A C++ source program for calculating the Goldbach-Xu's numbers with recursive method

Wan-Dong Xu Heng Xu

School of Science, Tianjin University, Tianjin 300072, China

E-mail : **xwandong@sohu.com**

Abstract: In this paper we have presented two source programs with C++ to calculate the distribution of prime numbers in the sequence of odd numbers and the Goldbach-Xu's numbers for every even number respectively. Form the result we can know that this number is oscillatingly increased as an even number increases.

Keywords: Goldbach's conjecture; prime; distribution of primes; Goldbach-Xu's number; source program.

MSC: 11P32; 11A41; 11N05; 11N35

1.Introduction

Chinese famous mathematician Luo-Geng Hua had presented in 1979 a route to analyzingly deduce the solution numbers of Goldbach's problem for every even number, $r(N)$, through solving a indefinite equation, $ax+by=N$, here N is an even, $a>0$, $b>0$, and $(a, b)=1$ [1]. He was failed. After that time, some researchers devoted themselves in this route, and were also not succeeded [2]. Recently, we have advanced a recursive method to calculate the number of rest sum formulae of two distinct odd prime numbers, which are to express every even number >6 in natural sequence [3]. Now we will list some programs for calculating them. By the results of calculating for them, we can know that the number of rest sum formulae, or say, Goldbach-Xu's number, is oscillatingly increased as an even number increases such that the Goldbach's conjecture is verified to be true.

2. Calculating the distribution of primes in the sequence of odd numbers

We have written a program, named prime.cpp, with C++ language to calculate the distribution of odd prime numbers in the sequence of odd numbers in Appendix A. There is a input number in that program is "nupto", it means to calculate the primes up to "nupto", and there is a output file named prim0101.tex led on disk G:, in which an odd prime number is denoted by symbol "1" and an odd composite number by symbol "0", and in which there are 100 figures in every row to indicate 100 odd numbers in order. This is completely the same as ref. [4].

In the Table 1 we listed a distribution table for some prime numbers up to 12,600.

3. Calculating the Goldbach-Xu's numbers

There is a program for calculating with a recursive method, named Goxu1.cpp in Appendix B, with C++ language, to calculate the numbers of rest sum formulae of two distinct odd prime numbers, or say, the Goldbach-Xu's numbers, for every even number >6. There is an input file named prim0101.tex on disk G:, which is the output file in the section 2 above. And there is an output file named Goxun1.tex led on disk G:, which could be opened by the written-board in Windows XP. In that program many of variant names are the same as symbols in ref. [3].

In the Table 2 we listed some Goldbach-Xu's numbers for every even numbers starting at 8 and ending up to 3366. And we can know that the Goldbach-Xu's number is oscillatingly increased as an even number increases such that the Goldbach's conjecture can be verified to be true.

There are two source programs in the author's hands, which could be sent readers if they need them and connect to the author.

Table1. The irregular distribution of primes in the sequence of odd numbers

Table 2. The oscillatingly increasing characteristic of the Goldbach-Xu's numbers

n	$Lr(n)$																												
1~30	1	1	1	1	2	2	2	2	3	2	2	3	2	3	4	1	3	4	3	3	5	4	3	5	3	3	6	2	5	6
31~60	2	5	6	4	5	7	4	4	8	4	4	9	4	4	7	3	6	8	5	5	8	6	7	10	6	5	12	3	5	10
61~90	3	7	9	5	5	8	7	7	11	5	5	12	4	8	11	4	8	10	5	5	13	9	6	11	7	6	14	6	8	13
91~120	5	8	11	6	9	13	8	8	14	6	7	19	6	7	13	6	9	11	7	6	12	9	7	15	9	9	18	8	9	16
121~150	6	9	16	8	8	14	10	8	16	8	9	19	7	10	16	6	14	16	8	12	17	10	8	19	8	10	21	8	10	15
151~180	8	12	17	8	10	15	11	11	20	6	10	24	6	10	19	9	13	17	10	8	16	13	10	20	9	9	22	7	14	18
181~210	8	14	18	10	11	22	13	9	19	11	9	27	11	10	21	6	14	17	11	13	20	13	11	21	10	11	30	10	12	21
211~240	9	14	19	13	11	21	14	13	21	11	13	27	12	11	24	8	16	28	12	12	24	15	13	23	14	10	29	10	14	23
241~270	9	19	22	13	13	23	13	14	27	15	14	32	11	13	23	11	17	24	11	14	25	14	17	22	13	13	30	9	13	30
271~300	11	19	23	10	11	23	18	13	24	12	13	31	11	16	26	12	19	25	12	12	29	16	15	27	12	15	32	12	14	27
301~330	13	20	26	14	19	26	18	16	31	11	16	41	10	13	28	15	18	25	17	16	27	21	15	29	13	19	41	13	16	31
331~360	11	21	33	14	17	28	21	16	30	16	16	39	11	18	30	13	24	31	18	18	24	16	17	37	14	14	39	14	15	31
361~390	15	21	31	14	19	29	18	19	31	17	19	39	14	17	35	14	21	30	17	16	31	26	18	32	16	14	44	14	18	30
391~420	15	22	34	16	14	38	21	15	32	16	14	39	18	20	34	16	20	29	16	21	34	22	22	33	18	16	51	17	17	32
421~450	15	25	31	20	19	39	18	16	33	16	21	46	18	19	36	13	25	39	21	17	37	23	19	34	20	18	48	15	17	34
451~480	15	31	31	19	18	35	23	19	47	17	18	43	17	19	36	18	24	34	18	20	33	25	23	37	19	21	45	16	18	45
481~510	17	27	32	16	19	35	26	16	39	20	23	52	13	25	37	16	28	36	18	17	42	25	23	39	18	19	51	18	22	42
511~540	18	25	36	21	27	40	26	21	39	18	19	57	18	24	44	19	27	37	24	24	39	25	21	40	20	27	54	19	21	39
541~570	18	26	48	22	18	40	28	24	44	25	25	54	16	22	41	22	34	47	19	22	39	26	22	49	23	19	58	17	24	38
571~600	26	27	36	18	22	42	29	25	43	24	22	58	18	21	49	19	26	40	20	19	43	33	23	45	24	23	54	18	28	43
601~630	20	32	42	21	21	49	27	25	45	21	22	55	28	24	42	17	34	44	23	26	45	28	23	51	20	21	68	21	26	42
631~660	21	27	40	26	25	42	27	25	46	21	29	60	23	25	49	20	33	53	24	23	46	30	23	46	27	25	66	20	26	53
661~690	22	41	47	25	25	45	27	27	54	24	22	60	22	20	41	24	33	44	27	26	48	28	27	47	23	27	61	19	24	59
691~720	20	30	44	24	24	45	34	26	48	23	25	58	18	30	47	19	34	41	22	23	57	35	25	50	22	22	60	27	28	45
721~750	20	36	49	25	34	48	33	29	47	24	24	73	22	30	51	25	34	52	29	24	52	31	27	50	28	32	67	26	27	51
751~780	25	33	59	22	26	56	31	26	48	28	27	69	22	32	47	25	46	46	21	29	51	31	28	62	24	25	72	29	28	51
781~810	25	38	57	22	26	47	31	34	58	28	26	71	25	26	64	24	36	53	23	30	47	42	35	53	27	31	65	23	35	55
811~840	29	40	60	27	27	67	35	26	52	25	28	76	28	30	55	26	39	54	28	34	53	37	35	56	28	30	83	24	31	56
841~870	26	37	57	32	28	52	34	30	55	29	32	78	26	26	68	20	38	64	29	30	53	36	25	55	35	33	76	25	33	55
871~900	25	48	52	26	30	55	41	28	69	31	31	73	28	26	53	33	37	59	25	30	52	36	36	66	31	27	75	31	31	72
901~930	28	36	53	27	27	53	46	26	58	33	29	76	25	34	62	27	37	54	32	28	70	38	28	54	32	28	76	34	32	53
931~960	28	43	58	29	36	61	38	32	61	30	28	91	35	31	63	33	36	55	36	29	58	36	30	66	28	35	81	30	34	58
961~990	30	39	68	27	33	65	41	29	58	34	31	83	29	34	53	27	48	56	26	28	62	38	32	71	33	32	82	29	32	58
991~1020	30	42	59	26	28	56	37	44	59	35	28	84	27	35	73	27	41	59	32	31	62	49	29	64	33	32	85	29	38	75
1021~1050	25	42	65	34	32	75	45	32	58	33	35	82	36	33	58	26	48	59	30	39	62	46	34	62	32	31	97	30	35	71
1051~1080	32	41	67	34	37	63	37	32	67	30	42	82	33	35	66	28	47	81	32	36	63	43	32	64	43	37	79	34	38	67
1081~1110	31	52	65	30	35	68	41	33	85	33	31	82	33	33	62	34	46	60	32	34	60	47	38	68	34	28	90	35	33	78
1111~1140	28	41	67	31	36	58	51	35	77	32	33	86	26	44	70	31	44	75	31	38	75	41	33	61	32	36	92	38	33	68
1141~1170	37	47	68	32	40	63	49	31	68	33	34	114	35	40	66	38	48	67	38	34	64	43	35	68	35	38	94	33	36	76
1171~1200	31	45	83	41	37	62	45	32	66	44	37	92	33	37	76	32	60	67	35	37	67	44	41	84	37	37	90	36	36	72
1201~1230	39	50	69	37	35	78	48	41	72	31	35	94	36	37	86	31	50	75	37	39	73	56	36	71	30	38	93	34	52	69
1231~1260	32	57	71	38	37	87	46	39	76	38	40	94	42	41	75	32	47	68	32	46	74	45	34	72	38	36	112	35	41	68
1261~1290	34	55	70	41	36	70	43	38	77	36	48	99	43	40	71	30	48	82	34	36	64	44	39	86	43	35	95	35	43	73
1291~1320	33	60	68	34	45	71	49	35	91	34	36	98	39	36	72	45	48	77	38	35	69	46	45	76	34	33	107	28	40	81
1321~1350	31	54	80	34	41	70	58	44	72	37	42	97	28	48	77	36	46	74	42	41	88	48	38	75	38	42	100	44	43	87
1351~1380	37	52	73	39	52	78	51	41	79	39	47	128	35	41	82	37	51	79	50	41	75	56	38	86	43	46	109	37	37	72
1381~1410	36	55	101	40	39	68	55	43	81	44	44	104	37	45	78	38	64	72	34	44	90	51	45	97	40	38	104	43	41	76
1411~1440	41	52	78	40	42	89	49	49	78	40	40	110	40	41	100	37	68	78	42	40	71	64	41	76	42	39	107	37	50	90
1441~1470	40	60	74	44	42	94	51	46	91	40	44	99	48	44	83	38	57	80	41	60	80	52	42	82	31	45	122	36	41	87
1471~1500	41	59	75	44	42	82	52	37	98	34	52	116	36	48	86	33	59	95	41	43	79	58	49	84	45	45	104	40	41	78
1501~1530	36	69	76	44	50	85	49	41	99	43	41	110	36	48	92	43	62	93	43	43	83	50	53	78	39	47	113	37	44	102
1531~1560	41	58	79	41	44	83	68	47	83	36	41	109	35	59	84	47	61	92	42	45	100	48	46	82	45	40	117	44	46	85
1561~1590	47	57	83	36	49	80	47	43	75	46	44	138	39	45	85	41	56	85	45	43	93	52	47	83	42	52	112	40	42	87
1591~1620	40	66	98	42	51	90	54	39	83	45	48	110	44	41	87	36	73	80	46	42	86	58	48	112	40	45	112	37	39	82
1621~1650	47	59	87	43	51	76	55	52	89	44	46	110	38	46	112	44	59	86	43	47	78	67	45	90	39	45	120	41	56	93
1651~1680	44	56	92	43	48	107	57	49	84	47	46	116	54	43	89	41	57	84	47	52	89	57	45	94	41	55	138	43	47	105
1681~1710	42	62	86	52	44	89	60	45	87	39	58	115	45	47	90	44	62	105	49	52	85	67	43	87	50	49	119	43	40	84
1711~1740	43	74	101	46	50	89	59	47	102	41	46	120	43	47	89	57	64	87	46	41	97	56	57	91	47	46	123	44	47	105
1741~1770	43	59	84	47	54	103	64	54	88	39	46	123	40	51	87	43	71	92	47	50	103	58	44	96	52	49	123	62	46	94
1771~1800	41	60	93	42	51	91	62	59	104	46	50	154	48	49	96	46	62	80	55	54	108	60	45	88	47	58	125	47	49	90
1801~1830	50	66	112	52	48	95	59	47	94	58	47	136	40	50	95	45	83	92	49	51	103	58	47	115	49	47	125	45	46	102
1831~1860	56	58	95	54	50	97	64	60	98	50	46	124	50	46	125	42	66	97	48	56	100	78	49	93	43	56	128	49	62	98
1861~1890	48	61	101	47	46	116	77	43	107	45	47	123	56	41	97	48	67	112	50	58	91	73	54	103	49	47	149	51	52	96
1891~1920	50	63	88	55	50	100	70	45	92	55	64	130	48	48	95	44	68	124	48	51	109	67	46	100	58	57	128	49	50	98
1921~1950	51	87	93	52	50	98	59	46	122	50	50	133	52	53	110	54	65	97	53	53	92	65	62	109	53	49	140	42	52	117
1951~1980	48	75	98	53	55	93	82	48	98	53	55	136	48	57	103	54	68	99	52	51	121	66	61	100	51	52	145	55	52	107
1981~2010	52	72	95	51	63	111	62	53	94	53	49	163	49	51	99	46	65	106	72	51	104	65	54	108	55	66	133	42	55	114
2011~2040	44	80	118	53	48	99	70	51	100	63	59	125	44	54	109	43	81	97	51	54	98	74	51	122	49	50	142	57	50	108
2041~2070	58	65	113	51	53	102	66	61	104	52	55	139	47	61	122	55	71	96	53	49	107	82	54	113	55	55	147	52	64	100
2071~2100	52	70	107	54	53	138	69	56	104	54	56	137	62	46	108	51	81	106	50	72	104	64	54	102	49	52	165	54	50	105
2101~2130	45	66	107	62	54	116	68	55	112	52	63	137	51	55	105	57	72	122	52	59	106	75	52	105	64	52	139	50	63	105
2131~2160	55	85	115	54	61	115	71	58	131	55	56	165	56	48	108	62	80	102	56	57	100	68	70	106	60	58	140	48	58	124
2161~2190	50	72	110	60	53	109	85	61	112	52	55	143	55	65	118	49	70	102	53	60	136	76	55	102	59	60	140	65	50	117
2191~2220	48	73	106	55	69	106	81	58	113	50	62	171	45	48	111	49	86	113	69	53	108	70	53	105	53	70	138	49	60	125
2221~2250	46	77	127	61	52	105	73	59	109	73	57	147	49	56	108	47	88	107	60	60	127	71	53	132	54	64	138	51	58	101
2251~2280	61	90	115	55	57	101	72	77	126	57	58	149	51	56	131	53	72	107	55	54	111	90	57	131	55	58	147	51	70	111
2281~2310	54	72	110	49	68	129	77	52	109	53	61	158	61	56	110	59	71	112	55	67	117	72	53	114	54	56	190	53	56	110
2311~2340	59	77	107	68	58	107	69	65	113	59	69	151	53	59	114	59	73	136	68	55	113	74	57	122	70	56	165	53	62	127
2341~2370	50	95	119	55	58	114	75	56	131	61	65	148	54	54	112	60	81	109	56	59	113	78	75	119	58	57	154	57	60	139
2371~2400	54	83	121	59	61	122	87	58	114	53	57	147	55	79	116	55	73	108	61	58	143	78	60	127	66	57	150	64	61	114
2401~2430	51	85	108	60	75	128	79	61	120	58	64	189	59	62	121	58	82	114	69	57	105	74	62	114	53	68	154	70	57	114
2431~2460	54	73	139	52	64	109	74	62	126	69	62	149	51	59	122	53	94	123	58	65	103	75	61	150	54	56	152	60	55	117
2461~2490	70	84	118	61	60	105	89	73	121	56	61	167	54	63	145	55	81	112	61	70	119	92	68	115	55	61	159	58	70	117
2491~2520	65	84	126	60	63	144	76	56	121	62	56	154	68	67	145	63	75	119	56	79	121	82	64	116	63	70	184	58	66	115
2521~2550	54	80	117	76	64	122	87	60	123	61	77	176	57	58	118	50	83	159	64	60	118	82	66	120	77	67	167	55	72	130
2551~2580	55	101	127	64	61	122	76	65	148	64	53	167	61	65	121	69	84	123	59	61	140	76	73	121	59	59	167	59	59	144
2581~2610	65	90	110	69	62	121	89	68	126	60	59	159	67	71	118	64	90	130	58	61	144	75	63	132	63	59	160	73	61	131
2611~2640	56	85	115	62	86	119	84	63	139	57	64	198	68	62	124	61	77	116	70	58	118	85	62	120	57	84	175	60	67	115
2641~2670	53	87	142	62	63	112	82	72	140	77	64	163	55	61	128	57	107	122	69	63	126	86	64	138	69	69	166	64	63	138
2671~2700	70	83	114	58	73	122	82	77	119	60	70	166	60	63	143	56	81	132	59	67	124	102	68	132	64	67	156	63	75	128
2701~2730	63	86	141	57	61	139	75	66	126	62	65	169	70	87	123	53	93	121	63	75	127	85	68	135	74	63	214	54	67	129
2731~2760	52	82	127	84	63	144	80	64	118	65	80	164	61	67	126	59	98	149	62	67	127	88	70	130	71	73	172	66	65	134
2761~2790	54	103	128	64	65	138	79	70	169	72	70	173	63	66	129	70	89	126	68	75	127	81	79	133	69	70	177	59	66	163
2791~2820	65	93	124	66	63	124	96	63	132	63	67	198	64	79	136	56	90	120	64	67	151	83	74	125	67	63	169	79	73	130
2821~2850	60	86	125	73	81	140	86	70	131	64	74	204	63	66	147	64	89	128	78	63	127	84	65	145	63	89	186	61	72	132
2851~2880	64	87	168	65	61	127	110	65	130	77	69	174	58	73	127	61	103	143	71	79	132	84	76	162	71	68	172	64	76	136
2881~2910	73	87	144	60	63	128	87	78	132	75	65	167	61	63	166	70	98	130	68	62	141	97	67	147	63	64	171	69	87	129
2911~2940	70	101	135	64	67	150	87	70	131	76	74	191	91	67	131	66	89	128	62	80	131	87	73	157	74	68	216	67	74	140
2941~2970	74	98	127	79	74	132	89	76	134	70	87	182	64	73	153	71	96	161	66	64	146	87	70	139	79	72	202	66	73	133
2971~3000	55	107	133	73	75	138	81	77	167	75	69	177	63	73	129	75	99	132	78	72	126	87	81	140	63	71	178	61	73	195

Note: Every even number > 6 in the table is $D(n)=2(n+3)$.

Table 2. The oscillatingly increasing characteristic of the Goldbach-Xu's numbers (continued)

n	$Lr(n)$																													
3001~3030	61	92	137	69	69	146	111	65	144	68	75	173	74	84	126	64	91	151	61	64	161	99	72	140	71	77	177	76	68	132
3031~3060	59	98	147	67	84	138	93	69	153	72	67	222	64	79	140	60	100	132	84	68	142	99	68	132	73	95	192	63	72	141
3061~3090	61	91	162	70	76	152	85	69	145	81	75	190	60	76	142	65	123	154	77	70	138	81	69	163	69	70	180	72	73	132
3091~3120	84	91	136	69	66	119	87	86	161	75	68	195	61	83	163	71	90	148	74	79	144	109	77	135	77	73	198	65	89	140
3121~3150	75	95	146	68	77	159	98	73	147	72	73	218	78	70	138	69	101	138	68	80	145	108	82	143	69	77	216	69	82	137
3151~3180	70	101	138	96	69	150	86	77	156	64	93	185	70	74	150	60	98	161	76	81	142	95	72	140	80	88	194	70	80	144
3181~3210	61	130	140	72	72	138	102	77	190	72	75	191	77	78	164	80	98	156	77	68	140	96	92	146	71	70	184	72	76	187
3211~3240	76	94	138	73	74	150	111	66	148	74	79	183	69	91	138	72	107	145	66	76	188	95	69	163	72	72	193	82	72	149
3241~3270	72	109	138	76	94	151	105	77	140	63	67	241	74	74	148	65	94	145	84	79	152	97	82	166	77	83	190	68	75	141
3271~3300	69	101	184	71	81	137	99	78	137	85	70	202	70	78	149	89	117	142	75	73	145	91	75	177	73	75	211	69	89	138
3301~3330	85	97	149	78	70	142	91	105	149	72	71	218	66	72	176	68	92	142	86	68	138	114	70	151	80	77	194	68	89	165
3331~3360	68	108	147	76	75	177	99	81	148	76	83	187	83	68	153	75	104	145	77	89	169	105	78	149	72	78	239	70	71	151
3361~3390	76	95	175	101	72	144	94	81	144	78	91	188	73	77	151	71	112	172	81	78	147	95	71	141	98	73	191	70	80	159
3391~3420	70	122	151	81	73	162	100	81	183	79	82	195	79	77	146	88	107	152	76	75	136	93	95	166	73	77	205	78	80	170
3421~3450	75	99	146	77	74	144	119	77	178	72	81	192	77	97	154	80	103	162	77	83	178	117	83	151	68	76	206	100	76	148
3451~3480	76	99	146	76	101	154	94	77	151	66	82	268	73	72	162	69	103	160	102	79	147	100	90	162	76	94	205	73	77	157
3481~3510	77	115	186	92	73	145	96	78	151	95	78	206	78	82	161	71	119	150	83	84	155	96	83	180	76	87	231	71	77	157
3511~3540	84	121	158	78	75	168	104	93	153	75	78	210	75	74	186	71	103	171	77	74	168	113	89	162	81	78	198	70	108	157
3541~3570	75	108	152	78	75	195	105	83	159	93	82	204	91	81	150	66	100	154	79	94	161	107	84	166	71	89	258	71	82	154
3571~3600	79	123	147	87	76	145	99	81	162	75	94	200	82	87	175	77	106	193	75	73	160	107	85	163	93	76	198	78	75	147
3601~3630	77	126	152	77	95	150	105	80	189	78	89	206	74	82	157	98	108	167	76	79	150	104	95	176	76	86	229	72	80	183
3631~3660	80	103	152	77	85	153	135	88	152	76	77	203	67	102	160	81	112	164	86	83	188	112	89	172	85	79	212	82	82	174
3661~3690	76	101	162	85	99	158	104	80	158	79	92	254	76	82	157	77	116	157	95	82	155	106	87	155	74	106	217	74	81	156
3691~3720	79	108	205	80	79	147	104	83	151	98	79	236	76	86	154	76	131	150	78	80	155	99	81	191	98	81	210	74	73	167
3721~3750	98	98	157	78	77	176	107	110	157	76	74	214	74	81	183	77	127	163	81	86	167	116	77	152	79	86	206	84	98	160
3751~3780	68	103	155	91	74	191	107	82	194	84	79	209	81	80	158	72	126	157	84	106	176	104	84	161	74	75	251	82	87	182
3781~3810	88	107	157	93	83	161	103	97	168	76	98	250	90	84	151	74	117	190	76	85	161	105	99	159	105	94	218	79	83	170
3811~3840	77	132	159	84	82	168	102	82	213	80	86	224	81	81	189	97	105	163	80	87	161	114	102	166	87	86	211	83	91	204
3841~3870	78	110	165	76	99	158	142	80	169	82	89	223	72	107	159	86	115	192	73	78	197	116	88	169	80	86	217	87	94	162
3871~3900	87	119	183	77	99	168	110	85	166	86	86	267	80	95	161	81	118	164	101	85	189	119	85	163	88	98	233	80	85	160
3901~3930	79	126	199	83	84	164	118	83	169	108	90	225	89	83	165	76	134	158	84	89	162	105	94	233	88	81	210	70	83	183
3931~3960	96	112	171	96	95	178	106	104	159	78	94	220	77	85	208	87	113	159	94	84	163	121	92	166	86	86	243	78	106	167
3961~3990	72	133	164	79	93	197	108	100	165	87	86	216	98	93	200	85	113	170	89	99	174	114	83	167	76	85	274	85	83	185
3991~4020	77	122	169	99	88	170	106	79	178	76	121	219	80	89	171	75	112	200	94	84	157	125	83	186	98	78	224	69	80	169
4021~4050	72	140	182	81	92	175	110	89	198	91	83	212	82	101	170	98	111	167	86	91	167	112	110	172	96	89	230	75	85	200
4051~4080	81	116	173	81	87	184	138	91	176	91	90	227	87	108	172	82	134	185	78	88	208	113	90	157	83	86	238	104	93	163
4081~4110	83	125	157	91	106	182	106	91	194	78	93	292	76	97	163	83	123	160	102	95	180	104	91	169	91	100	225	85	88	167
4111~4140	96	112	193	86	95	171	111	97	171	105	81	251	88	81	177	80	144	175	86	93	190	100	101	207	94	92	238	83	103	174
4141~4170	98	114	165	106	98	166	119	107	180	80	89	219	84	84	228	81	119	188	78	96	172	145	82	183	85	101	226	88	109	195
4171~4200	80	117	174	82	95	202	134	90	182	92	88	228	109	83	172	76	117	184	86	100	163	118	90	169	86	108	271	79	97	177
4201~4230	85	132	160	102	86	182	111	87	186	97	107	228	91	85	188	86	114	214	82	93	195	130	85	171	102	83	240	79	89	176
4231~4260	92	158	173	94	97	182	111	86	207	89	89	232	93	98	166	105	124	195	90	82	167	114	108	195	89	83	231	77	86	215
4261~4290	96	125	185	89	105	169	139	84	175	91	87	247	75	114	187	100	126	171	81	95	226	113	80	180	86	86	272	94	90	185
4291~4320	89	116	172	88	104	166	116	109	185	94	85	282	88	92	180	93	117	166	108	95	170	106	93	176	92	108	239	95	90	198
4321~4350	84	117	214	86	87	191	118	88	182	107	95	251	86	91	174	82	142	171	94	91	173	125	100	229	89	89	242	96	97	181
4351~4380	106	132	195	87	92	168	115	112	176	94	88	239	88	101	238	99	136	182	93	95	173	138	86	180	104	95	244	88	111	175
4381~4410	90	114	195	91	96	244	115	90	182	95	98	236	105	91	177	91	125	179	92	125	181	124	91	195	95	86	281	102	90	181
4411~4440	87	126	190	93	91	175	141	92	204	82	103	244	91	102	188	90	116	211	93	118	177	121	93	194	102	93	256	88	93	182
4441~4470	95	148	199	90	94	180	117	87	218	93	95	262	96	92	184	114	125	178	93	94	182	120	125	182	95	92	246	87	101	226
4471~4500	90	129	178	104	95	173	141	89	191	95	96	273	84	104	220	89	117	178	88	90	214	127	94	191	96	101	242	110	95	202
4501~4530	90	133	185	95	119	183	136	99	184	88	93	303	89	92	191	91	126	203	122	95	210	120	102	177	90	115	247	98	105	182
4531~4560	92	128	209	101	90	193	124	96	175	123	92	241	86	92	186	89	164	194	100	98	211	123	104	227	96	89	259	97	87	199
4561~4590	104	136	184	93	90	190	119	110	182	96	90	245	109	96	224	96	121	188	101	97	176	150	95	211	100	102	252	86	116	189
4591~4620	97	122	183	92	108	215	138	97	215	87	96	249	118	99	197	102	120	184	87	112	179	133	96	189	89	94	329	95	95	204
4621~4650	98	129	181	116	107	186	124	103	188	96	109	252	93	96	179	90	121	254	104	94	186	116	97	183	118	92	261	85	87	209
4651~4680	107	155	189	90	105	189	121	97	233	106	107	249	95	100	189	123	129	182	94	93	203	148	113	194	93	94	264	95	90	229
4681~4710	89	120	215	96	92	182	152	98	214	108	107	243	92	123	203	100	138	183	85	94	223	125	99	199	105	96	255	115	111	188
4711~4740	94	126	185	104	114	225	130	97	193	95	97	298	97	108	195	101	142	199	120	90	186	128	102	197	99	113	257	106	93	213
4741~4770	88	145	224	94	93	187	135	99	209	117	97	253	97	105	211	85	170	199	101	109	188	123	94	232	97	104	257	104	103	202
4771~4800	120	130	199	104	102	193	123	121	191	93	110	295	97	97	245	89	132	194	93	93	199	159	109	221	96	96	261	76	127	197
4801~4830	91	137	190	117	93	234	135	100	194	92	106	264	112	96	212	91	126	191	93	124	202	123	105	192	99	114	324	100	110	194
4831~4860	97	140	220	119	98	191	140	101	197	96	118	284	93	101	193	106	121	254	98	104	184	134	99	192	117	102	254	95	128	195
4861~4890	103	161	193	103	101	196	133	100	235	113	93	286	108	98	194	124	135	185	98	90	219	132	117	191	108	98	260	103	99	230
4891~4920	98	151	205	102	110	200	147	118	205	94	101	258	101	118	206	93	135	206	98	100	259	124	98	219	96	90	264	129	104	196
4921~4950	100	130	195	103	130	202	144	103	204	85	99	316	102	103	208	110	156	200	117	100	196	146	103	214	96	118	301	97	102	211
4951~4980	94	134	233	99	109	223	141	112	200	122	108	266	105	102	202	102	162	200	113	112	196	126	95	248	98	105	269	113	99	217
4981~5010	120	139	194	92	104	195	136	135	211	102	110	269	102	98	255	98	127	197	99	91	192	191	99	209	104	98	263	92	121	194
5011~5040	106	139	238	104	109	235	132	96	203	104	102	256	113	109	190	99	131	218	112	119	198	137	92	198	110	95	315	99	101	208
5041~5070	109	136	200	120	102	241	131	99	201	109	124	254	97	116	194	95	157	244	106	100	204	124	101	194	114	109	292	111	101	214
5071~5100	96	162	212	106	101	204	123	97	264	124	112	271	102	111	194	122	138	199	107	118	200	130	136	196	106	104	294	97	101	234
5101~5130	111	132	216	99	100	213	161	118	206	102	96	304	93	135	203	101	141	203	113	104	247	136	117	200	103	104	277	113	102	216
5131~5160	98	156	199	109	130	215	129	103	200	98	101	330	110	107	245	107	135	218	130	114	214	133	109	202	102	138	277	108	117	209
5161~5190	98	141	252	96	113	194	152	112	209	116	114	281	103	106	206	93	170	231	98	98	204	141	111	291	97	102	269	114	121	212
5191~5220	119	139	197	105	111	196	142	125	215	118	103	279	104	100	248	97	138	201	98	112	219	166	112	218	109	110	279	98	127	205
5221~5250	107	164	230	99	99	245	129	111	206	102	106	274	142	107	217	113	135	194	92	134	226	137	115	237	99	105	323	102	122	216
5251~5280	107	139	207	119	122	210	134	108	210	110	126	302	104	116	205	110	153	255	92	97	199	135	104	205	135	105	304	99	111	211
5281~5310	96	172	212	108	105	208	141	130	249	100	98	263	103	109	214	121	145	225	107	108	241	137	119	218	96	102	280	95	102	290
5311~5340	102	144	192	108	106	200	169	113	216	102	116	280	99	129	216	105	155	212	106	104	254	155	113	204	111	118	271	121	106	216
5341~5370	97	138	228	102	128	203	138	112	212	105	101	340	114	115	229	106	149	200	120	116	210	148	113	205	119	139	281	105	117	208
5371~5400	99	141	251	107	102	239	145	106	232	125	112	285	96	109	198	102	192	205	111	112	217	142	112	256	100	101	287	117	112	201
5401~5430	123	156	237	101	112	214	133	126	231	110	102	289	95	110	262	109	145	231	101	130	205	173	113	205	105	101	277	103	127	211
5431~5460	135	140	217	106	105	255	145	110	208	98	105	311	131	115	210	103	141	231	112	143	210	139	123	232	105	111	362	111	108	216
5461~5490	109	138	205	139	108	208	137	108	225	118	144	288	106	106	230	96	140	266	112	106	210	133	123	226	125	118	282	115	104	219
5491~5520	104	170	200	108	111	242	153	102	240	107	112	287	106	106	231	120	151	231	115	109	203	139	135	210	109	113	301	103	118	251
5521~5550	99	162	209	102	110	222	172	108	217	121	103	291	106	135	241	105	136	220	111	118	275	141	120	224	113	108	291	134	110	213
5551~5580	104	159	212	97	137	229	147	112	218	105	120	350	121	122	220	105	147	209	122	103	216	145	120	263	99	123	291	105	106	216
5581~5610	108	132	268	107	125	226	157	106	209	138	109	279	102	104	215	115	179	215	108	120	213	155	108	257	106	106	340	109	109	218
5611~5640	129	148	234	108	105	218	143	144	226	101	119	286	102	117	266	118	142	218	124	120	214	174	113	216	112	100	290	107	152	246
5641~5670	120	146	212	105	113	252	139	111	218	109	121	321	129	108	227	104	146	238	115	127	212	163	110	216	119	112	348	106	113	231
5671~5700	108	148	247	130	115	220	142	137	217	108	133	292	109	125	227	113	141	256	112	110	233	155	111	206	152	112	301	107	109	222
5701~5730	108	174	221	115	107	237	137	108	272	115	109	289	113	113	215	133	173	214	112	122	213	140	138	234	100	112	290	110	104	279
5731~5760	106	148	219	115	122	220	184	111	256	115	109	288	122	131	229	112	157	223	105	126	262	148	110	225	107	116	284	133	120	242
5761~5790	117	147	224	115	133	214	136	117	239	110	111	393	111	117	223	109	155	225	136	114	218	161	123	211	109	131	301	107	108	218
5791~5820	103	158	277	130	122	217	140	112	218	127	115	300	103	103	237	113	182	240	117	117	234	147	112	264	113	123	293	104	113	218
5821~5850	140	158	213	109	115	226	158	143	220	111	110	285	98	116	263	107	150	247	117	115	222	164	116	222	122	107	316	104	153	236
5851~5880	112	147	221	104	118	277	151	121	215	128	110	319	125	114	222	106	148	235	110	135	254	152	131	220	111	114	350	111	122	230
5881~5910	105	160	223	135	113	250	163	113	228	114	129	295	124	119	216	119	154	258	121	108	231	138	113	246	133	117	292	113	109	224
5911~5940	109	191	239	117	137	215	148	116	261	102	114	294	115	111	264	142	148	221	107	116	241	145	138	218	111	113	327	121	108	270
5941~5970	112	168	225	121	115	218	201	122	226	108	115	291	104	153	238	116	147	228	123	116	262	153	122	253	107	119	302	131	116	251
5971~6000	117	159	233	108	144	229	170	115	213	113	133	379	114	112	223	107	155	225	135	123	232	166	113	236	108	144	302	115	112	246

Note: Every even number > 6 in the table is $D(n)=2(n+3)$.

Table 2. The oscillatingly increasing characteristic of the Goldbach-Xu's numbers (continued)

n	$Lr(n)$																														
6001~6030	116	152	328	114	111	221	145	114	227	133	116	297	109	123	242	120	184	227	109	121	216	148	122	280	128	114	310	111	129	229
6031~6060	129	167	220	113	115	255	153	137	243	116	113	341	101	110	278	118	172	215	118	111	231	178	111	227	129	117	302	129	134	250
6061~6090	120	157	224	109	125	292	142	129	268	112	119	301	144	123	225	115	170	224	117	165	244	155	128	231	113	122	376	111	114	229
6091~6120	122	163	223	141	119	237	152	123	226	125	139	348	114	122	230	114	180	278	113	120	229	146	135	222	150	118	324	107	121	242
6121~6150	106	179	223	128	121	220	148	114	281	115	114	308	122	135	250	123	166	242	115	118	226	155	140	235	118	145	313	105	115	274
6151~6180	122	164	248	117	113	229	202	121	251	105	111	300	109	140	237	115	163	268	115	123	278	167	123	247	115	123	306	135	135	231
6181~6210	104	158	221	122	170	235	148	122	234	133	124	363	112	115	230	112	169	255	135	116	264	163	115	219	119	136	308	107	117	247
6211~6240	124	176	283	123	121	240	151	119	246	141	112	312	134	129	232	111	192	240	124	129	232	158	114	302	121	123	344	121	116	232
6241~6270	132	158	236	118	139	233	151	159	229	119	128	317	122	116	291	132	174	246	118	124	242	176	134	235	116	115	368	115	147	259
6271~6300	118	157	234	120	125	320	153	134	239	118	118	311	136	121	240	112	175	229	144	162	236	149	124	238	118	121	365	111	125	279
6301~6330	120	164	239	158	128	229	163	117	244	115	164	308	116	117	257	111	165	296	125	118	248	186	124	254	145	114	322	125	123	248
6331~6360	116	197	255	115	116	229	159	128	300	114	132	328	124	140	253	142	158	251	112	114	238	158	147	254	140	123	334	104	114	302
6361~6390	127	173	238	118	115	261	199	128	241	125	127	346	109	138	228	120	184	232	119	140	302	153	130	240	126	117	311	152	134	233
6391~6420	129	153	254	113	140	241	159	130	268	124	118	392	107	124	230	145	157	223	144	143	232	152	128	258	122	148	322	115	143	234
6421~6450	127	156	305	125	118	241	160	124	244	142	121	386	125	120	249	118	205	251	119	131	234	158	127	285	144	119	325	117	127	243
6451~6480	130	165	241	135	122	237	170	162	244	129	122	328	119	118	316	115	160	237	118	127	256	188	126	262	126	151	316	116	142	243
6481~6510	113	167	259	137	126	298	177	131	246	126	134	324	149	126	267	117	177	259	128	151	249	156	124	242	120	124	394	125	141	269
6511~6540	112	156	244	153	124	259	167	132	236	138	152	344	131	127	258	111	168	286	126	129	266	172	135	232	141	130	331	119	121	247
6541~6570	118	237	248	118	122	251	158	122	313	117	127	359	137	126	247	140	165	245	128	122	235	179	155	261	121	124	330	115	128	288
6571~6600	126	168	250	113	157	255	204	127	251	126	125	322	129	149	257	133	172	264	121	125	292	173	138	245	123	124	349	152	120	266
6601~6630	124	174	245	113	151	229	161	141	271	132	121	391	110	141	241	120	168	246	163	131	255	163	127	249	125	149	377	131	124	275
6631~6660	119	168	296	125	123	248	166	126	253	164	132	321	116	136	252	127	218	236	122	119	231	171	139	296	118	126	347	118	127	263
6661~6690	156	172	270	129	124	274	179	154	255	125	128	340	119	133	312	119	165	262	132	127	242	193	122	242	147	127	332	119	153	264
6691~6720	129	188	257	121	139	335	165	126	256	126	125	334	139	135	270	123	179	261	124	149	240	180	139	246	121	116	376	155	122	242
6721~6750	130	164	258	147	125	254	161	129	295	123	168	321	121	130	255	120	173	299	122	138	238	172	125	269	144	135	336	124	133	247
6751~6780	132	207	235	128	129	242	169	132	318	128	136	370	128	127	258	139	162	263	127	141	255	164	166	261	128	128	333	108	125	335
6781~6810	129	181	281	132	120	255	187	122	242	133	128	330	116	152	288	128	182	249	132	123	297	169	129	240	139	142	335	145	142	248
6811~6840	123	175	257	133	153	244	182	124	251	123	122	433	109	124	254	122	166	279	150	126	261	156	128	250	143	147	366	113	144	263
6841~6870	124	176	298	133	134	256	166	156	251	169	140	344	130	129	253	124	196	237	137	124	310	160	129	313	132	130	338	126	124	252
6871~6900	150	184	246	143	151	249	158	165	271	123	128	363	135	129	312	120	187	255	119	134	251	206	127	302	131	127	357	123	168	283
6901~6930	122	170	253	121	140	305	169	138	253	127	140	348	172	132	256	157	170	240	126	162	250	165	130	253	121	136	446	129	130	258
6931~6960	130	187	268	150	129	248	163	141	272	137	161	343	115	131	264	124	162	292	138	146	268	182	141	252	155	132	371	121	131	280
6961~6990	117	208	255	116	156	270	181	132	315	136	145	356	122	133	263	156	174	282	122	132	256	190	159	289	126	139	333	120	136	311
6991~7020	142	173	281	118	131	248	196	132	246	131	130	333	134	189	266	131	180	277	127	131	312	176	126	261	147	131	366	155	126	267
7021~7050	117	178	235	128	152	282	181	133	251	140	136	408	122	137	288	120	193	253	148	126	264	164	141	271	135	169	350	143	143	258
7051~7080	123	186	306	128	131	277	165	135	283	157	126	339	121	134	283	116	217	249	145	146	260	167	136	308	136	135	356	128	129	259
7081~7110	169	190	261	132	140	270	163	154	259	134	130	379	121	141	335	121	178	256	137	132	248	207	168	278	122	129	336	132	165	257
7111~7140	127	175	254	140	131	316	168	128	258	142	151	355	144	129	286	122	188	256	131	155	267	175	131	273	122	149	433	122	132	260
7141~7170	140	166	257	149	126	251	202	125	255	136	164	346	119	137	259	120	165	336	135	157	255	168	134	266	152	138	344	132	149	262
7171~7200	126	206	293	129	126	247	171	138	325	135	133	342	132	131	254	161	177	279	144	130	285	172	161	258	135	138	344	129	137	311
7201~7230	132	200	256	129	147	263	203	137	261	130	134	382	142	155	266	117	188	280	141	130	310	183	131	295	139	134	334	154	130	266
7231~7260	125	174	264	132	177	277	175	145	282	122	133	425	122	132	257	143	181	253	158	141	299	162	135	270	134	168	395	142	134	262
7261~7290	128	174	317	143	147	261	165	152	266	162	125	352	139	153	261	128	231	253	151	140	260	182	126	315	131	128	347	139	133	343
7291~7320	153	179	265	131	136	262	169	161	259	140	146	340	141	125	332	141	196	254	119	141	276	240	134	270	129	143	345	134	161	267
7321~7350	120	186	309	129	134	317	180	134	306	132	141	347	150	153	268	122	182	253	125	161	286	193	135	262	136	133	420	124	135	287
7351~7380	127	194	256	162	150	297	179	142	257	141	169	347	127	138	267	123	205	348	144	142	256	170	131	268	181	136	360	148	135	271
7381~7410	141	212	265	127	141	266	178	141	349	139	136	402	136	151	269	162	186	281	127	157	250	170	172	268	137	144	399	120	143	317
7411~7440	151	184	258	133	144	251	213	138	279	146	145	408	122	166	268	151	184	263	130	133	319	176	167	276	124	146	357	160	149	272
7441~7470	138	180	294	165	180	293	173	132	285	132	143	439	126	137	305	128	174	266	180	142	257	168	138	295	136	187	356	137	138	279
7471~7500	124	213	325	127	139	264	201	130	286	159	137	353	134	133	298	127	221	299	132	147	266	179	143	340	145	132	355	141	155	290
7501~7530	164	196	266	130	145	265	174	179	265	142	161	366	123	131	323	128	181	276	129	136	306	216	141	295	130	143	361	141	172	279
7531~7560	132	194	269	130	145	311	190	146	277	141	160	358	170	132	305	129	176	261	142	184	258	177	140	310	128	142	431	125	159	276
7561~7590	137	195	291	168	160	280	174	145	275	129	164	359	127	136	268	157	188	351	144	145	269	182	134	273	167	133	424	123	156	266
7591~7620	132	238	277	137	137	296	186	154	325	139	143	383	133	138	278	149	181	280	147	142	277	179	174	274	144	148	362	127	142	371
7621~7650	128	188	285	139	145	276	216	152	274	139	149	366	140	171	299	141	189	266	146	132	361	200	140	284	135	143	395	161	140	274
7651~7680	133	190	309	161	172	286	173	154	279	136	135	447	129	172	277	137	210	265	169	153	279	184	149	264	148	169	368	131	150	315
7681~7710	135	196	331	140	144	304	185	138	274	164	140	396	148	145	272	133	244	297	134	145	280	200	136	341	137	156	362	154	145	278
7711~7740	167	195	283	139	146	283	181	157	333	130	139	367	144	134	354	149	191	263	131	162	262	250	143	289	134	145	374	135	175	287
7741~7770	153	181	287	133	160	330	192	149	309	134	142	417	164	142	283	140	191	295	138	179	267	188	167	270	129	142	466	134	147	277
7771~7800	149	198	277	183	140	271	181	146	278	143	174	374	145	153	317	139	206	337	134	148	272	177	137	290	166	163	405	141	148	297
7801~7830	130	230	283	141	142	281	202	147	350	143	145	380	139	147	281	155	199	321	134	143	286	184	189	278	154	136	384	145	157	342
7831~7860	137	192	265	149	148	304	207	136	282	165	154	384	136	178	288	143	190	289	151	143	398	183	138	284	139	151	374	166	148	279
7861~7890	132	233	301	143	175	293	185	148	289	149	145	421	153	157	321	136	196	287	165	141	295	197	149	313	148	182	385	145	145	283
7891~7920	125	199	340	144	158	284	186	147	289	166	168	410	144	148	273	145	242	284	141	152	287	180	148	384	143	142	417	142	156	305
7921~7950	166	191	278	137	144	293	199	195	280	139	139	396	136	147	339	149	192	283	159	152	281	231	153	292	141	141	375	129	171	305
7951~7980	138	200	329	142	156	348	194	152	275	137	161	373	165	148	296	150	202	292	147	188	293	219	135	288	141	142	477	148	162	290
7981~8010	146	194	306	168	149	275	202	145	301	133	167	418	135	165	303	149	191	334	143	149	306	190	146	311	193	151	384	136	138	283
8011~8040	145	232	289	147	158	319	182	160	340	146	153	377	143	154	276	175	216	280	152	145	303	192	174	322	141	151	379	167	145	335
8041~8070	137	186	282	150	141	278	228	146	331	150	153	383	147	179	310	146	221	290	144	159	344	186	151	289	140	152	376	157	160	337
8071~8100	163	219	290	150	173	286	188	144	287	148	148	517	157	155	287	132	199	306	183	151	306	177	170	288	154	190	384	137	150	290
8101~8130	138	189	343	158	148	302	188	132	311	176	146	379	134	150	326	139	253	280	142	151	290	198	155	357	147	165	381	142	153	300
8131~8160	165	212	286	135	158	288	214	169	316	155	145	384	137	140	351	145	191	351	148	154	291	235	151	288	146	147	395	138	203	276
8161~8190	147	216	275	145	154	349	207	148	284	165	152	386	164	173	300	139	202	286	141	182	330	184	145	294	155	156	507	140	151	285
8191~8220	148	223	284	175	144	286	191	150	303	155	163	386	153	164	305	143	201	377	143	144	301	198	151	318	181	150	385	134	154	288
8221~8250	134	244	286	156	181	332	182	143	332	150	157	394	151	150	282	186	195	296	157	155	296	204	197	296	150	146	431	147	144	356
8251~8280	138	222	287	150	153	289	231	162	313	142	141	419	141	172	333	132	199	289	161	149	351	198	148	309	138	165	407	180	156	324
8281~8310	142	196	275	139	185	293	194	140	299	135	189	459	161	150	291	141	198	284	163	153	285	221	142	322	149	179	397	131	158	323
8311~8340	141	193	381	138	148	296	213	153	305	181	147	396	149	175	294	132	246	285	148	162	291	190	145	344	163	157	385	155	158	305
8341~8370	173	196	319	149	147	341	189	184	300	149	147	401	149	149	352	155	238	279	145	147	319	236	159	304	146	144	397	150	210	294
8371~8400	153	195	296	150	153	373	193	162	334	150	155	428	166	147	286	140	203	280	140	203	293	190	148	302	177	160	469	152	148	297
8401~8430	152	196	286	171	148	293	189	158	306	141	176	460	149	161	316	146	193	353	145	149	314	191	162	300	182	140	394	132	166	296
8431~8460	141	242	313	174	146	305	191	156	353	147	155	392	143	153	328	181	219	293	136	158	297	207	182	318	144	161	393	143	150	397
8461~8490	143	208	316	149	161	284	264	161	307	148	164	391	151	186	288	151	206	329	145	157	349	185	149	326	147	168	393	177	180	311
8491~8520	139	199	302	152	176	294	215	148	319	153	144	477	134	160	298	148	214	301	189	152	318	210	150	314	153	185	408	141	152	292
8521~8550	140	237	361	161	168	303	207	158	298	186	165	386	162	158	291	143	248	322	142	149	292	191	151	413	150	149	415	153	155	292
8551~8580	196	215	325	154	167	300	200	179	304	152	152	404	140	169	372	180	206	281	146	146	284	231	150	296	148	157	480	137	186	304
8581~8610	153	221	301	150	168	348	194	169	302	162	159	394	174	157	309	141	209	295	189	181	305	195	161	321	148	151	497	149	150	346
8611~8640	140	196	284	177	151	348	191	152	314	143	200	426	151	163	313	140	200	361	166	155	303	216	166	305	170	152	391	150	160	308
8641~8670	142	276	317	146	162	302	199	157	360	162	149	395	143	171	333	170	194	287	161	154	322	197	187	303	171	158	428	153	155	361
8671~8700	146	204	298	145	156	326	241	166	302	166	174	404	153	199	299	148	229	299	159	154	368	218	162	323	145	155	406	195	166	315
8701~8730	151	207	301	139	182	302	230	156	350	146	153	490	142	165	297	150	200	316	179	191	313	201	143	306	161	191	415	156	156	304
8731~8760	168	202	392	148	159	299	213	152	318	194	152	457	142	160	310	154	247	304	154	157	300	218	164	386	172	147	403	151	174	321
8761~8790	181	202	303	169	157	307	203	190	335	145	162	445	143	141	433	156	206	295	160	147	305	238	162	319	165	186	399	154	198	300
8791~8820	147	200	301	160	158	355	218	165	312	152	166	404	198	165	309	161	201	345	160	187	347	199	165	305	143	166	489	157	174	332
8821~8850	155	204	295	203	154	299	196	151	321	178	188	452	161	159	312	145	231	357	153	165	337	219	158	309	175	154	417	148	151	334
8851~8880	152	277	311	164	166	293	199	152	358	154	156	411	184	162	312	185	214	298	145	161	343	205	179	346	154	173	422	156	152	375
8881~8910	152	214	305	142	162	308	249	168	346	147	156	414	152	199	307	169	201	327	148	161	379	219	163	296	162	152	453	195	155	303
8911~8940	146	202	312	161	198	291	195	177	311	155	166	530	154	170	322	157	219	323	211	148	297	188	146	311	157	181	413	143	163	335
8941~8970	174	194	365	168	153	322	206	143	317	176	178	415	151	188	322	163	256	317	156	153	312	226	149	376	163	163	467	154	149	310
8971~9000	186	207	360	154	150	321	209	185	321	176	163	409	153	193	381	165	231	320	154	172	321	241	171	313	170	156	419	152	186	315

Note: Every even number > 6 in the table is $D(n)=2(n+3)$.

Appendix A

```
//This is a program to calculate the prime numbers and
//the distribution of primes

#include <iostream.h>
#include <math.h>
#include <fstream.h>
#include <iomanip.h>
#include <stdlib.h>
#define AA 60001

void main()
{
  int i,w,r,prime=0,nodd[AA],jj,n=0;
  int nupto,nprimes=0,odd=1,nnupto;
  float ii,fl_odd;

  cout << "input max loop number (input nupto)<=60000? ";
  cin >> nupto;                                          //input max loop number.

  for (i=1; i<=nupto; i++) nodd[i]=0;
  nnupto=nupto-2;

    do
  {
    odd=odd+2;
    w=0;
    ii=2.0;
    fl_odd=(float)odd;
    jj=int(sqrt(fl_odd));
loop_10:    r=int(fmod(fl_odd,ii));
        if (r==0) w=1;
        else ii=ii+(float)1.0;
        if (ii>jj || w!=0)
           if (w==0)
           { prime=odd;
              cout << setw(4) << prime << " ";
              nodd[prime]=1;
              nprimes=nprimes+1;}
           else;
        else goto loop_10;
  }
    while (odd<=nnupto);

              cout << endl;
      for (i=1; i<=nnupto+1; i=i+2)
      {if (i>=1) cout << nodd[i];
            n=n+1;
            if (n%50==0) cout << endl;}
      n=0;
      cout << endl;
      cout << nprimes << endl;
```

```
    fstream out1;                           //output file is G:\prim0101.txt
    out1.open("G:\\prim0101.txt",ios::out);
    for (i=1; i<=nnupto+1; i=i+2)
        out1 << nodd[i];
    out1.close();

    int iii=0;
    fstream out2;                           //output file is G:\prim0102.txt
    out2.open("G:\\prim0102.txt",ios::out);
    for (i=1; i<=nnupto+1; i=i+2)
    {   out2 << nodd[i];
        iii=iii+1;
        if (iii%100==0) out2 << endl;}

    out2.close();
    return;
}
```

Appendix B

```
//This program is to calculate the Goldbach-xu's numbers
//for every even number starting at 8!

#include <iostream.h>
#include <math.h>
#include <fstream.h>
#include <iomanip.h>
#include <stdlib.h>
#include <stdio.h>

#define MM 9001

static int J[MM],J0[MM],Jtable[MM],Lid[MM];
static int Lr[MM],Lt[MM],Lhd[MM],Jlt[MM];
static double fi,rr;
static int i,ii,r;
static int nn;



void part_0 (int ii)                    //part 0
{                                 //This to input the data of the distribution
    int inodd[MM];          //of primes starting at 3.

    ifstream input1;                                //input file
    input1.open("G:\\prim0101.txt",ios::in);
    if (!input1)
    {
        cout << "can't open file";
        exit(0);
    }
```

```
    for (i=0; i<=ii; i++)                    //input and turn 0101 data
          inodd[i]=input1.get();             //into ASCII code 48 and 49.
    input1.close();

    for (i=1; i<=ii; i++)                    //return ASCII code 48 and
    {
          if (inodd[i]==48)                  //49    into 0101 data.
                Jtable[i]=0;
          else
          if (inodd[i]==49)
                Jtable[i]=1;
          else
          break;
    }
     for (i=1; i<=ii; i++)
     {
           cout << Jtable[i];
           if (i%50==0)
                cout << endl;
     }
      cout << endl;
      return;
}


void part_a (int ii)                         //part A
{                                            //this is to calculate Ltr(n).
      for (i=1; i<=ii; i++)                  //DO 10
      {
            fi=(double)i;
            r=int(fmod(fi,2.0));
            if (r!=0)
                Lt[i]=(i+1)/2;
            else
                Lt[i]=i/2;}                  //end DO 10

            for (i=1; i<=ii; i++)
            {
                 cout << setw(5) << Lt[i];
                 if (i%30==0)
                      cout << endl;
            }
            cout << "         above is Lt(n)";
            cout << endl<< endl;
            return;
}


static int cn0,cn01,dn,w;
static int w1,bn;

void part_b (int ii)                                 //part B
{                                                    //next is to calculate the LHD(n).
      Lhd[1]=0; Lhd[2]=0;
```

```
        i=3;
        do                                          //DO 20
        {


            fi=(double)i;
            r=int(fmod(fi,2.0));
            if (r==0)                               //When n is an even number!
            {
                cn0=Lt[i-1]+1;
                cn01=Jtable[cn0];
                w=2*Lt[i]+1;
                dn=Jtable[w];
                if (cn01==0)
                    if (dn==0)
                        Lhd[i]=Lhd[i-1];
                     else
                        Lhd[i]=Lhd[i-1]-1;
                else
                    if (dn==0)
                        Lhd[i]=Lhd[i-1]+1;
                    else
                        Lhd[i]=Lhd[i-1];
           }
           else                                     //When n is an odd number!
           {
                w=2*Lt[i];
                dn=Jtable[w];
                if (dn==0)
                    Lhd[i]=Lhd[i-1]+1;
                else
                    Lhd[i]=Lhd[i-1];
           }
            i++;
        }                                           // end DO 20
        while (i<=ii);

        for (i=1; i<=ii; i++)
        {
            cout << setw(5) << Lhd[i];
            if (i%30==0)
                cout << endl;
        }
                cout << "       above is LHD(n)" <<endl<<endl;
        return;
}


void part_c (int ii)                                //part C
{                                                   //next is to calculate the LID(n).
        for (i=1; i<=ii; i++)
            Lid[i]=0;
        i=7;
        do                                          //DO 30
```

```
        {
            fi=(double)i;
            r=int(fmod(fi,2));
            w1=Lt[i];
            bn=Jtable[w1];
            if (r==0)
                Lid[i]=Lid[i-1];
            else
            {
                if (bn==0)
                    Lid[i]=Lid[i-1]+1;
                else
                    Lid[i]=Lid[i-1];
            }
            i++;
        }
        while (i<=ii);                          //end DO30

        for (i=1; i<=ii; i++)
        {
            cout << setw(5) << Lid[i];
            if (i%30==0)
                cout << endl;
        }
        cout << "          above is LID(n)" <<endl<<endl;
        return;
}

                //*********************************************

static int i1,ll;
static int Jj,Jj1,Jj2,Jjt1,Jjt2;
static int Jjta,Jl1,Jl2,Lt1,Lt2;
static int Llj1,Llj2,Jlta,Llt1,Llt2;

void part_d (int ii)                                    //part D
{                                               //next is to calculate the J(n)~JLT(n).
        for (i=1; i<=ii; i++)                           //do 35.
            Jlt[i]=0;                                   //end do 35.
        for (i=9; i<=ii; i++)                       //do 40.
        {                                               //a for {
            fi=(double)i;
            rr=fmod(fi,2.0);
            ll=Lt[i]-1;
            for (i1=1; i1<=3; i1++)                 //do 45.
                J[i1]=0;
            if (rr==0.0)                                //When n is an even number!
            {
                J0[i]=0;                            //b if {
                for     (i1=4; i1<=ll; i1++)
                                                    //do 80.
                {                                       //c for {
                    Jj=2*Lt[i]+1-i1+1;
                    Jl1=Jtable[i1];
```

```
                Jl2=Jtable[Jj];
                if (Jl1==0 && Jl2==0)
                    J[i1]=1;                    //J[i1]:J(I1)
                else
                    J[i1]=0;
                J0[i]=J0[i]+J[i1];
            };                                  //end do 80.      end c for }
                                //i1=ll has finished!
            Llt1=Lt[i-1];                       //Llt1:LLt1, Lt[i]:LT(n).
            Llt2=Lt[i-1]+2;
            Llj1=Jtable[Llt1];                  //Llj1:LLJ1,
            Llj2=Jtable[Llt2];
            if (Llj1==0 && Llj2==0)
                Jlta=1;                         //Jlta:JLTA,
            else
                Jlta=0;
            Jlt[i]=J0[i]+Jlta;                  //end if (r==0)
          }                                     //end b if }
        else                                    //When n is an odd number!
        {
            for (i1=4; i1<=ll; i1++)            //d else {
                                                //do 90.
            {                                   //e for {
              Jj=2*Lt[i]-i1+1;
              Jl1=Jtable[i1];
              Jl2=Jtable[Jj];
              if (Jl1==0 && Jl2==0)
                  J[i1]=1;
              else
                  J[i1]=0;
                  J0[i]=J0[i]+J[i1];
            };                                  //end do 90.      end e for }
                                                //i1=ll has finished!
            Llt1=Lt[i-1]+1;
            Llt2=Lt[i-1]+2;
            Llj1=Jtable[Llt1];
            Llj2=Jtable[Llt2];
            if (Llj1==0 && Llj2==0)
                Jlta=1;
            else
                Jlta=0;
                Jlt[i]=J0[i]+Jlta;
        }                                       //d end if else }
    };                                          //end do 40.      end for }
    for (i=1; i<=ii; i++)                       //verify JLT(n).
    {
        cout << setw(5) << Jlt[i];
        if (i%30==0)
            cout << endl;
    };
    cout <<endl;
    cout << "       above is Jlt(n)=JLT(n)" << endl << endl;
    return;
}
```

```
                //***********************************************

void main ()
{
    int mm;

    cout <<"please input the cycle number=nn <=9000==? ";
    cin >> nn;

    part_0 (nn);
    part_a (nn);
    part_b (nn);
    part_c (nn);
    part_d (nn);

    for (i=1; i<=nn; i++)                    //This to calculate the Goldbach-
    {                                                //Xu's number.
        Lr[i]=Lt[i]-Lhd[i]-Lid[i]+Jlt[i];
        cout << setw(5) << Lr[i];
        if (i>=1)
        {
            if (i%30==0)
            cout << endl;
        };
    }
    cout << endl;
    cout << "above data are Lr(n)";

    cout << endl;

    fstream output2;                                    //output data file
    output2.open("G:\\Goxun9.txt",ios::out);
    if (!output2)
    {
        cout << "can't open file";
        exit(0);
    }
    int iii;
    iii=0;
    mm=0;

    output2 <<1 <<"~"<<30<<" ";

    for (i=1; i<=nn; i++)
    {
        if (i>=1)
        output2 << setw(4) << Lr[i];
        iii=iii+1;
        if (iii%30==0 && iii<nn)
        {
            output2 << endl;
            mm=mm+1;
            output2 << 30*mm+1 << "~" << 30*mm+30<<" ";
        };
```

```
        };

    output2.close();
    return;
}
```

Reference

[1] Luo-Geng Hua, a lecture in Cambridge University, English, 1979 (unpublished).
[2] Ji-Sheng Na, "An estimation on a sum formula", KEXUETONGBAO, **9**(1986) 641-647.
[3] Wan-Dong Xu, "A new two-dimension sieve method and the proof of the Goldbach's conjecture" (www.paper.edu.cn/0606266) (unpublished).
[4] Wan-Dong Xu, "The irregular distribution of primes up to 300,000 in the sequence of odd numbers" (www.paper.edu.cn/0607179) (unpublished).