 11 Tahun 2017 *Tentang Manajemen Pegawai Negeri Sipil*. (n.d.).
- PP No. 20 Tahun 2013 *Tentang Asuransi Sosial Pegawai Negeri Sipil*. (n.d.).
- PP No. 25 Tahun 1981 *Tentang Asuransi Sosial Pegawai Negeri Sipil*. (n.d.).
- PP No. 45 Tahun 2015 *Tentang Penyelenggaraan Program Jaminan Pensiun*. (n.d.).
- Rivanda, M. R. (2019). *Penentuan Pembiayaan Dana Pensiun Dengan Metode Attained Age Normal, Projected Unit Credit Dan Entry Age Normal Skripsi Program Studi Matematika*. Program Studi Matematika Fakultas Sains dan Teknologi Uin Syarif Hidayatullah .
- Sagala, S. M. (2018). Perbandingan Metode Projected Unit Credit dengan Individual Level Premium dalam Perhitungan Aktuaria Dana Pensiun. *The University Institutional Repository*.
- Solidayah, W., Sunendiari, S., & Wachidah, L. (2015). Modifikasi Peringkat Bertanda Wilcoxon Untuk Masalah Dua Sampel Berpasangan. *Prosiding Statistika, 1*, 19–26.
- Taylor, H. M., & Karlin, S. (1998). *An Introduction to Stochastic Modeling, Third Edition*. Academic Press.
- UU No. 11 Tahun 1992 *Tentang Dana Pensiun*. (n.d.).
- utama, andrew shandy. (2022). Penyelesaian Sengketa Dana Pensiun Berdasarkan Undang - Undang Nomor 11 Tahun 1992. *Journal of Juridische Analyse, 2830–6023*.
- Wardhani, I. G. A. K. K., Widana, I. N., & Tastarawati, N. K. (2014). Perhitungan Dana Pensiun Dengan Metode Projected Unit Credit Dan Individual Level Premium. *E-Jurnal Matematika, 2303–75, 64–74*. <http://ojs.unud.ac.id/index.php/mtk>

Winklevoss, H. E. (1993). *Pension Mathematics with Numerical Illustration* (Second Edition). University of Pennsylvania Press.

LAMPIRAN

Lampiran 1. Tabel Mortalitas

Usia	Laki-Laki	Perempuan
0	0,00524	0,00266
1	0,00053	0,00041
2	0,00042	0,00031
3	0,00034	0,00024
4	0,00029	0,00021
5	0,00026	0,00020
6	0,00023	0,00022
7	0,00021	0,00023
8	0,00020	0,00022
9	0,00020	0,00021
10	0,00019	0,00019
11	0,00019	0,00018
12	0,00019	0,00020
13	0,00020	0,00022
14	0,00023	0,00023
15	0,00027	0,00023
16	0,00031	0,00024
17	0,00037	0,00024
18	0,00043	0,00025
19	0,00047	0,00026
20	0,00049	0,00027
21	0,00049	0,00028
22	0,00049	0,00030
23	0,00049	0,00032
24	0,00050	0,00034
25	0,00052	0,00038
26	0,00055	0,00042
27	0,00060	0,00046
28	0,00065	0,00049
29	0,00070	0,00052
30	0,00075	0,00056
31	0,00081	0,00060
32	0,00087	0,00064
33	0,00093	0,00069
34	0,00099	0,00074
35	0,00107	0,00080
36	0,00116	0,00086
37	0,00127	0,00093

Usia	Laki-Laki	Perempuan
38	0,00139	0,00100
39	0,00155	0,00108
40	0,00173	0,00118
41	0,00193	0,00128
42	0,00216	0,00141
43	0,00241	0,00154
44	0,00270	0,00169
45	0,00302	0,00187
46	0,00338	0,00209
47	0,00377	0,00230
48	0,00418	0,00253
49	0,00461	0,00277
50	0,00508	0,00305
51	0,00556	0,00335
52	0,00609	0,00368
53	0,00667	0,00403
54	0,00727	0,00442
55	0,00789	0,00483
56	0,00847	0,00524
57	0,00898	0,00563
58	0,00939	0,00601
59	0,00971	0,00636
60	0,00999	0,00671
61	0,01024	0,00707
62	0,01046	0,00746
63	0,01071	0,00788
64	0,01104	0,00833
65	0,01146	0,00883
66	0,01199	0,00940
67	0,01260	0,01005
68	0,01329	0,01076
69	0,01405	0,01150
70	0,01485	0,01229
71	0,01574	0,01314
72	0,01670	0,01406
73	0,01777	0,01508
74	0,01895	0,01620
75	0,02026	0,01743

Usia	Laki-Laki	Perempuan
76	0,02369	0,01879
77	0,02738	0,02030
78	0,03130	0,02326
79	0,03693	0,02880
80	0,04518	0,03569
81	0,05527	0,04208
82	0,06732	0,04907
83	0,08228	0,05520
84	0,09478	0,06086
85	0,10465	0,06715
86	0,11533	0,07318
87	0,12698	0,08155
88	0,13947	0,09045
89	0,15271	0,10001
90	0,16659	0,10913
91	0,17991	0,11521
92	0,19390	0,12499
93	0,20874	0,13826
94	0,22451	0,15451
95	0,24126	0,17429
96	0,25715	0,19155
97	0,27419	0,20596
98	0,29249	0,22227
99	0,31215	0,23736
100	0,33331	0,25810
101	0,35163	0,28068
102	0,37132	0,30562
103	0,39250	0,33315
104	0,41527	0,36369
105	0,43973	0,39318
106	0,46602	0,42883
107	0,49429	0,46604
108	0,52467	0,50427
109	0,55733	0,54477
110	0,59244	0,58702
111	1,00000	1,00000

Lampiran 2. Perhitungan Pendanaan Pensiun Pegawai A

X	v	$l_x^{(T)}$	$D_x^{(T)}$	$N_x^{(T)}$
20	0,35423	1000000	354225	1582797,64
21	0,33649	756292	254483	1228572,39
22	0,31964	586023	187315	974089,675
23	0,30363	464265	140965	786775,051
24	0,28842	375358	108262	645810,311
25	0,27398	309132	84695,9	537548,013
26	0,26026	258938	67390,8	452852,151
27	0,24722	220251	54451,4	385461,381
28	0,23484	189999	44619,9	331010,018
29	0,22308	166004	37032,4	286390,151
30	0,21191	146724	31092,1	249357,778
31	0,2013	131027	26375,2	218265,676
32	0,19121	118116	22585,5	191890,484
33	0,18164	107392	19506,5	169304,998
34	0,17254	98375	16973,7	149798,536
35	0,1639	90727	14870,1	132824,811
36	0,15569	84176	13105,5	117954,681
37	0,14789	78503	11610,1	104849,218
38	0,14049	73543	10331,8	93239,114
39	0,13345	69169	9230,66	82907,2894
40	0,12677	65276	8274,86	73676,6245
41	0,12042	61777	7439,09	65401,7611
42	0,11439	58605	6703,68	57962,6694
43	0,10866	55695	6051,75	51258,9847
44	0,10322	53006	5471,11	45207,2356
45	0,09805	50499	4951,3	39736,1243
46	0,09314	48145	4484,08	34784,826
47	0,08847	45926	4063,18	30300,7465
48	0,08404	43818	3682,53	26237,562
49	0,07983	41808	3337,64	22555,0322
50	0,07583	39884	3024,57	19217,3966
51	0,07204	38039	2740,19	16192,8239
52	0,06843	36270	2481,9	13452,6368
53	0,065	34570	2247,1	10970,7356
54	0,06175	32942	2034,04	8723,6371
55	0,05865	31383	1840,72	6689,60175
56	0,05572	30960	1724,97	4848,87726
57	0,05293	30508	1614,65	3123,90908

58	0,05027	30020	1509,25	1509,25432
----	---------	-------	---------	------------

Lampiran 3. Perhitungan Pendanaan Pensiun Pegawai B

X	v	$l_x^{(T)}$	$D_x^{(T)}$	$N_x^{(T)}$
30	0,211909	146724	31092,1	249357,8
31	0,201296	131027	26375,19	218265,7
32	0,191214	118116	22585,49	191890,5
33	0,181638	107392	19506,46	169305
34	0,172541	98375	16973,72	149798,5
35	0,1639	90727	14870,13	132824,8
36	0,155691	84176	13105,46	117954,7
37	0,147894	78503	11610,1	104849,2
38	0,140487	73543	10331,82	93239,11
39	0,133451	69169	9230,665	82907,29
40	0,126767	65276	8274,863	73676,62
41	0,120418	61777	7439,092	65401,76
42	0,114388	58605	6703,685	57962,67
43	0,108659	55695	6051,749	51258,98
44	0,103217	53006	5471,111	45207,24
45	0,098047	50499	4951,298	39736,12
46	0,093137	48145	4484,08	34784,83
47	0,088472	45926	4063,184	30300,75
48	0,084041	43818	3682,53	26237,56
49	0,079832	41808	3337,636	22555,03
50	0,075834	39884	3024,573	19217,4
51	0,072036	38039	2740,187	16192,82
52	0,068428	36270	2481,901	13452,64
53	0,065001	34570	2247,099	10970,74
54	0,061746	32942	2034,035	8723,637
55	0,058654	31383	1840,724	6689,602
56	0,055716	30960	1724,968	4848,877
57	0,052926	30508	1614,655	3123,909
58	0,050275	30020	1509,254	1509,254

Lampiran 4. Suku Bunga CIR dan Faktor Diskon CIR

T	Suku Bunga CIR	$\frac{1}{(1 + r_t)}$	v^t
58	0,049984534	0,952395	1
59	0,049195865	0,953111	0,907738
60	0,049036508	0,953256	0,865306
61	0,049090664	0,953206	0,824816
62	0,049544659	0,952794	0,78588
63	0,049479947	0,952853	0,748828
64	0,048918277	0,953363	0,713905
65	0,049858541	0,952509	0,680001
66	0,049319411	0,952999	0,64804
67	0,048765197	0,953502	0,617907
68	0,048397975	0,953836	0,589383
69	0,048815562	0,953456	0,561951
70	0,050252467	0,952152	0,535062
71	0,049507785	0,952828	0,509822
72	0,049211241	0,953097	0,48591
73	0,049302961	0,953014	0,463079
74	0,049610962	0,952734	0,441191
75	0,048905239	0,953375	0,42062
76	0,049442109	0,952887	0,400804
77	0,049674643	0,952676	0,381836
78	0,04975013	0,952608	0,36374
79	0,048524285	0,953721	0,346907
80	0,048856753	0,953419	0,330747
81	0,049309242	0,953008	0,315205
82	0,049064444	0,95323	0,300463
83	0,049625814	0,95272	0,286257
84	0,050061082	0,952326	0,27261
85	0,048196404	0,95402	0,260075
86	0,048491448	0,953751	0,248047
87	0,049097572	0,9532	0,236439
88	0,048963215	0,953322	0,225402
89	0,049691061	0,952661	0,214732
90	0,049970021	0,952408	0,204512
91	0,04957612	0,952766	0,194852
92	0,049303984	0,953013	0,185697
93	0,049025043	0,953266	0,177018
94	0,049984534	0,952395	0,168591
95	0,049195865	0,953111	0,160686

96	0,049036508	0,953256	0,153175
97	0,049090664	0,953206	0,146008
98	0,049544659	0,952794	0,139115
99	0,049479947	0,952853	0,132556
100	0,048918277	0,953363	0,126374
101	0,049858541	0,952509	0,120373
102	0,049319411	0,952999	0,114715
103	0,048765197	0,953502	0,109381
104	0,048397975	0,953836	0,104332
105	0,049479947	0,952853	0,099413
106	0,048918277	0,953363	0,094776
107	0,049858541	0,952509	0,090275
108	0,049319411	0,952999	0,086032
109	0,048765197	0,953502	0,082032
110	0,048397975	0,953836	0,078245

Lampiran 5. Kodingan Python

```
#Mengimport Library yang dibutuhkan
import math
import pandas as pd
import matplotlib.pyplot as plt
import numpy as np

#Import Data Insample
Data_Insample = pd.read_excel('/content/Insample.xlsx')
Data_Insample.head()

#Membuat Grafik Data Insample
plt.figure(figsize=(12, 6))
plt.plot(Data_Insample['Waktu'], Data_Insample['Nilai
Suku Bunga'])
plt.title('Pergerakan BI Rates ')
plt.xlabel('Tahun')
plt.ylabel('Nilai Suku Bunga')
plt.grid(True)
plt.show()

N = len(Data_Insample)
N

import pandas as pd

# Menggunakan atribut .values untuk mengubah DataFrame
menjadi array NumPy
tsb = Data_Insample['Nilai Suku Bunga'].values
print(tsb)

# Estimasi Parameter Alpha dan Miu

ip = []
rtp = []
rt1p = []
rt2p = []
rt3p = []
srt = 0
srt1 = 0
srt2 = 0
srt3 = 0
```

```

for i in range((N-1)):
    rt = tsb[i]
    rt1 = tsb[i + 1]
    rt2 = 1 / rt
    rt3 = rt1 / rt
    SRT = srt + rt
    SRT1 = srt1 + rt1
    SRT2 = srt2 + rt2
    SRT3 = srt3 + rt3
    srt = SRT
    srt1 = SRT1
    srt2 = SRT2
    srt3 = SRT3
    ip.append(i)
    rtp.append(rt)
    rt1p.append(rt1)
    rt2p.append(rt2)
    rt3p.append(rt3)

CIR_est_data = {
    't': ip,
    'r(t)': rtp,
    'r(t+1)': rt1p,
    '1/r(t)': rt2p,
    'r(t+1)/r(t)': rt3p
}

df = pd.DataFrame(CIR_est_data)
print(df)

print('Diperoleh:')
print('R1 =', "%.9f" % SRT)
print('R2 =', "%.9f" % SRT1)
print('R3 =', "%.9f" % SRT2)
print('R4 =', "%.9f" % SRT3)

a = (N**2 - 2*N + 1 + SRT1*SRT2 - SRT*SRT2 - (N-1)*SRT3) / (N**2 - 2*N + 1 - SRT*SRT2)
mu = ((N-1)*SRT1 - SRT3*SRT) / (N**2 - 2*N + 1 + SRT1*SRT2 - SRT*SRT2 - (N-1)*SRT3)

print('Estimasi a =', "%.9f" % a)
print('Estimasi mu =', "%.9f" % mu)
import numpy as np

```

```

# Estimasi Parameter Sigma
n = 136
rt = tsb
mu = 0.051543334
alpha = 0.029737958
sum_term = 0
for i in range(1, len(rt)):
    if rt[i - 1] > 0:
        sum_term += (np.sqrt(rt[i]) - np.sqrt(rt[i - 1]) - mu * np.sqrt(rt[i - 1]) + alpha * np.sqrt(rt[i - 1])) ** 2
if n > 2:
    sigma_hat = np.sqrt(1 / (n - 2) * sum_term / (n - 1))
else:
    sigma_hat = None

# Tampilkan hasil
if sigma_hat is not None:
    print("σ^ :", sigma_hat)
else:
    print("Tidak bisa menghitung σ^ karena n ≤ 2")

# Membuat Simulasi Lintasan
import numpy as np
import matplotlib.pyplot as plt

def CIR_model(rt_minus_1, alpha, mu, sigma, dt):
    dW = np.sqrt(dt) * np.random.normal(0, 1)
    rt = rt_minus_1 + alpha * (mu - rt_minus_1) * dt + sigma * np.sqrt(rt_minus_1) * dW + (1/4) * sigma**2 * (dW**2 - dt)
    return rt

rt_minus_1 = 0.04
alpha = 0.029737958
mu = 0.051543334
sigma = 0.0006426645950542172
T = 3
N = 100
dt = T/N

num_paths = 36 # Number of paths to simulate
paths = [] # Initialize a list to store the paths

```

```

for _ in range(num_paths):
    r0 = rt_minus_1
    r_path = [r0]

    for _ in range(N):
        r = CIR_model(r0, alpha, mu, sigma, dt)
        r_path.append(r)
        r0 = r
final_rates = [path[-1] for path in paths]
best_path_index = np.argmax(final_rates)
best_path = paths[best_path_index]
best_final_rate = final_rates[best_path_index]

# Plot lintasan terbaik
plt.plot(np.linspace(0, T, N + 1), best_path)
plt.xlabel('Time (Year)')
plt.ylabel('Predicted Interest Rate')
plt.title('CIR Model Paths')
plt.legend()
plt.show()

    paths.append(r_path)

# Plot all the paths together
for path in paths:
    plt.plot(np.linspace(0, T, N + 1), path)

plt.xlabel('Time (Years)')
plt.ylabel('Predicted Interest Rate')
plt.title('CIR Model Paths')
plt.show()

import pandas as pd

# Data lintasan suku bunga akhir
final_rates = [path[-1] for path in paths]

# Membuat DataFrame
df = pd.DataFrame({
    'Lintasan': range(1, len(paths) + 1), # Nomor
    lintasan
    'Prediksi Tingkat Suku Bunga': final_rates
})

```

```
# Menyimpan DataFrame dalam format Excel
excel_file = "lintasan_suku_bunga1.xlsx"
df.to_excel(excel_file, index=False)

# Menghitung MAPE
predict =
pd.read_excel('/content/lintasan_suku_bunga1.xlsx')
predict
Acc = pd.read_excel('/content/Data Outsample vs Data
Prediksi.xlsx')
actual = Acc['Bunga Aktual']
predicted = Acc['Bunga Prediksi']
mape = np.mean(np.abs((actual - predicted) / actual)) *
100

print(f'MAPE: {mape:.2f}%')
```

Lampiran 6 Tabel Gaji Pegawai Sesuai Peraturan Presiden Republik Indonesia Nomor 16 Tahun 2019 Tentang Penyesuaian Gaji Pokok Pegawai Negeri Sipil Menurut Peraturan Pemerintah Nomor 30 Tahun 2015 ke Dalam Gaji Pokok Pegawai Negeri Sipil Menurut Peraturan Pemerintah Nomor 15 Tahun 2015

MK G	Golongan II				Golongan III				Golongan IV				
	A	b	c	D	a	b	c	d	a	b	c	d	e
0	2.022.200				2.579.400	2.688.500	2.802.300	2.920.800	3.044.300	3.173.100	3.307.300	3.447.200	3.593.100
1	2.054.100												
2					2.660.700	2.773.200	2.890.500	3.012.800	3.140.200	3.273.100	3.411.500	3.555.800	3.706.200
3	2.118.800	2.208.400	2.301.800	2.399.200									
4					2.744.500	2.860.500	2.981.500	3.107.700	3.239.100	3.376.100	3.518.900	3.667.800	3.822.900
5	2.185.500	2.277.900	2.374.300	2.474.700									
6					2.830.900	2.950.600	3.075.500	3.205.500	3.341.100	3.482.500	3.629.800	3.783.300	3.943.300
7	2.254.300	2.349.700	2.449.100	2.552.700									
8					2.920.100	3.043.600	3.172.300	3.306.500	3.446.400	3.592.100	3.744.100	3.902.500	4.067.500
9	2.325.300	2.423.700	2.526.200	2.633.100									
10					3.012.000	3.139.400	3.272.200	3.410.600	3.554.900	3.705.300	3.862.000	4.025.400	4.195.700
11	2.398.600	2.500.000	2.605.800	2.716.000									
12					3.106.900	3.238.300	3.375.300	3.518.100	3.666.900	3.822.000	3.983.600	4.152.200	4.327.800
13	2.474.100	2.578.800	2.687.800	2.801.500									
14					3.204.700	3.340.300	3.481.600	3.628.900	3.782.400	3.942.400	4.109.100	4.282.900	4.464.100
15	2.552.000	2.660.000	2.772.500	2.889.800									
16					3.305.700	3.445.500	3.591.200	3.743.100	3.901.500	4.066.500	4.238.500	4.417.800	4.604.700
17	2.632.400	2.743.800	2.859.800	2.980.800									
18					3.409.800	3.554.000	3.704.300	3.861.000	4.024.400	4.194.600	4.372.000	4.557.000	4.749.700
19	2.715.300	2.830.200	2.949.900	3.074.700									
20					3.517.200	3.665.900	3.821.000	3.982.600	4.151.100	4.326.700	4.509.700	4.700.500	4.899.300
21	2.800.800	2.919.300	3.042.800	3.171.500									
22					3.627.900	3.781.400	3.941.400	4.108.100	4.281.800	4.463.000	4.651.800	4.848.500	5.053.600
23	2.889.100	3.011.300	3.138.600	3.271.400									
24					3.742.200	3.900.500	4.065.500	4.237.500	4.416.700	4.603.500	4.798.300	5.001.200	5.212.800
25	2.980.000	3.106.100	3.237.500	3.374.400									
26					3.860.100	4.023.300	4.193.500	4.370.900	4.555.800	4.748.500	4.949.400	5.158.700	5.377.000
27	3.073.900	3.203.900	3.339.400	3.480.700									
28					3.981.600	4.150.100	4.325.600	4.508.600	4.699.300	4.898.100	5.105.300	5.321.200	5.546.300
29	3.170.700	3.304.800	3.444.600	3.590.300									
30					4.107.000	4.280.800	4.461.800	4.650.600	4.847.300	5.052.300	5.266.100	5.488.800	5.721.000
31	3.270.600	3.408.900	3.553.100	3.703.400									
32					4.236.400	4.415.600	4.602.400	4.797.000	5.000.000	5.211.500	5.431.900	5.661.700	5.901.200
33	3.373.600	3.516.300	3.665.000	3.820.000									

Lampiran 7 Service Table

x	l_x^T	d_x^m	d_x^t	d_x^d	d_x^r	d_x^T	$r-xp_x^{(T)}$
20	1.000.000	442	243.002	263	0	243.708	0,02949
21	756.292	350	169.718	201	0	170.270	0,03899
22	586.023	286	121.314	158	0	11.757	0,05032
23	464.265	238	88.543	126	0	88.907	0,06352
24	375.358	202	65.921	103	0	66.226	0,07857
25	309.132	176	49.933	85	0	50.194	0,09540
26	258.938	156	38.460	72	0	38.688	0,11389
27	220.251	140	30.049	62	0	30.251	0,13390
28	189.999	129	23.814	53	0	23.996	0,15522
29	166.004	119	19.113	47	0	19.280	0,17765
30	146.724	112	15.529	56	0	15.697	0,20100
31	131.027	107	12.754	50	0	12.911	0,22508
32	118.116	103	10.576	45	0	10.725	0,24968

33	107.392	101	8.875	41	0	9.017	0,27461
34	98.375	99	7.510	38	0	7.647	0,29978
35	90.727	98	6.419	35	0	6.552	0,32505
36	84.176	98	5.534	41	0	5.673	0,35035
37	78.503	99	4.816	46	0	4.960	0,37567
38	73.543	100	4.224	50	0	4.374	0,40100
39	69.169	102	3.738	54	0	3.893	0,42636
40	65.276	104	3.338	57	0	3.499	0,45179
41	61.777	108	3.004	60	0	3.172	0,47738
42	58.605	114	2.727	69	0	2.910	0,50322
43	55.695	123	2.491	76	0	2.690	0,52951
44	53.006	133	2.290	83	0	2.506	0,55637
45	50.499	144	2.121	89	0	2.354	0,58399
46	48.145	156	1.969	94	0	2.219	0,61255
47	45.926	169	1.841	99	0	2.108	0,64214
48	43.818	181	1.721	107	0	2.009	0,67303
49	41.808	194	1.616	115	0	1.925	0,70539
50	39.884	206	1.517	121	0	1.845	0,73942
51	38.039	219	1.424	127	0	1.769	0,77528
52	36.270	230	1.335	135	0	1.700	0,81310
53	34.570	241	1.244	142	0	1.628	0,85308
54	32.942	252	1.159	148	0	1.559	0,89524
55	31.383	267	0	156	0	423	0,93971
56	30.960	286	0	166	0	452	0,95255
57	30.508	305	0	182	0	487	0,96666
58	29.491	326	0	203	0	529	1,00000