Neural, Parallel, and Scientific Computations, **28**, No. 1 (2020), 89-95 ISSN: 1061-5369

NEW EXTENDED ALGORITHM FOR FINDING GREATEST COMMON DIVISOR

Anton Iliev[1], Nikolay Kyurkchiev[2] and Asen Rahnev[3]

[1,2,3]Faculty of Mathematics and Informatics
University of Plovdiv Paisii Hilendarski
24, Tzar Asen Str., 4000 Plovdiv, BULGARIA

ABSTRACT: In our research [7]–[30] which is dedicated to regular and extended algorithms for finding greatest common divisor we developed new approaches for optimizing the ways of finding solutions of these problems. Here we present a new extended algorithm, which is based on "remainder" and "difference" operations. For long numbers the computational cost of "remainder" operation is high and this motivate us to construct such computational process. The algorithm presented here is next step of research in our previous papers [7]–[30] and does not depend on both the hardware computer system and the software environment as all others that we already have developed.

AMS Subject Classification: 11A05, 68W01
Key Words: extended algorithm, greatest common divisor, reduced number of operations

Received: May 22, 2020; **Accepted:** October 15, 2020;
Published: October 17, 2020. **doi:** 10.12732/npsc.v28i1.10
Dynamic Publishers, Inc., Acad. Publishers, Ltd. https://acadsol.eu/npsc

1. INTRODUCTION

For two natural numbers a and b we look for integer numbers x and y such that $x * a + y * b = gcd$ where gcd is the greatest common divisor. Our way of organizing computational process is near to optimal because in every presented by us algorithm we try to minimize the computational operations which are used. This is typical for all other algorithms [7]–[30] which we are reorganized. Also these new algorithms [7]–[30] are highly symmetrized. This is different to approaches given in [1]–[6] and

[31]–[38], etc. which are using not so well organized in computational point of view Knuth's implementation [31].

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.

2. Main Results.

We propose the following:

Algorithm 1.

```
a0 = a; b0 = b;
x1 = 1; x2 = 0;
do
if (a > b)
{ q = a / b; a %= b; t = x2; x2 = x1 - q * x2; x1 = t;
if (a < 1) { gcd = b; x = x1; y = (b - x * a0) / b0; break; }
else
{ b -= a; t = x2; x2 = x1 - x2; x1 = t;
if (a == b) { gcd = a; x = x1; y = (b - x * a0) / b0; break; }
} }
else
{ q = b / a; b %= a; t = x1; x1 = x2 - q * x1; x2 = t;
if (b < 1) { gcd = a; x = x2; y = (a - x * a0) / b0; break; }
else
{ a -= b; t = x1; x1 = x2 - x1; x2 = t;
if (a == b) { gcd = b; x = x2; y = (a - x * a0) / b0; break; }
} }
while (true);
```

and its recursive implementation as

Algorithm 2.

```
static long Euclid(long a, long b, ref long x, ref long y)
{
```

```
long r = a % b; long q1 = a / b;
if (r < 1) { x = 1; y = 0; return b; }
long u = b - r;
if (u == r) { x = -q1; y = 1; return r; }
long d;
if (r > u) d = Euclid(r, u, ref x, ref y);
else d = Euclid(u, r, ref y, ref x);
y -= x; x -= q1 * y;
return d;
}
```

Numerical Example.

We will compare the new algorithms 1. and 2. with extended Knuth's iterative and recursive [31], [8] implementations respectively.

```
long a, b, x1, x2, x = 0, y = 0, q, r, t, ao, bo, eegcd, gcd, d = 0, a0, b0;
for (int i = 1; i < 100000001; i++) { a = i; b = 200000002 - i;
//here is the source code of every one of algorithms 1 and extended Knuth's
//iterative, and calling of algorithms 2 and extended Knuth's recursive
d += gcd; // For extended Knuth's it will be d += eegcd; [8]
}
Console.WriteLine(d);
```

Both (Algorithm 2 and Knuth's) recursive implementations can be called respectively by:

```
if (a > b) gcd = Euclid(a, b, ref y, ref x); else gcd = Euclid(b, a, ref x, ref y);
```

and

```
if (a > b) eegcd = Euclid(a, b, ref x, ref y); else eegcd = Euclid(b, a, ref y, ref x);
```

CPU time of Algorithm 1 is: **33.465 seconds.**
CPU time of Knuth's iterative is: **35.065 seconds.**

CPU time of Algorithm 2 is: **51.153 seconds.**
CPU time of Knuth's recursive is: **59.604 seconds.**

3. Conclusion

The proposed algorithms are in style of Strassen [35] because they decrease the number of "remainder" operations but increase the number of "difference" operations.

Acknowledgement

This paper is supported by the Project FP19-FMI-002 "Innovative ICT for Digital Research Area in Mathematics, Informatics and Pedagogy of Education" of the Scientific Fund of the University of Plovdiv Paisii Hilendarski, Bulgaria.

REFERENCES

[1] A. Akritas, A new method for computing polynomial greatest common divisors and polynomial remainder sequences, *Numerische Mathematik*, **52** (1988), 119–127.

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

[3] F. Chang, Factoring a Polynomial with Multiple-Roots, *World Academy of Science, Engineering and Technology*, **47** (2008), 492–495.

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

[5] 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)

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

[7] 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.

[8] A. Iliev, N. Kyurkchiev, A. Golev, A Note on Knuth's Implementation of Extended Euclidean Greatest Common Divisor Algorithm, *International Journal of Pure and Applied Mathematics*, **118** (2018), 31–37.

[9] A. Iliev, N. Kyurkchiev, A. Rahnev, A Note on Adaptation of the Knuth's Extended Euclidean Algorithm for Computing Multiplicative Inverse, *International Journal of Pure and Applied Mathematics*, **118** (2018), 281–290.

[10] 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.

[11] A. Iliev, N. Kyurkchiev, A Note on Least Absolute Remainder Euclidean Algorithm for Greatest Common Divisor, *International Journal of Scientific Engineering and Applied Science*, **4** No. 3 (2018), 31–34.

[12] A. Iliev, N. Kyurkchiev, A Note on Knuth's Algorithm for Computing Extended Greatest Common Divisor using SGN Function, *International Journal of Scientific Engineering and Applied Science*, **4** No. 3 (2018), 26–29.

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

[14] A. Iliev, N. Kyurkchiev, 80th Anniversary of the birth of Prof. Donald Knuth, *Biomath Communications*, **5** (2018), 7 pp.

[15] A. Iliev, N. Kyurkchiev, New Realization of the Euclidean Algorithm, *Collection of scientific works of Eleventh National Conference with International Participation Education and Research in the Information Society*, Plovdiv, ADIS, June 1–2, (2018), 180–185. (in Bulgarian)

[16] A. Iliev, N. Kyurkchiev, New Organizing of the Euclid's Algorithm and one of its Applications to the Continued Fractions, *Collection of scientific works from conference "Mathematics. Informatics. Information Technologies. Application in Education"*, Pamporovo, Bulgaria, 10–12 October 2018, (2019), 199–207.

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

[18] A. Iliev, N. Kyurkchiev, The faster extended Euclidean algorithm, *Collection of scientific works from conference*, Pamporovo, Bulgaria, 28–30 November 2018, (2019), 21–26.

[19] P. Kyurkchiev, V. Matanski, The faster Euclidean algorithm for computing polynomial multiplicative inverse, *Collection of scientific works from conference*, Pamporovo, Bulgaria, 28–30 November 2018, (2019), 43–48.

[20] V. Matanski, P. Kyurkchiev, The faster Lehmer's greatest common divisor algorithm, *Collection of scientific works from conference*, Pamporovo, Bulgaria, 28–30 November 2018, (2019), 37–42.

[21] 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.

[22] A. Iliev, N. Kyurkchiev, A. Rahnev, A New Improvement of Harris–Stein Modification of Euclidean Algorithm for Greatest Common Divisor. II, *International Journal of Pure and Applied Mathematics*, **120** No. 3 (2018), 379–388.

[23] 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.

[24] A. Iliev, N. Kyurkchiev, A. Rahnev, A New Improvement of Tembhurne–Sathe Modification of Euclidean Algorithm for Greatest Common Divisor. IV, *Dynamic Systems and Applications*, **28** No. 1 (2019), 143–152.

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

[26] 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)

[27] H. Gyulyustan, A Note on Euclidean Sequencing Algorithm, *Proceedings of the Scientific Conference "Innovative ICT for Digital Research Area in Mathematics, Informatics and Pedagogy of Education"*, Pamporovo, 7–8 November 2019, Plovdiv University Press, 2020, 57–64.

[28] A. Iliev, N. Kyurkchiev, A. Rahnev, New Algorithm for Finding Greatest Common Divisor, preprint.

[29] A. Iliev, N. Kyurkchiev, A. Rahnev, A New Improvement of Stein's Binary Algorithm for Finding Greatest Common Divisor, preprint.

[30] A. Iliev, N. Kyurkchiev, A. Rahnev, A New Improvement of Extended Stein's Binary Algorithm, preprint.

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

[32] Hr. Krushkov, A. Iliev, *Practical programming guide in Pascal, Parts I and II*, Koala press, Plovdiv (2002). (in Bulgarian)

[33] P. Nakov, P. Dobrikov, *Programming=++Algorithms*, 5th ed., Sofia (2015). (in Bulgarian)

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

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

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

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

[38] V. Strassen, Gaussian Elimination is not Optimal, *Numer. Math.* **13**, (1969), 354–356.