International Electronic Journal of Pure and Applied Mathematics
Volume 16 No. 1 2022, 47-53
ISSN: 1314-0744
url: https://www.e.ijpam.eu
doi: 10.12732/iejpam.v16i1.5

NEW REFINED ENHANCED HYBRID ALGORITHM FOR GREATEST COMMON DIVISOR

Anton Iliev[1,2], Nikolay Kyurkchiev[1,2]
Asen Rahnev[1] and Vesselin Kyurkchiev[1]

[1]Faculty of Mathematics and Informatics
University of Plovdiv Paisii Hilendarski
24, Tzar Asen Str., 4000 Plovdiv, BULGARIA

[2]Institute of Mathematics and Informatics
Bulgarian Academy of Sciences
Acad. G. Bonchev Str., Bl. 8, 1113 Sofia, BULGARIA

Abstract: In this note we extend the algorithm in [21] to obtain other new hybrid algorithm for finding greatest common divisor (gcd) of natural numbers a and b. For regular numbers Euclidean algorithm is quite appropriate [10], [17], [18]. For long numbers binary algorithms are extremely useful [19], [20]. Hybrid algorithms are appropriate for the numbers which are between long and regular numbers [1], [2], [21].

AMS Subject Classification: 11A05, 68W01
Key Words: Euclidean algorithm, hybrid algorithm

1. Introduction

Let a and b be a natural numbers. We construct a new computational way for greatest common divisor of these numbers. For research on Euclidean algorithms, see [1]–[9] and [24]–[40]. We try to give new treatment to this task in [10]–[23]. Our approach leads to faster computational iteration processes.

For testing purposes we will use the following computer: processor – Intel(R) Core(TM) i7-6700HQ CPU 2.60GHz, 2592 Mhz, 4 Core(s), 8 Logical Processor(s), RAM 16 GB, Microsoft Windows 10 Enterprise x64, Microsoft Visual C# 2017 x64.

Received: April 28, 2022
Revised: June 4, 2022
Published: June 6, 2022

International Electronic Journal of Pure and Applied Mathematics – IEJPAM, Volume 16, No. 1 (2022)

2. Main Results

We present new hybrid optimized iterative

Algorithm 1.

```
int g = 0;
if ((a & 1) == 0 && (b & 1) == 0)
do { a >>= 1; b >>= 1; g++; }
while ((a & 1) == 0 && (b & 1) == 0);
while ((a & 1) == 0) a >>= 1;
while ((b & 1) == 0) b >>= 1;
do
if (a > b)
{
a %= b;
if (a < 1) { gcd = b << g; break; }
while ((a & 1) == 0) a >>= 1;
b -= a;
do b >>= 1; while ((b & 1) == 0);
}
else
{
b %= a;
if (b < 1) { gcd = a << g; break; }
while ((b & 1) == 0) b >>= 1;
a -= b;
do a >>= 1; while ((a & 1) == 0);
}
while (true);
```

and its recursive version as

Algorithm 2.

```
static long Euclid(long a, long b)
{
if (a > b)
{
a %= b;
if (a < 1) return b;
if ((a & 1) == 0)
```

International Electronic Journal of Pure and Applied Mathematics – IEJPAM, Volume 16, No. 1 (2022)

```
return Euclid(b, a >> 1);
b -= a;
if ((b & 1) == 0)
return Euclid(b >> 1, a);
}
else
{
b %= a;
if (b < 1) return a;
if ((b & 1) == 0)
return Euclid(a, b >> 1);
a -= b;
if ((a & 1) == 0)
return Euclid(a >> 1, b);
}
return Euclid(a, b);
}
```

The recursive function can be called by:

```
int g = 0;
if ((a & 1) == 0 && (b & 1) == 0)
do { a >>= 1; b >>= 1; g++; }
while ((a & 1) == 0 && (b & 1) == 0);
while ((a & 1) == 0) a >>= 1;
while ((b & 1) == 0) b >>= 1;
gcd = Euclid(a, b) << g;
```

Numerical Example.

For testing of Algorithms 1 and 2 we will use the following main function:

```
long a, b, gcd, d1 = 0;
for (int i = 1; i < 100000001; i++) { a = i; b = 200000002 - i;
//here are placed the source code of algorithm 1
//as well as calling of recursive algorithm 2
d1 += gcd;
}
Console.WriteLine(d1);
```

International Electronic Journal of Pure and Applied Mathematics – IEJPAM, Volume 16, No. 1 (2022)

CPU time results are:

CPU time of Algorithm 1 is: **35.683 seconds.**

CPU time of Algorithm 2 is: **43.325 seconds.**

We will indicate that for the same numerical example **31.620 seconds** and **68.119 seconds** are times of Harris iterative and recursive algorithms [1] respectively and **40.455 seconds** and **95.786 seconds** are times of Tembhurne-Sathe iterative and recursive algorithms [2] respectively

3. Conclusion

We construct new effective hybrid algorithm. Numerical experiments approve its applicability and high speed.

Acknowledgements

This work has been accomplished with the financial support by the Grant No BG05M2OP001-1.001-0003, financed by the Science and Education for Smart Growth Operational Program (2014-2020) and co-financed by the European Union through the European structural and Investment funds.

References

[1] V. Harris, An algorithm for finding the greatest common divisor, Fibonacci Quarterly, 8 (1970), 102–103.

[2] J. Tembhurne, S. Sathe, New Modified Euclidean and Binary Greatest Common Divisor Algorithm, IETE Journal of Research, 62, No. 6 (2016), 852–858.

[3] Th. Cormen, Ch. Leiserson, R. Rivest, Cl. Stein, *Introduction to Algorithms*, 3rd ed., The MIT Press, Cambridge (2009).

[4] K. Garov, A. Rahnev, *Textbook-notes on programming in BASIC for facultative training in mathematics for 9.–10. Grade of ESPU*, Sofia (1986). (in Bulgarian)

[5] A. Golev, *Textbook on algorithms and programs in C#*, University Press "Paisii Hilendarski", Plovdiv (2012). (in Bulgarian)

[6] T. Terzieva, *Introduction to web programming*, University Press "Paisii Hilendarski", Plovdiv (2021), ISBN 978-619-202-623-3. (in Bulgarian)

[7] T. Terzieva, *Development of algorithmic thinking in the Informatics Education*, University Press "Paisii Hilendarski", Plovdiv (2021), ISBN 978-619-202-622-6. (in Bulgarian)

[8] T. Terzieva, *Educational tools for teaching in digital environment*, University Press "Paisii Hilendarski", Plovdiv (2021). (in Bulgarian)

[9] S. Enkov, *Programming in Arduino Environment*, University Press "Paisii Hilendarski", Plovdiv (2017). (in Bulgarian)

[10] A. Iliev, N. Kyurkchiev, A Note on Knuth's Implementation of Euclid's Greatest Common Divisor Algorithm, *International Journal of Pure and Applied Mathematics*, **117** (2017), 603–608.

[11] A. Iliev, N. Kyurkchiev, A Note on Euclidean and Extended Euclidean Algorithms for Greatest Common Divisor for Polynomials, *International Journal of Pure and Applied Mathematics*, **118** (2018), 713–721.

[12] A. Iliev, N. Kyurkchiev, *New Trends in Practical Algorithms: Some Computational and Approximation Aspects*, LAP LAMBERT Academic Publishing, Beau Bassin (2018).

[13] A. Iliev, N. Kyurkchiev, The faster Euclidean algorithm, *Collection of scientific works from conference*, Pamporovo, Bulgaria, 28–30 November 2018, (2019), 15–20.

[14] A. Iliev, N. Kyurkchiev, A. Rahnev, *Nontrivial Practical Algorithms: Part 2*, LAP LAMBERT Academic Publishing, Beau Bassin (2019).

[15] A. Iliev, N. Valchanov, T. Terzieva, Generalization and Optimization of Some Algorithms, *Collection of scientific works of National Conference "Education in Information Society"*, Plovdiv, ADIS, 12–13 May 2009, (2009), 52–58. (in Bulgarian)

[16] A. Iliev, N. Kyurkchiev, A. Rahnev, New Algorithm for Finding Greatest Common Divisor, *Neural, Parallel, and Scientific Computations*, 28 No. 1 (2020), 69–74.

[17] A. Iliev, N. Kyurkchiev, A. Rahnev, A New Improvement Euclidean Algorithm for Greatest Common Divisor. I, *Neural, Parallel, and Scientific Computations*, **26** No. 3 (2018), 355–362.

[18] A. Iliev, N. Kyurkchiev, A. Rahnev, A New Improvement of Least Absolute Remainder Algorithm for Greatest Common Divisor. III, *Neural, Parallel, and Scientific Computations*, **27** No. 1 (2019), 1–9.

International Electronic Journal of Pure and Applied Mathematics – IEJPAM, Volume 16, No. 1 (2022)

[19] A. Iliev, N. Kyurkchiev, A. Rahnev, A New Improvement of Stein's Binary Algorithm for Finding Greatest Common Divisor, *Neural, Parallel, and Scientific Computations*, 28 No. 1 (2020), 75–80.

[20] A. Iliev, N. Kyurkchiev, A. Rahnev, T. Terzieva, A Refinement of the Böh's Algorithm for Computing Modular Multiplicative Inverse, *International Electronic Journal of Pure and Applied Mathematics*, **15** No. 1 (2021), 45–53.

[21] A. Iliev, N. Kyurkchiev, A. Rahnev, V. Kyurkchiev, New Hybrid Algorithm For Greatest Common Divisor, (2022). (preprint)

[22] A. Iliev, N. Kyurkchiev, A. Rahnev, Efficient Binary Algorithm for Kronecker Symbol, *Communications in Applied Analysis*, 25 No. 1 (2021), 11–21.

[23] A. Iliev, N. Kyurkchiev, A. Rahnev, Efficient Algorithm for Kronecker Symbol, *International Electronic Journal of Pure and Applied Mathematics*, 15 No. 1 (2021), 23–30.

[24] D. Knuth, *The Art of Computer Programming, Vol. 2, Seminumerical Algorithms*, 3rd ed., Addison-Wesley, Boston (1998).

[25] A. Rahnev, K. Garov, O. Gavrailov, *Textbook for extracurricular work using BASIC*, MNP Press, Sofia (1985). (in Bulgarian)

[26] A. Rahnev, K. Garov, O. Gavrailov, *BASIC in examples and tasks*, Government Press "Narodna prosveta", Sofia (1990). (in Bulgarian)

[27] N. Kasakliev, *C# Programming Guide*, University Press "Paisii Hilendarski", Plovdiv (2016). (in Bulgarian)

[28] A. Rahnev, N. Pavlov, N. Valchanov, T. Terzieva, *Object Oriented Programming*, Lightning Source UK Ltd., London (2014).

[29] D. Rachmawati, M. Budiman, On Using The First Variant of Dependent RSA Encryption Scheme to Secure Text: A Tutorial, *J. Phys.: Conf. Ser.*, (2020), 1542 012024.

[30] J. A. Erho, J. I. Consul, B. R. Japheth, Juggling Versus Three-Way-Reversal Sequence Rotation Performance Across Four Data Types, *International Journal of Scientific Research in Computer Science and Engineering*, **7** No. 6 (2019), 10–18.

[31] J. L. Butar-butar, F. Sinuhaji, Faktorisasi Polinomial Square-Free dan bukan Square-Free atas Lapangan Hingga Zp, *Jurnal Teori dan Aplikasi Matematika*, **3** No. 2 (2019), 132–142.

International Electronic Journal of Pure and Applied Mathematics – IEJPAM, Volume 16, No. 1 (2022)

[32] L. Akcay, B. Ors, Comparison of RISC-V and transport triggered architectures for a post-quantum cryptography application, *Turk J Elec Eng & Comp Sci*, **29**, (2021), 321–333.

[33] C. Falcon Rodriguez, M. Cruz, C. Falcon, Full Euclidean Algorithm by Means of a Steady Walk, *Applied Mathematics*, **12** (2021), 269–279.

[34] J. L. Butar-Butar, Y. B. P. Siringoringo, Kode Siklik Berulang Dari Kode Linear Fp Atas Lapangan Hingga F Pl Dengan L Bilangan Prima Tertentu, *Barekeng: J. Il. Mat. & Ter.*, **15**, No. 02 (2021), 231–240.

[35] V. Matanski, An Efficient Binary Algorithm for Solving Equation $GCD * 2^{|J-K|} = X * A0 + Y * B0$, Proceedings of Anniversary International Scientific Conference "Computer Technologies and Applications", 15-17 September 2021, Pamporovo, Bulgaria, *Plovdiv University Press*, 79–86, ISBN: 978-619-202-702-5.

[36] H. Gyulyustan, A Note on Euclidean Sequencing Algorithm, Proceedings of the Scientific Conference "Innovative ICT for Digital Research Space in Mathematics, Informatics and Educational Pedagogy", Pamporovo, 7-8.11.2019, *Plovdiv University Press*, (2020), 57–63, ISBN 978-619-202-572-4.

[37] P. Kyurkchiev, V. Matanski, The Faster Euclidean Algorithm for Computing Polynomial Multiplicative Inverse, Proceedings of the Scientific Conference Innovative ICT in Research and Education: Mathematics, Informatics and Information Technologies, Pamporovo, 29-30 November 2018, (2019), 43–48, ISBN: 978-619-202-439-0.

[38] V. Matanski, P. Kyurkchiev, The Faster Lehmer's Greatest Common Divisor Algorithm, Proceedings of the Scientific Conference Innovative ICT in Research and Education: Mathematics, Informatics and Information Technologies, Pamporovo, 29-30 November 2018, (2019), 37–42, ISBN: 978-619-202-439-0.

[39] Z. Ibran, E. Aljatlawi, A. Awin, On Continued Fractions and Their Applications, *Journal of Applied Mathematics and Physics*, **10** (2022), 142–159.

[40] Y. Fan, G. Chen, M. Cui, Formalization of Finite Field GF(2^n) Based on COQ, *Computer Science*, **47** No. 12 (2020), 311–318.

International Electronic Journal of Pure and Applied Mathematics – IEJPAM, Volume 16, No. 1 (2022)