LAMPIRAN I

TABLE PERIODIC TERMS FOR PLANET : EARTH[1]

[1] Jean Meeus, *Astronomical Algorithms,* First Edition, Virginia: Willman-Bell, Inc., 1991, hlm. 381

Appendix III

Planets : Periodic Terms

In this Appendix, pages 414–454, the most important periodic terms from the French planetary theory VSOP87 are given. The successive columns contain the following data:

— the name of the planet;

— the label of the series (L for the heliocentric longitude, B for the latitude, R for the radius vector);

— the current No. of the term in the series;

— the quantities A, B, and C, which all are positive (or zero).

In each series, the terms are sorted by decreasing values of A.

For example:

Planet	Series	No.	A	B	C
VENUS	R0	1	72334821	0	0
		2	489824	4.021518	10213.285546
		3	1658	4.9021	20426.5711
		4	1632	2.8455	7860.4194
		5	1378	1.1285	11790.6291
		6	498	2.587	9683.595
		7	374	1.423	3930.210
		8	264	5.529	9437.763
		9	237	2.551	15720.839
		10	222	2.013	19367.189
		11	126	2.728	1577.344
		12	119	3.020	10404.734
VENUS	R1	1	34551	0.89199	10213.28555
		2	234	1.772	20426.571
		3	234	3.142	0

For more explanation about the use of these terms, see Chapter 32.

VENUS	B4	1	14	0.32	10213.29
VENUS	R0	1	72334821	0	0
		2	489824	4.021518	10213.285546
		3	1658	4.9021	20426.5711
		4	1632	2.8455	7860.4194
		5	1378	1.1285	11790.6291
		6	498	2.587	9683.595
		7	374	1.423	3930.210
		8	264	5.529	9437.763
		9	237	2.551	15720.839
		10	222	2.013	19367.189
		11	126	2.728	1577.344
		12	119	3.020	10404.734
VENUS	R1	1	34551	0.89199	10213.28555
		2	234	1.772	20426.571
		3	234	3.142	0
VENUS	R2	1	1407	5.0637	10213.2855
		2	16	5.47	20426.57
		3	13	0	0
VENUS	R3	1	50	3.22	10213.29
VENUS	R4	1	1	0.92	10213.29
EARTH	L0	1	175347046	0	0
		2	3341656	4.669256 8	6283.075 850 0
		3	34894	4.626 10	12566.151 70
		4	3497	2.744 1	5753.384 9
		5	3418	2.828 9	3.523 1
		6	3136	3.627 7	77713.771 5
		7	2676	4.418 1	7860.419 4
		8	2343	6.135 2	3930.209 7
		9	1324	0.742 5	11506.769 8
		10	1273	2.037 1	529.691 0
		11	1199	1.109 6	1577.343 5
		12	990	5.233	5884.927
		13	902	2.045	26.298
		14	857	3.508	398.149
		15	780	1.179	5223.694
		16	753	2.533	5507.553
		17	505	4.583	18849.228
		18	492	4.205	775.523
		19	357	2.920	0.067
		20	317	5.849	11790.629
		21	284	1.899	796.298
		22	271	0.315	10977.079
		23	243	0.345	5486.778
		24	206	4.806	2544.314
		25	205	1.869	5573.143
		26	202	2.458	6069.777
		27	156	0.833	213.299
		28	132	3.411	2942.463
		29	126	1.083	20.775
		30	115	0.645	0.980
		31	103	0.636	4694.003
		32	102	0.976	15720.839
		33	102	4.267	7.114
		34	99	6.21	2146.17
		35	98	0.68	155.42
		36	86	5.98	161000.69

EARTH (cont.)	L0	37	85	1.30	6 275.96
		38	85	3.67	71 430.70
		39	80	1.81	17 260.15
		40	79	3.04	12 036.46
		41	75	1.76	5 088.63
		42	74	3.50	3 154.69
		43	74	4.68	801.82
		44	70	0.83	9 437.76
		45	62	3.98	8 827.39
		46	61	1.82	7 084.90
		47	57	2.78	6 286.60
		48	56	4.39	14 143.50
		49	56	3.47	6 279.55
		50	52	0.19	12 139.55
		51	52	1.33	1 748.02
		52	51	0.28	5 856.48
		53	49	0.49	1 194.45
		54	41	5.37	8 429.24
		55	41	2.40	19 651.05
		56	39	6.17	10 447.39
		57	37	6.04	10 213.29
		58	37	2.57	1 059.38
		59	36	1.71	2 352.87
		60	36	1.78	6 812.77
		61	33	0.59	17 789.85
		62	30	0.44	83 996.85
		63	30	2.74	1 349.87
		64	25	3.16	4 690.48
EARTH	L1	1	628 331 966 747	0	0
		2	206 059	2.678 235	6 283.075 850
		3	4 303	2.635 1	12 566.151 7
		4	425	1.590	3.523
		5	119	5.796	26.298
		6	109	2.966	1 577.344
		7	93	2.59	18 849.23
		8	72	1.14	529.69
		9	68	1.87	398.15
		10	67	4.41	5 507.55
		11	59	2.89	5 223.69
		12	56	2.17	155.42
		13	45	0.40	796.30
		14	36	0.47	775.52
		15	29	2.65	7.11
		16	21	5.34	0.98
		17	19	1.85	5 486.78
		18	19	4.97	213.30
		19	17	2.99	6 275.96
		20	16	0.03	2 544.31
		21	16	1.43	2 146.17
		22	15	1.21	10 977.08
		23	12	2.83	1 748.02
		24	12	3.26	5 088.63
		25	12	5.27	1 194.45
		26	12	2.08	4 694.00
		27	11	0.77	553.57
		28	10	1.30	6 286.60
		29	10	4.24	1 349.87
		30	9	2.70	242.73
		31	9	5.64	951.72
		32	8	5.30	2 352.87
		33	6	2.65	9 437.76
		34	6	4.67	4 690.48

EARTH (cont.)	R0	20	86	1.27	161 000.69
		21	65	0.27	17 260.15
		22	63	0.92	529.69
		23	57	2.01	83 996.85
		24	56	5.24	71 430.70
		25	49	3.25	2 544.31
		26	47	2.58	775.52
		27	45	5.54	9 437.76
		28	43	6.01	6 275.96
		29	39	5.36	4 694.00
		30	38	2.39	8 827.39
		31	37	0.83	19 651.05
		32	37	4.90	12 139.55
		33	36	1.67	12 036.46
		34	35	1.84	2 942.46
		35	33	0.24	7 084.90
		36	32	0.18	5 088.63
		37	32	1.78	398.15
		38	28	1.21	6 286.60
		39	28	1.90	6 279.55
		40	26	4.59	10 447.39
EARTH	R1	1	103 019	1.107 490	6 283.075 850
		2	1 721	1.0644	12 566.151 7
		3	702	3.142	0
		4	32	1.02	18 849.23
		5	31	2.84	5 507.55
		6	25	1.32	5 223.69
		7	18	1.42	1 577.34
		8	10	5.91	10 977.08
		9	9	1.42	6 275.96
		10	9	0.27	5 486.78
EARTH	R2	1	4 359	5.7846	6 283.075 8
		2	124	5.579	12 566.152
		3	12	3.14	0
		4	9	3.63	77 713.77
		5	6	1.87	5 573.14
		6	3	5.47	18 849.23
EARTH	R3	1	145	4.273	6 283.076
		2	7	3.92	12 566.15
EARTH	R4	1	4	2.56	6 283.08
MARS	L0	1	620 347 712	0	0
		2	18 656 368	5.050 371 00	3 340.612 426 70
		3	1 108 217	5.400 998 4	6 681.224 853 4
		4	91 798	5.754 79	10 021.837 28
		5	27 745	5.970 50	3.523 12
		6	12 316	0.849 56	2 810.921 46
		7	10 610	2.939 59	2 281.230 50
		8	8 927	4.157 0	0.017 3
		9	8 716	6.110 1	13 362.449 7
		10	7 775	3.339 7	5 621.842 9
		11	6 798	0.364 6	398.149 0
		12	4 161	0.228 1	2 942.463 4
		13	3 575	1.661 9	2 544.314 4
		14	3 075	0.857 0	191.448 3
		15	2 938	6.078 9	0.067 3
		16	2 628	0.648 1	3 337.089 3

EARTH	L2	1	52 919	0	0
		2	8 720	1.072 1	6 283.075 8
		3	309	0.867	12 566.152
		4	27	0.05	3.52
		5	16	5.19	26.30
		6	16	3.68	155.42
		7	10	0.76	18 849.23
		8	9	2.06	77 713.77
		9	7	0.83	775.52
		10	5	4.66	1 577.34
		11	4	1.03	7.11
		12	4	3.44	5 573.14
		13	3	5.14	796.30
		14	3	6.05	5 507.55
		15	3	1.19	242.73
		16	3	6.12	529.69
		17	3	0.31	398.15
		18	3	2.28	553.57
		19	2	4.38	5 223.69
		20	2	3.75	0.98
EARTH	L3	1	289	5.844	6 283.076
		2	35	0	0
		3	17	5.49	12 566.15
		4	3	5.20	155.42
		5	1	4.72	3.52
		6	1	5.30	18 849.23
		7	1	5.97	242.73
EARTH	L4	1	114	3.142	0
		2	8	4.13	6 283.08
		3	1	3.84	12 566.15
EARTH	L5	1	1	3.14	0
EARTH	B0	1	280	3.199	84 334.662
		2	102	5.422	5 507.553
		3	80	3.88	5 223.69
		4	44	3.70	2 352.87
		5	32	4.00	1 577.34
EARTH	B1	1	9	3.90	5 507.55
		2	6	1.73	5 223.69
EARTH	R0	1	100 013 989	0	0
		2	1 670 700	3.098 463 5	6 283.075 850 0
		3	13 956	3.055 25	12 566.151 70
		4	3 084	5.198 5	77 713.771 5
		5	1 628	1.173 9	5 753.384 9
		6	1 576	2.846 9	7 860.419 4
		7	925	5.453	11 506.770
		8	542	4.564	3 930.210
		9	472	3.661	5 884.927
		10	346	0.964	5 507.553
		11	329	5.900	5 223.694
		12	307	0.299	5 573.143
		13	243	4.273	11 790.629
		14	212	5.847	1 577.344
		15	186	5.022	10 977.079
		16	175	3.012	18 849.228
		17	110	5.055	5 486.778
		18	98	0.89	6 069.78
		19	86	5.69	15 720.84

LAMPIRAN II

TABEL PERIODIC TERMS FOR NUTATION IN LONGITUDE[2]

[2] Jean Meeus, *Astronomical Algorithms,* First Edition, Virginia: Willman-Bell, Inc., 1991, hlm. 133

Periodic Terms for the nutation in longitude ($\Delta\psi$). The unit is 0,0001 second

Argument Multiple of					$\Delta\psi$ Coefficient of the Sine of the Argument	
D	M_o	M_c	F	Ω^c		
0	0	0	0	1	-171996	-174,2
-2	0	0	2	2	-13187	-1,6
0	0	0	2	2	-2274	-0,2
0	0	0	0	2	2062	0,2
0	1	0	0	0	1426	-3,4
0	0	1	0	0	712	0,1
-2	1	0	2	2	-517	1,2
0	0	0	2	1	-386	-0,4
-2	-1	0	2	2	217	-0,5
-2	0	0	2	1	129	0,1
0	0	1	0	1	63	0,1
0	0	-1	0	1	-58	-0,1
0	2	0	0	0	17	-0,1
-2	2	0	2	2	-16	0,1
0	0	1	2	2	-301	0
-2	0	1	0	0	-158	0
0	0	-1	2	2	123	0
2	0	0	0	0	63	0
2	0	-1	2	2	-59	0
0	0	1	2	1	-51	0
-2	0	2	0	0	48	0
0	0	-2	2	1	46	0
2	0	0	2	2	-38	0
0	0	2	2	2	-31	0
0	0	2	0	0	29	0

-2	0	1	2	2	29	0
0	0	0	2	0	26	0
-2	0	0	2	0	-22	0
0	0	-1	2	1	21	0
2	0	-1	0	1	16	0
0	1	0	0	1	-15	0
-2	0	1	0	1	-13	0
0	-1	0	0	1	-12	0
0	0	2	-2	0	11	0
2	0	-1	2	1	-10	0
2	0	1	2	2	-8	0
0	1	0	2	2	7	0
-2	1	1	0	0	-7	0
0	-1	0	2	2	-7	0
2	0	0	2	1	-7	0
2	0	1	0	0	6	0
-2	0	2	2	2	6	0
-2	0	1	2	1	6	0
2	0	-2	0	1	-6	0
2	0	0	0	1	-6	0
0	-1	1	0	0	5	0
-2	-1	0	2	1	-5	0
-2	0	0	0	1	-5	0
0	0	2	2	1	-5	0
-2	0	2	0	1	4	0
-2	1	0	2	1	4	0
0	0	1	-2	0	4	0
-1	0	1	0	0	-4	0
-2	1	0	0	0	-4	0

1	0	0	0	0	-4	0
0	0	1	2	0	3	0
0	0	-2	2	2	-3	0
-1	-1	1	0	0	-3	0
0	1	1	0	0	-3	0
0	-1	1	2	2	-3	0
2	-1	-1	2	2	-3	0
0	0	3	2	2	-3	0
2	-1	0	2	2	-3	0

LAMPIRAN III

TABEL PERIODIC TERMS FOR NUTATION IN OBLIQUITY[3]

[3] Jean Meeus, *Astronomical Algorithms,* First Edition, Virginia: Willman-Bell, Inc., 1991, hlm. 134

Periodic Terms for the nutation in obliquity (ε_o). The unit is 0,0001 second

Argument multiple of					$\Delta\varepsilon$	
D	M_o	M_c	F	Ω^c	Coeffisient of the cosine of the Arguments	
0	0	0	0	1	92025	8,9
-2	0	0	2	2	5736	-3,1
0	0	0	2	2	977	-0,5
0	0	0	0	2	-895	0,5
0	1	0	0	0	54	-0,1
-2	1	0	2	2	224	-0,6
0	0	1	2	2	129	-0,1
-2	-1	0	2	2	-95	0,3
0	0	1	0	0	-7	0
0	0	0	2	1	200	0
-2	0	0	2	1	-70	0
0	0	-1	2	2	-53	0
0	0	1	0	1	-33	0
2	0	-1	2	2	26	0
0	0	-1	0	1	32	0
0	0	1	2	1	27	0
0	0	-2	2	1	-24	0
2	0	0	2	2	16	0
0	0	2	2	2	13	0
-2	0	1	2	2	-12	0
0	0	-1	2	1	-10	0
2	0	-1	0	1	-8	0
-2	2	0	2	2	7	0
0	1	0	0	1	9	0
-2	0	1	0	1	7	0

0	-1	0	0	1	6	0
2	0	-1	2	1	5	0
2	0	1	2	2	3	0
0	1	0	2	2	-3	0
0	-1	0	2	2	3	0
2	0	0	2	1	3	0
-2	0	2	2	2	-3	0
-2	0	1	2	1	-3	0
2	0	-2	0	1	3	0
2	0	0	0	1	3	0
-2	-1	0	2	1	3	0
-2	0	0	0	1	3	0
0	0	2	2	1	3	0

LAMPIRAN IV

PERIODIC TERMS FOR THE LONGITUDE OF THE MOON[4]

[4] Jean Meeus, *Astronomical Algorithms,* First Edition, Virginia: Willman-Bell, Inc., 1991, hlm. 309

Periodic Terms for The Longitude of The Moon, the unit is 0.000001 degree

Arguments Multiple of				Σl
D	M	M'	F	Coefficient of the sine of the arguments
0	0	1	0	6288774
2	0	-1	0	1274027
2	0	0	0	658314
0	0	2	0	213618
0	1	0	0	-185116
0	0	0	2	-114332
2	0	-2	0	58793
2	-1	-1	0	57066
2	0	1	0	53322
2	-1	0	0	45758
0	1	-1	0	-40923
1	0	0	0	-34720
0	1	1	0	-30383
2	0	0	-2	15327
0	0	1	2	-12528
0	0	1	-2	10980
4	0	-1	0	10675
0	0	3	0	10034
4	0	-2	0	8548
2	1	-1	0	-7888
2	1	0	0	-6766
1	0	-1	0	-5163
1	1	0	0	4987
2	-1	1	0	4036
2	0	2	0	3994

4	0	0	0	3861
2	0	-3	0	3665
0	1	-2	0	-2689
2	0	-1	2	-2602
2	-1	-2	0	2390
1	0	1	0	-2348
2	-2	0	0	2236
0	1	2	0	-2120
0	2	0	0	-2069
2	-2	-1	0	2048
2	0	1	-2	-1773
2	0	0	2	-1595
4	-1	-1	0	1215
0	0	2	2	-1110
3	0	-1	0	-892
2	1	1	0	-810
4	-1	-2	0	759
0	2	-1	0	-713
2	2	-1	0	-700
2	1	-2	0	691
2	-1	0	-2	596
4	0	1	0	549
0	0	4	0	537
4	-1	0	0	520
1	0	-2	0	-487
2	1	0	-2	-399
0	0	2	-2	-381
1	1	1	0	351
3	0	-2	0	-340

4	0	-3	0	330
2	-1	2	0	327
0	2	1	0	-323
1	1	-1	0	299
2	0	3	0	294

LAMPIRAN V

PERIODIC TERMS FOR THE LATITUDE OF THE MOON[5]

[5] Jean Meeus, *Astronomical Algorithms,* First Edition, Virginia: Willman-Bell, Inc., 1991, hlm. 311

Periodic Terms for The Latitude of The Moon, the unit is 0.000001 degree

Argument Multiple of				Σb
D	M	M'	F	Coefficient of the sine of the arguments
0	0	0	1	5128122
0	0	1	1	280602
0	0	1	-1	277693
2	0	0	-1	173237
2	0	-1	1	55413
2	0	-1	-1	46271
2	0	0	1	32573
0	0	2	1	17198
2	0	1	-1	9266
0	0	2	-1	8822
2	-1	0	-1	8216
2	0	-2	-1	4324
2	0	1	1	4200
2	1	0	-1	-3359
2	-1	-1	1	2463
2	-1	0	1	2211
2	-1	-1	-1	2065
0	1	-1	-1	-1870
4	0	-1	-1	1828
0	1	0	1	-1794
0	0	0	3	-1749
0	1	-1	1	-1565
1	0	0	1	-1491
0	1	1	1	-1475
0	1	1	-1	-1410

0	1	0	-1	-1344
1	0	0	-1	-1335
0	0	3	1	1107
4	0	0	-1	1021
4	0	-1	1	833
0	0	1	-3	777
4	0	-2	1	671
2	0	0	-3	607
2	0	2	-1	596
2	-1	1	-1	491
2	0	-2	1	-451
0	0	3	-1	439
2	0	2	1	422
2	0	-3	-1	421
2	1	-1	1	-366
2	1	0	1	-351
4	0	0	1	331
2	-1	1	1	315
2	-2	0	-1	302
0	0	1	3	-283
2	1	1	-1	-229
1	1	0	-1	223
1	1	0	1	223
0	1	-2	-1	-220
2	1	-1	-1	-220
1	0	1	1	-185
2	-1	-2	-1	181
0	1	2	1	-177
4	0	-2	-1	176

4	-1	-1	-1	166
1	0	1	-1	-164
4	0	1	-1	132
1	0	-1	-1	-119
4	-1	0	-1	115
2	-2	0	1	107

LAMPIRAN VI

PERIODIC TERMS FOR THE DISTANCE OF THE MOON[6]

[6] Jean Meeus, *Astronomical Algorithms,* First Edition, Virginia: Willman-Bell, Inc., 1991, hlm. 310

Periodic Terms for The Distance of The Moon, the unit in 0.001 km

Argumen Multiple of				Σr Coefficient of the Cosine of the argument
D	M	M'	F	
0	0	1	0	-20905355
2	0	-1	0	-3699111
2	0	0	0	-2955968
0	0	2	0	-569925
2	0	-2	0	246158
2	-1	0	0	-204586
2	0	1	0	-170733
2	-1	-1	0	-152138
0	1	-1	0	-129620
1	0	0	0	108743
0	1	1	0	104755
0	0	1	-2	79661
0	1	0	0	48888
4	0	-1	0	-34782
2	1	0	0	30824
2	1	-1	0	24208
0	0	3	0	-23210
4	0	-2	0	-21636
1	1	0	0	-16675
2	0	-3	0	14403
2	-1	1	0	-12831
4	0	0	0	-11650
2	0	2	0	-10445
2	0	0	-2	10321
2	-1	-2	0	10056

2	-2	0	0	-9884
2	0	-1	-2	8752
1	0	-1	0	-8379
0	1	-2	0	-7003
1	0	1	0	6322
0	1	2	0	5751
2	-2	-1	0	-4950
0	0	2	-2	-4421
2	0	1	-2	4130
4	-1	-1	0	-3958
3	0	-1	0	3258
0	0	0	2	-3149
2	1	1	0	2616
2	2	-1	0	2354
0	2	-1	0	-2117
4	-1	-2	0	-1897
1	0	-2	0	-1739
4	-1	0	0	-1571
4	0	1	0	-1423
0	2	1	0	1165
0	0	4	0	-1117

LAMPIRAN VII

CODING PEMROGRAMAN APLIKASI ZEPHEMERIS PADA ECLIPSE JUNO 4.0.

```
package com.example.zephemeris;


import java.math.BigDecimal;
import java.math.RoundingMode;
import java.util.Calendar;
import java.util.Date;

import com.example.zephemeris.R;

import
android.support.v7.app.ActionBar
Activity;
import android.os.Bundle;
import android.view.Menu;
import android.view.MenuItem;
import android.view.Window;
import
android.view.WindowManager;
import android.widget.Button;
import android.widget.EditText;
import android.widget.TextView;

public class DataEphemeris
extends ActionBarActivity {
	EditText tgl, bln, thn,
jam, menit, detik,cbtgl;
	Button htg;
	private Integer zona,
jsek, msek, dtsek;
	String tang, bul, tah, jm,
mt, dt, wak, nama,
jskg,mskg,dtskg,zonwa,tglsk,blsk
,thsk;
	TextView hsl, hslb,
coba,Cbkale;



	@Override
	protected void
onCreate(Bundle
savedInstanceState) {

	super.onCreate(savedInstan
ceState);



	setContentView(R.layout.ac
tivity_data_ephemeris);

		jam =
(EditText)findViewById(R.id.jam)
;
		menit =
(EditText)findViewById(R.id.meni
t);
		detik =
(EditText)findViewById(R.id.deti
k);

		cbtgl =
(EditText)findViewById(R.id.cbka
l);
		//Cbkale =
(TextView)findViewById(R.id.cbka
le);

		Calendar calendar =
Calendar.getInstance();
	calendar.setTime(new
Date());
	zona =
calendar.get(Calendar.ZONE_OFFSE
T);
	jsek =
calendar.get(Calendar.HOUR_OF_DA
Y);
	msek =
calendar.get(Calendar.MINUTE);
	dtsek =
calendar.get(Calendar.SECOND);

		hsl =
(TextView)findViewById(R.id.Ephe
m);
		hslb =
(TextView)findViewById(R.id.Ephe
mb);
		//coba =
(TextView)findViewById(R.id.Coba
);
		//waktu =
(TextView)findViewById(R.id.wkt)
;
		String [] wak =
{"Januari","Februari","Maret","A
pril","Mei","Juni","Juli","Agust
us","September","Oktober","Novem
ber","Desember"};
```

```
            //True Geocentric
Distance
            Double tgd =
((aa9+aa10*(t/10)+aa11*(t/10)*(t
/10)+aa12*(t/10)*(t/10)*(t/10)+a
a13*(t/10)*(t/10)*(t/10)*(t/10))
/100000000);
            Double tgd1 = tgd *
100000000; Integer tgd2 =
tgd1.intValue(); Double tgdm =
tgd2 / 1000000000.0;
            EL9 = ((-
20.4898/3600.0)/tgd);
            EL10 =
(EL8+EL9+nut65)%360.0; //Bujur
Matahari
            Double sd =
((959.63/3600)/tgd);//Semi
diamter
            Double dekl; //
deklinasi
            dekl =
Math.asin(Math.sin((aa14/3600.0)
*Math.PI/180)*Math.cos(eps41*Mat
h.PI/180)+Math.cos((aa14/3600.0)
*Math.PI/180)*Math.sin(eps41*Mat
h.PI/180)*Math.sin(EL10*Math.PI/
180))*180/Math.PI;
            String dkm; if
(dekl <0){dkm ="-";}else{dkm ="
";}
            Double  asen1,
asen2; //asensiorekta
            asen1 =
(Math.atan2(Math.sin(EL10*Math.P
I/180)*Math.cos(eps41*Math.PI/18
0)-
Math.tan((aa14/3600.0)*Math.PI/1
80)*Math.sin((aa14/3600.0)*Math.
PI/180),Math.cos(EL10*Math.PI/18
0)))*180/Math.PI;
            if (asen1 <
0){asen2 = (asen1+360.0);}else
{asen2 = (asen1%360.0);}
            Double epu,
equ;//equation
            aa6 =
(((280.4664567+360007.6982779*(t
/10)-
0.03032028*(t/10)*(t/10)+(t/10)*
(t/10)*(t/10)/49931-
(t/10)*(t/10)*(t/10)*(t/10)/1529
9-
(t/10)*(t/10)*(t/10)*(t/10)*(t/1
0)/1988000)%360.0)*Math.PI/180)%
360.0;
            equ =
(((aa6*180/Math.PI)-0.0057183 -
asen2 +
nut65*Math.cos(eps41*Math.PI/180
))*4)%20.0; if (equ<-16){epu=
equ +20;}else {epu = equ;}
            String equa; if
(epu <0){equa ="-";}else{equa ="
";}
//tampilan hasil
            Integer dBM=
EL10.intValue();
        Double mmBM = (EL10-
dBM)*60.0; Integer mBM =
mmBM.intValue();
        Double ssBM = (mmBM-
mBM)*60.0;
        Integer swBM =
ssBM.intValue();
        Double stsBM = (ssBM -
ssBM.intValue())*100.0;
        Integer stBM =
stsBM.intValue();
        Double sBM = swBM +
stBM/100.0; //ecliptic longitude

        Integer dlmth=
lmth.intValue();
        Double mmlmth = (lmth-
dlmth)*60.0; Integer mlmth =
mmlmth.intValue();
        Double sslmth = (mmlmth-
mlmth)*60.0;     Integer swlmth
= sslmth.intValue();Double
stslmth = (sslmth -
sslmth.intValue())*100.0;
Integer stlmth =
(int)stslmth.intValue();
        Double slmth =
swlmth.doubleValue() +
stlmth/100.0; //ecliptic
latitude
        String lintmat; if (lmth
<0){lintmat ="-";}else{lintmat
=" ";}
        Integer dasen=
asen2.intValue();
        Double mmasen = (asen2-
dasen)*60.0; Integer masen =
mmasen.intValue();
        Double ssasen = (mmasen-
masen)*60.0;     Integer swasen
```

```
= ssasen.intValue();Double
stsasen = (ssasen -
ssasen.intValue())*100.0;
Integer stasen =
(int)stsasen.intValue();
        Double sasen =
swasen.doubleValue() +
(int)(stasen)/100.0; //Asensio
        Integer dDM=
dekl.intValue();
        Double mmDM = (dekl-
dDM)*60.0; Integer mDM =
mmDM.intValue();
        Double ssDM = (mmDM-
mDM)*60.0;     Integer swDM =
ssDM.intValue();Double stsDM =
(ssDM - ssDM.intValue())*100.0;
Integer stDM = (int)
stsDM.intValue();
        Double sDM =
swDM.doubleValue() +
(int)(stDM)/100.0; //Declination
        Integer dSD=
sd.intValue();
        Double mmSD = (sd-
dSD)*60.0; Integer mSD =
mmSD.intValue();
        Double ssSD = (mmSD-
mSD)*60.0;     Integer swSD =
ssSD.intValue();Double stsSD =
(ssSD - ssSD.intValue())*100.0;
Integer stSD =
(int)stsSD.intValue();
        Double sSD =
swSD.doubleValue() +
(int)(stSD)/100.0; //Semi
diameter
        Integer dTO=
eps41.intValue();
        Double mmTO = (eps41-
dTO)*60.0; Integer mTO =
mmTO.intValue();
        Double ssTO = (mmTO-
mTO)*60.0;     Integer swTO =
ssTO.intValue();Double stsTO =
(ssTO - ssTO.intValue())*100.0;
Integer stTO =
(int)stsTO.intValue();
        Double sTO =
swTO.doubleValue() +
(int)(stTO)/100.0; //Obliquity
        Integer mEoT=
epu.intValue();
        Double ssEoT = (epu-
mEoT)*60.0; Integer dEoT =
mEoT.intValue();
        Double stEoT = (ssEoT -
dEoT)*100; Integer sdEoT = (int)
stEoT.intValue();
        Double dtEOT = dEoT +
(int)(sdEoT)/100.0;

        hsl.setText("Ecliptic
Longitude "+dBM+"° "+mBM+"'
"+round(sBM,2)+"''\n" +
                    "Ecliptic
Latitude "+lintmat+"0° "+"0'
"+Math.abs(lmth)+"''\n" +
                    "Apparent
Right Ascension "+dasen+"°
"+masen+"'
"+round(sasen,2)+"''\n" +
                    "Apparent
Declination
"+dkm+Math.abs(dDM)+"°
"+Math.abs(mDM)+"'
"+round(Math.abs(sDM),2)+"''\n"
+
                    "True
Geocentric Distance "+tgdm+" AU
\n"+
                    "Semi
Diameter "+dSD+"° "+mSD+"'
"+round(sSD,2)+"''\n" +
                    "True
Obliquity "+dTO+"° "+mTO+"'
"+round(sTO,2)+"''\n" +
                    "Equation of
Time "+equa+Math.abs(mEoT)+" m
"+round(Math.abs(dtEOT),2)+"
s");

      //mean longitude bln
        aa1 =(((218.3164591 +
481267.88134236*t -
0.0013268*t*t + t*t*t/538841 -
t*t*t*t/65194000))%360.0)*Math.P
I/180.0;
        //elongasi rata2 bln
        aa2 = (((297.8502042 +
445267.1115168*t - 0.00163*t*t +
t*t*t/545868 -
t*t*t*t/113065000))%
360.0)*Math.PI/180.0;
        //anomali rata2 mthari
        aa3 = (((357.5291092 +
35999.0502909*t - 0.0001536*t*t
```

```
+ t*t*t/24490000))%
360.0)*Math.PI/180.0;
        //anomali rata2 bln
        double aa4=
(((134.9634114 +
477198.8676313*t + 0.008997*t*t
+ t*t*t/69699 -
t*t*t*t/14712000))%360.0)*Math.P
I/180.0;
        //argumen bjur bln
        aa5 = (((93.2720993 +
483202.0175273*t - 0.0034029*t*t
- t*t*t/3526000 +
t*t*t*t/863310000))
%360.0)*Math.PI/180.0;
        //eksentrisitas orbit
bln
        aa6 =  1 - 0.002516*t -
0.0000074*t*t;
        //argumen 1
        aa7 = (((119.75 +
131.849*t))
%360.0)*Math.PI/180.0;
        //argumen 2
        aa8 = (((53.09 +
479264.29*t))%
360.0)*Math.PI/180.0;
        //argumen 3
        aa9 = (((313.45 +
481266.484*t))%360.0)*Math.PI/18
0.0;
         //apparent longitude
bln
        Double jrkbb =
385000.56+koki2; //TGD bln
        Double tgdb1 = jrkbb -
jrkbb.intValue(); Double tgdb2 =
tgdb1 *10000; Integer tgdb3 =
tgdb2.intValue();
        Double tgdb =
jrkbb.intValue() + tgdb3 /
100000.0;
        Double parb =
(Math.asin(6378.14/jrkbb))*180.0
/Math.PI; //Horizontal parallaks
        Double sdb =
358473400/(jrkbb*3600); //semi
diameter
        //asensio bln
        double asen1b =
(Math.atan2(Math.sin(bjrb*Math.P
I/180.0)*Math.cos(eps41*Math.PI/
180.0)-
Math.tan(koki1*Math.PI/180.0)*Ma
th.sin(eps41*Math.PI/180.0),Math
.cos(bjrb*Math.PI/180.0)))*180.0
/Math.PI;
        Double asenb;if
(asen1b<=0){asenb =
asen1b+360.0;}else{asenb =
asen1b%360.0;}
        //deklinasi Bln
        Double dekb =
(Math.asin(Math.sin(koki1*Math.P
I/180.0)*Math.cos(eps41*Math.PI/
180.0)+Math.cos(koki1*Math.PI/18
0.0)*Math.sin(eps41*Math.PI/180.
0)*Math.sin(bjrb*Math.PI/180.0))
)*180.0/Math.PI;
        String deklib; if (dekb
<0){deklib ="-";}else{deklib ="
";}
        //tampilan
        Integer dBB=
bjrb.intValue();
        Double mmBB = (bjrb-
dBB)*60.0; Integer mBB =
mmBB.intValue();
        Double ssBB = (mmBB-
mBB)*60.0;
        Integer swBB =
ssBB.intValue();
        Double stsBB = (ssBB -
ssBB.intValue())*100.0;
        Integer stBB =
(int)stsBB.intValue();
        Double sBB = swBB +
(int)(stBB)/100.0; //Apparent
longitude
        Integer dLB=
koki1.intValue();
        Double mmLB = (koki1-
dLB)*60.0; Integer mLB =
mmLB.intValue();
        Double ssLB = (mmLB-
mLB)*60.0;
        Integer swLB =
ssLB.intValue();
        Double stsLB = (ssLB -
ssLB.intValue())*100.0;
        Integer stLB =
(int)stsLB.intValue();
        Double sLB = swLB +
(int)(stLB)/100.0; //Apparent
latitude
        Integer dAB=
asenb.intValue();
```

```
        Double mmAB = (asenb-
dAB)*60.0; Integer mAB =
mmAB.intValue();
        Double ssAB = (mmAB-
mAB)*60.0;
        Integer swAB =
ssAB.intValue();
        Double stsAB = (ssAB -
ssAB.intValue())*100.0;
        Integer stAB =
(int)stsAB.intValue();
        Double sAB = swAB +
(int)(stAB)/100.0; //Apparent
Right Ascension
        Integer dDeB=
dekb.intValue();
        Double mmDeB = (dekb-
dDeB)*60.0; Integer mDeB =
mmDeB.intValue();
        Double ssDeB = (mmDeB-
mDeB)*60.0;
        Integer swDeB =
ssDeB.intValue();
        Double stsDeB = (ssDeB -
ssDeB.intValue())*100.0;
        Integer stDeB =
(int)stsDeB.intValue();
        Double sDeB = swDeB +
(int)(stDeB)/100.0; //Apparent
Declination
        Integer dDB=
sdb.intValue();
        Double mmDB = (sdb-
dDB)*60.0; Integer mDB =
mmDB.intValue();
        Double ssDB = (mmDB-
mDB)*60.0;
        Integer swDB =
ssDB.intValue();
        Double stsDB = (ssDB -
ssDB.intValue())*100.0;
        Integer stDB =
(int)stsDB.intValue();
        Double sDB = swDB +
(int)(stDB)/100.0; //Moon Semi
Diameter
        Integer dHP=
parb.intValue();
        Double mmHP = (parb-
dHP)*60.0; Integer mHP =
mmHP.intValue();
        Double ssHP = (mmHP-
mHP)*60.0;
        Integer swHP =
ssHP.intValue();
        Double stsHP = (ssHP -
ssHP.intValue())*100.0;
        Integer stHP =
(int)stsHP.intValue();
        Double sHP = swHP +
(int)(stHP)/100.0; //Moon
Horizontal Parallax
        hslb.setText("Apparent
Longitude "+dBB+"° "+mBB+"'
"+round(sBB,2)+"''\n" +
                    "Apparent
Latitude "+alb+Math.abs(dLB)+"°
"+Math.abs(mLB)+"'
"+round(Math.abs(sLB),2)+"''\n"
+
                    "Apparent
Right Ascension "+dAB+"°
"+mAB+"' "+round(sAB,2)+"''\n" +
                    "Apparent
Declination
"+deklib+Math.abs(dDeB)+"°
"+Math.abs(mDeB)+"'
"+round(Math.abs(sDeB),2)+"''\n"
+
                    "True
Geocentric Distance "+tgdb+" Km
\n"+
                    "Horizontal
Parallax "+dHP+"° "+mHP+"'
"+round(sHP,2)+"''\n" +
                    "Semi
Diameter "+dDB+"° "+mDB+"'
"+round(sDB,2)+"''\n"+"\n"
                    +"Hasil
Perhitungan Pada "+ta+"
"+wak[bl-1]+" "+tah+" Pukul
"+jm+":"+mt+":"+dt+" WIB ");
        //waktu.setText("Hasil
Perhitungan Pada "+ta+"
"+wak[bl-1]+" "+tah+" Pukul
"+jm+":"+mt+":"+dt+" WIB ");
      }
      double round(double
number, int decimalPlaces) {
          if(decimalPlaces == 0)
{ // to remove all values after
the dot
              return (int)
number;
          }
          int format = (int)
Math.pow(10.0, decimalPlaces);
```

```
            return (double)
Math.round(format * number)  /
format;
        }
        @Override
        public boolean
onCreateOptionsMenu(Menu menu) {
                // Inflate the
menu; this adds items to the
action bar if it is present.

        getMenuInflater().inflate(
R.menu.data_ephemeris, menu);
                return true;
        }

        @Override
        public boolean
onOptionsItemSelected(MenuItem
item) {
                // Handle action
bar item clicks here. The action
bar will
                // automatically
handle clicks on the Home/Up
button, so long
                // as you specify a
parent activity in
AndroidManifest.xml.
                int id =
item.getItemId();
                if (id ==
R.id.action_settings) {
                        return true;
                }
                return
super.onOptionsItemSelected(item
);
        }
}
```

DAFTAR RIWAYAT PENDIDIKAN PENULIS

Nama : Zul Amri Fathinul Inshafi

Tempat/Tanggal Lahir : Singkawang, 03 Februari 1995

Alamat Asal : Jl. Pramuka No.10, Kelurahan Condong Dalam, Kecamatan Singkawang Tengah, Kota Singkawang, Kalimantan Barat.

Alamat Sekarang : Jl. Wahyu Asri Dalam I No. 15B, RT 05/ RW 06, Kecamatan Ngaliyan, Kelurahan Tambak Aji, Semarang Barat, Kota Semarang, Jawa Tengah.

Riwayat Pendidikan :

- TK Aisyiah (1999-2000)
- SDN 17 Singkawang Tengah (2000-2006)
- MTsN Darussalam (2006-2009)
- MA Unggulan Darul ‘Ulum (2009-2012)
- Strata 1 UIN Walisongo (2012-2016)

Pengalaman Organisasi:

- Pengurus DepKomInfo CSS MoRA (Community of Santri Scholars of Ministry of Religious Affair) UIN Walisongo tahun 2013 – 2014
- Pegiat di forum intelektual Farabi Institute (2013-sekarang)

Demikian riwayat hidup ini saya buat dengan sebenar-benarnya untuk menjadi maklum adanya.

Semarang. 20 November 2016

Zul Amri Fathinul Inshafi
NIM. 122111133