DOI https://doi.org/10.15407/csc.2022.02.070
UDC 511

V.K. BILYK, Ph.D. Eng. Sciences, Senior Research Associate,
V.M.Glushkov Institute of Cybernetics of the NAS of Ukraine,
Acad. Glushkov Ave., 40, Kyiv, 03187, Ukraine,
BilykVK@gmail.com

SIMPLE AND VISUAL ALGORITHM FOR FACTORIZING INTEGER NUMBERS

An iterative algorithm for decomposing an integer composite number C into prime factors X_1 and X_2 is proposed in which the properties of Vieta's theorem are used for the reduced quadratic equations $X^2+B\cdot X\cdot C = 0$, when the first approximation in iterative computation is taken equal to the square root of the composite number C, then is $\sqrt{C}$, and B is equal to the rounded up to a larger integer from the number $\sqrt{C}$, that is, $B = \overline{\sqrt{C}}$. In this case, the calculations are carried out by linearly increasing the approximations by one.

Keywords: *factorization of numbers, prime and composite numbers.*

Introduction

As you know, factorization of a natural number is called its decomposition into a product of prime factors [1, 2]. The existence and uniqueness (up to the order of the factors) of such a decomposition follows from the main theorem of arithmetic. The problem of finding effective ways to factorize integers into factors has been of interest to mathematicians for a long time, especially specialists in the field of number theory. Many areas of mathematics and computer science find application in solving this problem. Among them: elliptic curves, algebraic number theory and quantum computing. The assumption that the factorization problem is computationally difficult for large numbers underlies widely used algorithms (for example, RSA), which are practically used in the field of encryption. The RSA algorithm is based on the idea of public key cryptography, where a number must be decomposed into prime factors to break a system.

Previous research

Shor's algorithm is widely known among specialists for factorizing numbers with polynomial complexity, but it can only be implemented on a quantum computer [3]. This prompted specialists to create a quantum computer. The leading countries of the world (USA, China, etc.) are allocating serious funds for this today. At the same time, the question of the existence of a factorization algorithm with polynomial complexity on a classical computer remains one of the important open problems in modern number theory. In [4], an analytical method for the factorization of composite numbers is proposed, where, using a residue ring modulo *m*, the

problem is reduced to solving a quadratic equation on a classical computer. The disadvantage of this method, in my opinion, is the problem of choosing the module m for calculations.

The Main Idea of the Proposal

A simple and intuitive iterative algorithm for decomposing integers into factors is proposed. It is based on the use of Vieta's theorem for quadratic equations. First, it is known [5] that if the reduced quadratic equation $X^2 - 2BX + C = 0$ has real roots, then their sum $X_1 + X_2 = 2B$, and the product $X_1 \cdot X_2 = C$. It is tempting to use the last fact. Secondly, it is known from the theory of numbers [2] that the smallest divisor of a composite integer C, different from one, is greater than one, there is a prime number and it does not exceed $\sqrt{C}$.

Example 1. At Fig. 1 shows a graphical illustration before decomposing the number 21 into factors $3 \cdot 7 = 21$. Here the dotted line is a vertical straight line $X = \sqrt{C} = \sqrt{21} = 4{,}582...$ and the letter "B" marks the values of the coefficient in the quadratic equation corresponding to each parabola.

If the integer is not prime, but composite, then it has at least two factors that are not equal to one, corresponding to the two roots of the quadratic equation.

Let us construct parabolas corresponding to the quadratic equation $X^2 - 2 \cdot B \cdot X + C = 0$, for a specific value of C, in this case $C = 21$, and several values of B near the values of $\sqrt{C}$ (see Fig. 1). For $B = \sqrt{C}$ we get a parabola with intersection with the abscissa axis at one point $X = \sqrt{C} = 4{,}582$. In this case, the determinant of the quadratic equation $D = \sqrt{\left(\left(\sqrt{C}\right)^2 - 21\right)} = 0$.

For $B = 4$, we obtain the upper parabola in Fig. 1. In this case, the determinant of the quadratic equation $D = \sqrt{\left((4)^2 - 21\right)} = \sqrt{(-5)} = j\sqrt{5}$ is not an integer and complex number. There are no intersections of the parabola with the abscissa at all.

And if we take $B = 5$, then we get the lower parabola $X^2 - 2 \cdot 5 + 21 = 0$ with intersections at two points with the abscissa at the points with coordinates $X_{1,2} = \sqrt{25} \pm \sqrt{\left((5)^2 - 21\right)} = 5 \pm 2$. The determinant is integer and positive. We get $X_1 = 7, X_2 = 3$. We check, $7 \cdot 3 = 21 = C$. Lucky, but this does not mean that it will always be so.

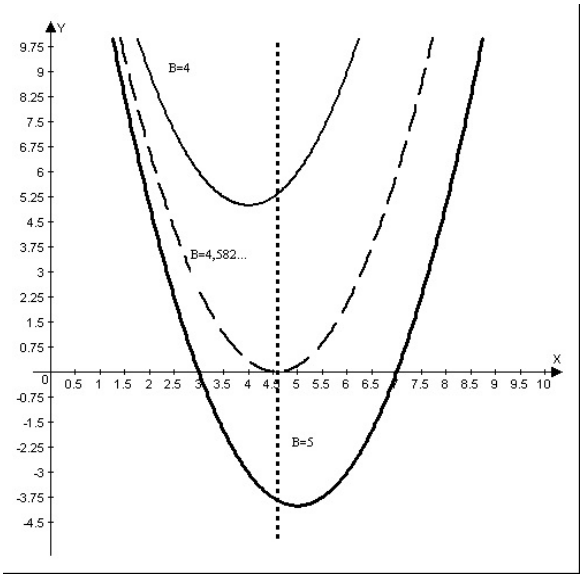

Fig. 1. An illustration of the decomposition of a composite number 21 into prime factors

From the above it follows that in the search for decomposition into prime factors, one should consider parabolas located not higher than the abscissa axis or equations with a coefficient B greater than or equal to $\sqrt{C}$.

Proposition 1. An algorithm is proposed for decomposition of composite integers into prime factors using the formula

$$X_{1,2} = \overline{\sqrt{C}} \pm \sqrt{\left(\overline{\sqrt{C}}\right)^2 - C} = B \pm \sqrt{\left(B^2 - C\right)}. \qquad (*).$$

Here C is a composite number and a free term of the quadratic equation $\overline{\sqrt{C}}$, is the nearest integer greater than the fractional value of the square root, B is the coefficient of the quadratic equation at variable X. Let us check this assumption with a typical example that requires several iterations for computations in comparison with the first example.

Example 2. Let us also illustrate numerically and graphically the execution of the algorithm by the example of the decomposition of an integer $C = 51$ into integer factors $51 = 17 \cdot 3$.

1. Calculate $\sqrt{51} = 7{,}14$. Round up to $\overline{\sqrt{C}} = B = 8$.

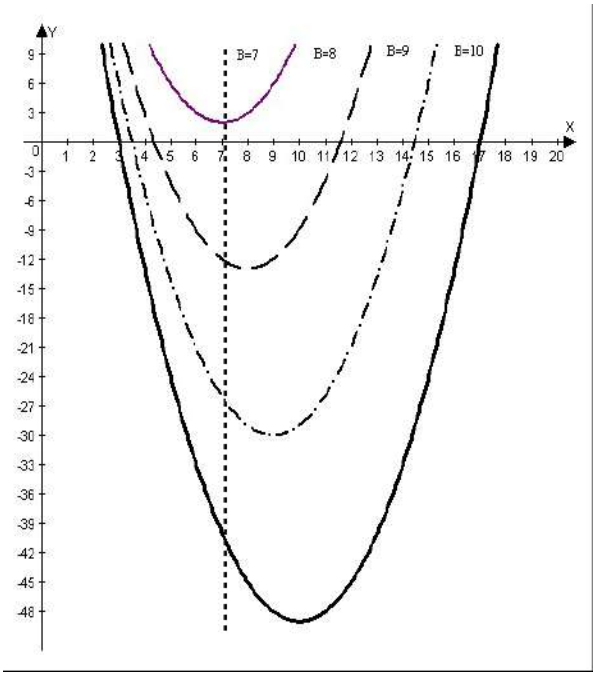

Fig. 2. An illustration of the decomposition of a composite number 51 into prime factors

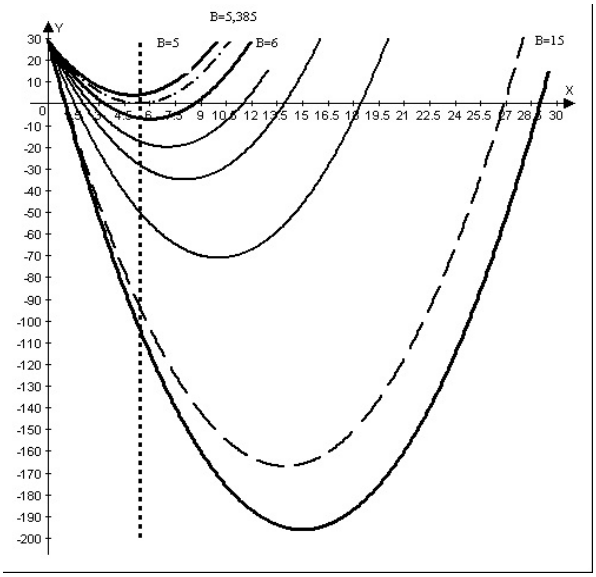

Fig. 3. An illustration of the decomposition of a composite number 29 into prime factors

2. Determinant $D = \sqrt{((\overline{\sqrt{C}})^2 - C)} = \sqrt{\left((B)^2 - C\right)} =$ $= \sqrt{(64 - 51)} = 3{,}6$ — non-integer — go to item 3, and if integer, then go to item 4.

3. Replace $\overline{\sqrt{C}}$ with $(\overline{\sqrt{C}} + 1)$, (or B with $(B+1)$), and go to step 2.

4. Calculate by the formula (*). We check: if $X_1 \cdot X_2 = C$, then we print X_1=3 and X_2=17, otherwise — a failure.

In this case (see Fig. 3), the process will stop at $\overline{\sqrt{C}} = B = 10$ (after three iterations).

Example 3. Let us consider the operation of the algorithm for the extreme case, when an "indecomposable" number, that is, a prime number, arrives at its input. Fig. 3 shows a graphical illustration of the algorithm for calculating the components of the number 29. In this case, it will be a quadratic equation $X^2 - 2 \cdot \sqrt{29} \cdot X + 29 = 0$.

1. Calculate $\sqrt{29} = 5{,}385$. Round up to $\overline{\sqrt{C}} = B = 6$.

2. Calculate the determinant $D = \sqrt{((\overline{\sqrt{C}})^2 - C)} = \sqrt{\left((6)^2 - 29\right)} = \sqrt{7}$ — non-integer — go to item 3, and if it is an integer, then go to item 4.

3. Replace $\overline{\sqrt{C}}$ with $\overline{\sqrt{C}} + 1$ and go to step 2.

4. Check: if $X_1 \cdot X_2 = C$, then we print $X_1 = 1$ and $X_2 = 29$.

In this case (see Fig. 4), the process will stop at $B = 15$.

On the computation time

It can be seen that, in general, the time for computing two factors is approximately equal to the time for computing two square roots (in fact, the time for computing one factor is approximately equal to the time for computing one square root). An unpleasant exception is the decomposition of "non-decomposable", that is, prime numbers (see Fig. 3). In the latter case, you have to pay more — the computation time increases to 10 iterations.

Proposition 2 (updated). Based on the consi-deration of the above particular examples, we write in general form the algorithm for decomposing the composite number C in the form of the following sequence of actions. Fig. 4 shows a graph-diagram of the algorithm.

So, we enter the original composite number C into the shift register and proceed to the calculations (see Fig. 4).

1. Calculate $\sqrt{C}$ and round the result to the nearest larger integer $\overline{\sqrt{C}}$.

2. Calculate the determinant $D = \sqrt{\left(\left(\overline{\sqrt{C}}\right)^2 - C\right)}$.

3. Compare: if D is a fractional number, then go to step 4, otherwise go to step 5.

4. Replace $\overline{\sqrt{C}}$ with $\overline{\sqrt{C}}+1$ (increase by 1) and go to step 2.

5. Calculate $X_{1,2} = \overline{\sqrt{C}} \pm \sqrt{\left(\left(\overline{\sqrt{C}}\right)^2 - C\right)}$.

6. Compare: if $X_2 = 1$, then we write into memory the result X_1 — a prime number, otherwise we enter into the shift register two numbers X_1 and X_2 and alternately serve them according to clause 2 instead of the original number C.

7. The computation process stops when the numbers in the shift register are exhausted.

To prove the correctness of the proposed algorithm, consider the following example.

Example 4. Let us demonstrate the implementation of the proposed algorithm by the example of decomposing a composite number 135 into factors up to prime (factors) $135 = 15 \cdot 9 = (5 \cdot 3) \cdot (3 \cdot (3 \cdot 1))$ (see Fig. 5 and 6) Let's write down the algorithm step by step.

1. $C = 135$. $\sqrt{C} = 11{,}619$. Round to $\overline{\sqrt{C}} = B = 12$. Equation $X^2 - 2 \cdot 12 \cdot X + 135 = 0$. The determinant of the equation $D = \sqrt{\left(\left(\overline{12}\right)^2 - 135\right)} = \sqrt{(144 - 135)} = \sqrt{9} = 3$. We calculate according to the formula (*). $X_1 = 12 + 3 = 15$. $X_2 = 12 - 3 = 9$. We continue the expansion of the obtained factors.

2. $C = 15$. $\sqrt{15} = 3{,}872$. Round up to $\overline{\sqrt{C}} = B = 4$. Equation $X^2 - 2 \cdot 4 \cdot X + 15 = 0$. Determinant $D = \sqrt{(16 - 15)} = 1$. $X_1 = 4 + 1 = 5$. $X_2 = 4 - 1 = 3$.

3. $C = 9$. $\sqrt{9} = 3$. "Round" to $\overline{\sqrt{C}} = B = 3$. Determinant $D = \sqrt{(9 - 9)} = 0$. $X_1 = X_2 = 3$.

4. $C = 5$. $\sqrt{5} = 2{,}23$. Round up to $\overline{\sqrt{C}} = B = 3$. Determinant $D = \sqrt{(9 - 5)} = 2$. $X_1 = 3 + 2 = 5$, $X_2 = 3 - 2 = 1$.

5. $C = 3$. $\sqrt{3} = 1{,}73$. Round up to $\overline{\sqrt{C}} = B = 2$. Determinant $D = \sqrt{(4 - 3)} = 1$. $X_1 = 2 + 1 = 3$, $X_2 = 2 - 1 = 1$.

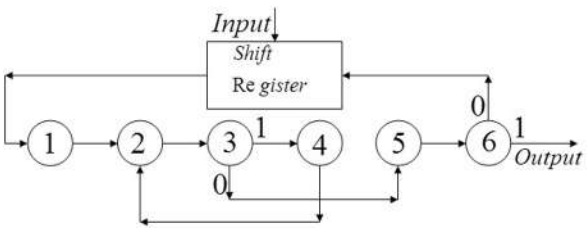

Fig. 4. Graph-diagram of the proposed algorithm for factorizing composite numbers

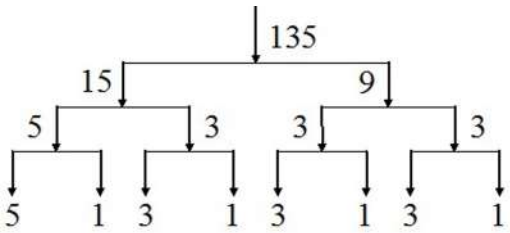

Fig. 5. Graph-illustration of obtaining the results of factorization of the number 135

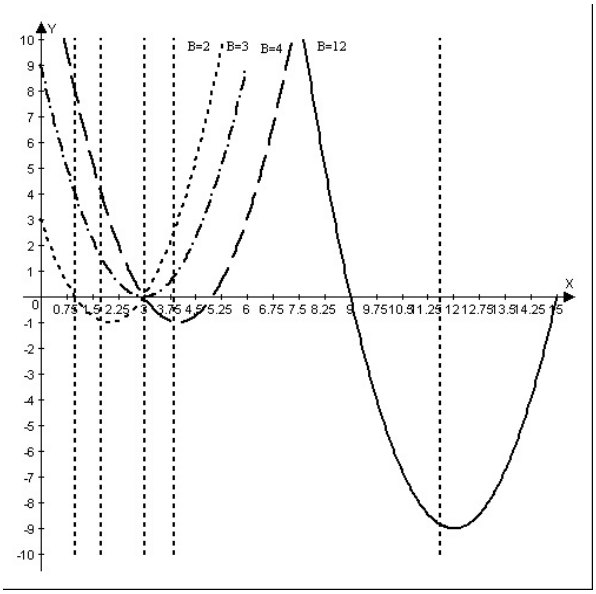

Fig. 6. Graph-illustration of obtaining the results of factorization of the number 135

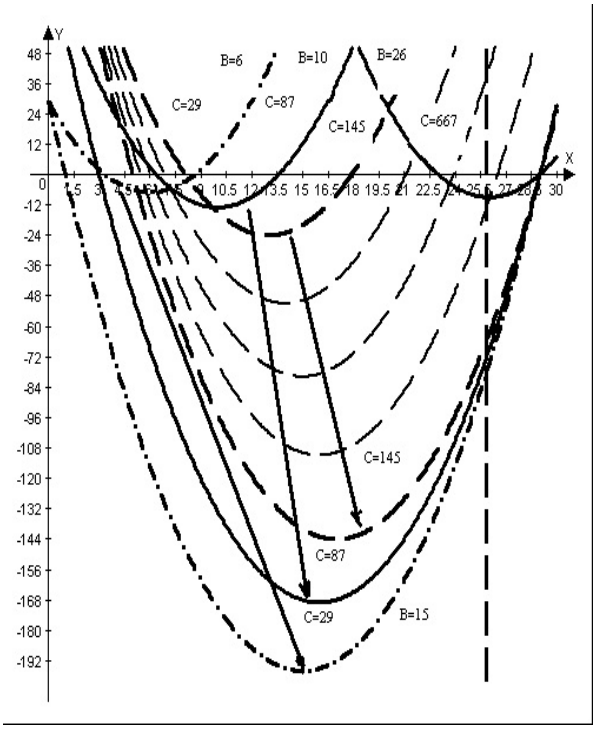

Fig. 7. An illustration of the accelerated decomposition of the number 29 into prime factors

6. $C = 1$. $\sqrt{1} = 1$. "Round off" to $\overline{\sqrt{C}} = B = 1$. Determinant $D = \sqrt{(1-1)} = 0$. $X_1 = X_2 = 1$.

At the same time, the following numbers were received from the output of the shift register: 135, 15, 9, 5, 3, 3, 3. From the output of block 6, prime numbers will come out — factors 5, 3, 1. Fig. 5 shows a graph-illustration of obtaining the results of factorization of the number 135.

In Fig.6. an illustration of the decomposition of the composite number 135 into simple integer factors in the form of parabolic graphs is given. Here, the vertical dotted lines correspond to the $\sqrt{C}$ values, and the parabolas are arranged from right to left as the C numbers decrease (135, 15, 9, 3). Fig. 5 and 6 clearly show that the proposed algorithm is a convergent process. So in Fig.6 it can be seen that the smaller of the two divisors of the original number C is always less than $\overline{\sqrt{C}}$ and in the process of iterative calculations, the left branches of the parabola are gradually shifted from right to left towards one.

Example 5. After we are convinced of the successful completion of iterative calculations for decomposing various integers into prime factors, we can think about improving the proposed algorithm (now there is something to improve). As can be seen from Fig. 3, one of the reasons for the large number of iterations, in the case of the number 29, is the minimum (equal to 1) value of the "partner" in the pair (1,29). Let's try to replace it. In the equality $X_1 \cdot X_2 = C$, we multiply both sides of the equality by the value of a previously known prime number less than 29.

For example, if $1 \cdot 29 = 29$, then, multiplying by 3, we get $3 \cdot 29 = 87$ (see in Fig. 7 the transition from arrow $C = 29$ to arrow $C = 87$). Now we will expand the number $87 = 3 \cdot 29$. Let's compare the number of iterations: $10 - 7 = 3$. It looks like we are heading in the right direction. Let's take an even larger prime number 5. We get $C = 5 \cdot 29 = 145$. Let's expand the number 145. See the transition to the arrow $C = 145$ in Fig. 7. Let's compare the number of iterations in comparison with the previous case $7 - 5 = 2$. It's good. The iterations will stop completely for the pair $C = 17 \cdot 29 = 493$.

But an even more decisive step can be taken. We have multiplied before, by an almost random prime number. But if we knew the previous prime number before the decomposed, then no iterations would be required at all, and the result would be obtained by calculating only two square roots.

Indeed, for the number 29, the previous prime is the number 23. We know that $X_1 + X_2 = 2 \cdot B$. In this example, $23 + 29 = 52$, from which we get $B = 26$. Calculate $C = 29 \cdot 23 = 667$, $X_{1,2} = 26 \pm \pm \sqrt{((26) \cdot 2 - 667)} = 26 \pm 3$. $X_1 = 23, X_2 = 29$. See the parabola in Fig. 7. in the upper right corner. We also see that the sign of the completion of calculations in this case is that the determinant is equal to an integer $\sqrt{((26)\, 2 - 667)} = 3$ and $X_2 = 23$.

Note that in this case, the comparison of the smaller of the two roots $X_2 = 1$ in item 6 should be replaced by $X_2 = 23$ in the block diagram in Fig. 4.

Conclusion

After considering the examples, it can be seen that, by alternately decomposing any composite number into two factors, we find all integer factors, including both prime and multiple for a given integer. This confirms the validity of formula (*).

The proposed algorithm has distinctive features from that indicated in the literature [4]. The relationship between the proposed algorithm and the one known from the literature [4] is the same that exists between direct and iterative methods for solving systems of linear algebraic equations. The use of one or the other depends on the specific circumstances (the advantages and disadvantages of iterative algorithms are known for computing on a computer). The article proposes one of the options for eliminating the reduction in the number of iterations (disadvantage). In this case, one should point out such advantages of the proposed algorithm as simplicity and a visual representation of its implementation.

REFERENCES

1. *Ishmukhametov, Sh.T.,* 2011. Methods of factorization of natural numbers: textbook. Kazan: Kazan University, 190 p. (In Russian).
2. *Vinogradov, I.M.*, 1981. Fundamentals of number theory. M.: Nauka. 176 p (In Russian).
3. *Shor, P.*, 1997. "Polynomial – Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer". SIAM Jour.Comp., Vol. 26, N 5, pp. 1484–1509.
4. *Semotyuk, M.V.*, 2013. "On the analytical method of factorization of composite numbers". Computers zasobi, festoon and systems. N 12, pp. C. 5–10. (In Ukrainian).
5. *Gusev, V.A., Mordkovich, A.G.,* 1988. Mathematics: Ref. materials. M.: Education, 416 p. (In Russian).

Received 30.03.2021

ЛІТЕРАТУРА

1. *Ишмухаметов Ш.Т.* Методы факторизации натуральных чисел: учебное пособие. Казань: Казан. університет, 2011. 200 с.
2. *Виноградов І.М*. Основи теорії чисел. М .: Наука.1981. 176 с.
3. *Shor P.* Polynomial – Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer. SIAM Jour. Comp., 1997. Vol. 26. N 5. P. 1484–1509.
4. *Семотюк М. В.* Про аналітичному методі факторизации складених чисел. Комп'ютерні засоби, мережі та системи. 2013, No 12. С. 5-10.
5. *Гусєв В. А., Мордкович А. Г*. Математика: Справочные материалы. М.: Просвещение, 1988. 16 с.

Надійшла 30.03.2021

В.К. Білик, кандидат технічних наук, старший науковий співробітник,
Інститут кібернетики імені В.М. Глушкова НАН України,
03187, м, Київ, просп. Академіка Глушкова, 40, Україна,
BilykVK@gmail.com

ПРОСТИЙ НАОЧНИЙ АЛГОРИТМ ФАКТОРИЗАЦІЇ ЦІЛИХ ЧИСЕЛ

Вступ. Пошук ефективних способів розкладання цілих чисел на прості множники цікавить спеціалістів не тільки в області теорії чисел, але й інформатики. Проблема у складності (швидкості) обчислень.

Ціль статті. Пропонується альтернативній варіант алгоритму з покращеними характеристиками.

Методи. Замість відомих переборних алгоритмів запропоновано спосіб на іншій (ітераційній) основі.

Результати. Запропоновано простий наочний алгоритм розкладання цілих чисел на співмножники, заснований на використанні теореми Вієта для квадратних рівнянь. По-перше відомо, що якщо квадратне рівняння $X^2 - 2 \cdot B \cdot X + C = 0$ має дійсні корені, то їхня сума $X_1 + X_2 = 2 \cdot B$, а добуток $X_1 \cdot X_2 = C$. Використаємо цей факт.

По-друге, з теорії чисел відомо, що найменший, відмінний від одиниці, дільник цілого складеного числа C не перевищує $\sqrt{C}$. Використаємо те, що $\sqrt{C} = 1/2 \cdot (X_1 + X_2) + \Delta$, тобто. $\sqrt{C}$ знаходиться між двома цілими числами.

Розглянемо запропонований алгоритм на прикладі розкладання цілого складеного числа $C = 51$ на його прості цілі співмножники.

1. Обчислюємо $\sqrt{51} = 7{,}14$. Округлимо до найближчого більшого цілого $\overline{\sqrt{C}} = B = 8$.

2. Детермінант відповідного квадратного рівняння $D = \sqrt{\left(B^2 - C\right)} = \sqrt{(64 - 51)} = 3{,}6$ — нецілe число, переходимо до п.3, а якщо детермінант — ціле або нуль, то переходимо до п.4.

3. Замінюємо B на $B + 1$ і переходимо до п.2.

4. Перевіряємо: якщо $X_1 \cdot X_2 = C$, то друкуємо в даному випадку, $X_1 = 3$ и $X_2 = 17$. Інакше — збій.

У випадку, що розглядається, ітераційний процес зупиниться на третій ітерації при $B = 10$.

В статті розглянуто варіанти скорочення числа ітерацій, аж до однієї.

Висновок. Запропоновано простий і наочний безперебірний ітераційний алгоритм розкладання цілих чисел на співмножники, заснований на використанні теоремі Вієта для квадратних рівнянь, який може конкурувати з відомими алгоритмами.

Ключові слова: *факторизація чисел, прості та складені числа.*