

Algorithms in *Everyday Mathematics*

Algorithms in School Mathematics	1
Algorithms in <i>Everyday Mathematics</i>	3
Invented Procedures.....	3
Alternative Algorithms	5
Focus Algorithms.....	6
Algorithmic Thinking	7
References	8

An **algorithm** is a step-by-step procedure designed to achieve a certain objective in a finite time, often with several steps that repeat or “loop” as many times as necessary. The most familiar algorithms are the elementary school procedures for adding, subtracting, multiplying, and dividing, but there are many other algorithms in mathematics.

Algorithms in School Mathematics

The place of algorithms in school mathematics is changing. One reason is the widespread availability of calculators and computers outside of school. Before such machines were invented, the preparation of workers who could carry out complicated computations by hand was an important goal of school mathematics. Today, being able to mimic a \$5 calculator is not enough: Employers want workers who can think mathematically. How the school mathematics curriculum should adapt to this new reality is an open question, but it is clear that proficiency at complicated paper-and-pencil computations is far less important outside of school today than in the past. It is also clear that the time saved by reducing attention to such computations in school can be put to better use on such topics as problem solving, estimation, mental arithmetic, geometry, and data analysis (NCTM, 1989).

Another reason the role of algorithms is changing is that researchers have identified a number of serious problems with the traditional approach to teaching computation. One problem is that the traditional approach fails with a large number of students. Despite heavy emphasis on paper-and-pencil computation, many students never become proficient in carrying out algorithms for the basic operations. In one study, only 60 percent of U.S. ten-year-olds achieved mastery of subtraction using the standard “borrowing” algorithm. A Japanese study found that only 56 percent of third graders and 74 percent of fifth graders achieved mastery of this algorithm. A principal cause for such failures is an overemphasis on procedural proficiency with insufficient attention to the conceptual basis for the procedures. This unbalanced approach produces students who are plagued by “bugs,” such as always taking the smaller digit from the larger in subtraction, because they are trying to carry out imperfectly understood procedures.

An even more serious problem with the traditional approach to teaching computation is that it engenders beliefs about mathematics that impede further learning. Research indicates that these

beliefs begin to be formed during the elementary school years when the focus is on mastery of standard algorithms (Hiebert, 1984; Cobb, 1985; Baroody & Ginsburg, 1986). The traditional, rote approach to teaching algorithms fosters beliefs such as the following:

- mathematics consists mostly of symbols on paper;
- following the rules for manipulating those symbols is of prime importance;
- mathematics is mostly memorization;
- mathematics problems can be solved in no more than 10 minutes — or else they cannot be solved at all;
- speed and accuracy are more important in mathematics than understanding;
- there is one right way to solve any problem;
- different (correct) methods of solution sometimes yield contradictory results; and
- mathematics symbols and rules have little to do with common sense, intuition, or the real world.

These inaccurate beliefs lead to negative attitudes. The prevalence of math phobia, the social acceptability of mathematical incompetence, and the avoidance of mathematics in high school and beyond indicate that many people feel that mathematics is difficult and unpleasant. Researchers suggest that these attitudes begin to be formed when students are taught the standard algorithms in the primary grades. Hiebert (1984) writes, “Most children enter school with reasonably good problem-solving strategies. A significant feature of these strategies is that they reflect a careful analysis of the problems to which they are applied. However, after several years many children abandon their analytic approach and solve problems by selecting a memorized algorithm based on a relatively superficial reading of the problem.” By third or fourth grade, according to Hiebert, “many students see little connection between the procedures they use and the understandings that support them. This is true even for students who demonstrate in concrete contexts that they do possess important understandings.” Baroody and Ginsburg (1986) make a similar claim: “For most children, school mathematics involves the mechanical learning and the mechanical use of facts — adaptations to a system that are unencumbered by the demands of consistency or even common sense.”

A third major reason for changes in the treatment of algorithms in school mathematics is that a better approach exists. Instead of suppressing children’s natural problem-solving strategies, this new approach builds on them (Hiebert, 1984; Cobb, 1985; Baroody & Ginsburg, 1986; Resnick, Lesgold, & Bill, 1990). For example, young children often use counting strategies to solve problems. By encouraging the use of such strategies and by teaching even more sophisticated counting techniques, the new approach helps children become proficient at computation while also preserving their belief that mathematics makes sense. This new approach to computation is described in more detail below.

Reducing the emphasis on complicated paper-and-pencil computations does not mean that paper-and-pencil arithmetic should be eliminated from the school curriculum. Paper-and-pencil skills are practical in certain situations, are not necessarily hard to acquire, and are widely expected as an outcome of elementary education. If taught properly, with understanding but without demands for “mastery” by all students by some fixed time, paper-and-pencil algorithms can reinforce students’ understanding of our number system and of the operations themselves. Exploring algorithms can also build estimation and mental arithmetic skills and help students see mathematics as a meaningful and creative subject.

Algorithms in *Everyday Mathematics*

Everyday Mathematics includes a comprehensive treatment of computation. Students learn to compute mentally, with paper and pencil, and by machine; they learn to find both exact and approximate results; and, most importantly, they learn what computations to make and how to interpret their answers. The following sections describe in general terms how *Everyday Mathematics* approaches exact paper-and-pencil methods for basic operations with whole numbers. For details about particular algorithms and for information about how the program teaches mental arithmetic, estimation, and computation with decimals and fractions, see the *Everyday Mathematics Teacher's Reference Manual*.

In *Everyday Mathematics*, computational proficiency develops gradually. In the beginning, before they have learned formal procedures, students use what they know to solve problems. They use their common sense and their informal knowledge of mathematics to devise their own procedures for adding, subtracting, and so on. As students describe, compare, and refine their approaches, several alternative methods are identified. Some of these alternatives are based on students' own ideas; others are introduced by the teacher or in the materials. For each basic operation, students are expected to become proficient at one or more of these alternative methods.

The materials also identify one of the alternative algorithms for each operation as a *focus algorithm*. The purpose of the focus algorithms is two-fold: (i) to provide back-up methods for those students who do not achieve proficiency using other algorithms, and (ii) to provide a common basis for further work. All students are expected to learn the focus algorithms at some point, though, as always in *Everyday Mathematics*, students are encouraged to use whatever method they prefer when they solve problems.

The following sections describe this process in more detail. Note, however, that although the basic approach is similar across all four operations, the emphasis varies from operation to operation because of differences among the operations and differences in students' background knowledge. For example, it is easier to invent efficient procedures for addition than for division. There is, accordingly, less expectation that students will devise efficient procedures for solving multidigit long division problems than that they will succeed in finding their own good ways to solve multidigit addition problems.

Invented Procedures

When they are first learning an operation, *Everyday Mathematics* students are asked to solve problems involving the operation before they have developed or learned systematic procedures for solving such problems. In second grade, for example, students are asked to solve multidigit subtraction problems. They might solve such problems by counting up from the smaller to the larger number, or by using tools such as number grids or base-10 blocks, or they may use some other strategy that makes sense to them. This stage of algorithm development may be called the invented procedures phase.

To succeed in devising effective procedures, students must have a good background in the following areas:

- *Our system for writing numbers*. In particular, students need to understand place value.
- *Basic facts*. To be successful at carrying out multistep computational procedures,

students need proficiency with the basic arithmetic facts.

- *The meanings of the operations and the relationships among operations.* To solve $37 - 25$, for example, a student might reason, “What number must I add to 25 to get 37?”

Research indicates that students can succeed in inventing their own methods for solving basic computational problems (Madell, 1985; Kamii & Joseph, 1988; Cobb & Merkel, 1989; Resnick, Lesgold, & Bill, 1990; Carpenter, Fennema, & Franke, 1992). Inventing procedures flourishes when:

- the classroom environment is accepting and supportive;
- adequate time for experimentation is allotted;
- computational tasks are embedded in real-life contexts; and
- students discuss their solution strategies with the teacher and with one another.

The discussion of students’ methods is especially important. Through classroom discussion, teachers gain valuable insight into students’ thinking and progress, while students become more skilled at communicating mathematics and at understanding and critiquing others’ ideas and methods. Talking about why a method works, whether a method will work in every case, which method is most efficient, and so on, helps students understand that mathematics is based on common sense and objective reason, not the teacher’s whim. Such discussions lay the foundations for later formal work with proof.

The invented-procedures approach to algorithm development has many advantages:

- Students who invent their own methods learn that their intuitive methods are valid and that mathematics makes sense.
- Inventing procedures promotes conceptual understanding of the operations and of base-10 place-value numeration. When students build their own procedures on their prior mathematical knowledge and common sense, new knowledge is integrated into a meaningful network so that it is understood better and retained more easily.
- Inventing procedures promotes proficiency with mental arithmetic. Many techniques that students invent are much more effective for mental arithmetic than standard paper-and-pencil algorithms. Students develop a broad repertoire of computational methods and the flexibility to choose whichever procedure is most appropriate in any particular situation.
- Inventing procedures involves solving problems that the students do not already know how to solve, so they gain valuable experience with non-routine problems. They must learn to manage their resources: *How long will this take? Am I wasting my time with this approach? Is there a better way?* Such resource management is especially important in complex problem solving. As students devise their own methods, they also develop persistence and confidence in dealing with difficult problems.
- Students are more motivated when they don’t have to learn standard paper-and-pencil algorithms by rote. People are more interested in what they can understand, and students generally understand their own methods.
- Students become adept at changing the representations of ideas and problems, translating readily among manipulatives, words, pictures, and symbols. The ability to represent a problem in more than one way is important in problem solving. Students also develop the

ability to transform any given problem into an equivalent, easier problem. For example, $32 - 17$ can be transformed to $35 - 20$ by adding 3 to both numbers.

Another argument in favor of the invented-procedures approach is that learning a single standard algorithm for each operation, especially at an early stage, may actually inhibit the development of students' mathematical understanding. Premature teaching of standard paper-and-pencil algorithms can foster persistent errors and buggy algorithms and can lead students to use the algorithms as substitutes for thinking and common sense.

Alternative Algorithms

Over the centuries, people have invented many algorithms for the basic arithmetic operations. Each of these historical algorithms was developed in some context. For example, one does not need to know the multiplication tables to do “Russian Peasant Multiplication” — all that is required is doubling, halving, and adding. Many historical algorithms were “standard” at some time and place, and some are used to this day. The current “European” method of subtraction, for example, is not the same as the method most Americans learned in school.

The U.S. standard algorithms—those that have been most widely taught in this country in the past 100 years—are highly efficient for paper-and-pencil computation, but that does not necessarily make them the best choice for school mathematics today. The best algorithm for one purpose may not be the best algorithm for another purpose. The most efficient algorithm for paper-and-pencil computation is not likely to be the best algorithm for helping students understand the operation, nor is it likely to be the best algorithm for mental arithmetic and estimation. Moreover, if efficiency is the goal, in most situations it is unlikely that *any* paper-and-pencil algorithm will be superior to mental arithmetic or a calculator.

If paper-and-pencil computation is to continue to be part of the elementary school mathematics curriculum, as the authors of *Everyday Mathematics* believe it should, then alternatives to the U.S. standard algorithms should be considered. Such alternatives may have better cost-benefit ratios than the standard algorithms. Historical algorithms are one source of alternatives. Student-invented procedures are another rich source. A third source is mathematicians and mathematics educators who are devising new methods that are well adapted to our needs today. The *Everyday Mathematics* approach to computation uses alternative algorithms from all these sources.

In *Everyday Mathematics*, as students explain, compare, and contrast their own invented procedures, several common alternative methods are identified. Often these are formalizations of approaches that students have devised. The column-addition method, for example, was shown and explained to the *Everyday Mathematics* authors by a first grader. Other alternative algorithms, including both historical and new algorithms, are introduced by the teacher or the materials. The partial-quotients method, for example, first appeared in print in Isaac Greenwood's *Arithmeticks* in 1729.

Many alternative algorithms, whether based on student methods or introduced by the teacher, are highly efficient and easier to understand and learn than the U.S. traditional algorithms. For example, lattice multiplication requires only a knowledge of basic multiplication facts and the ability to add strings of single-digit numbers, and yet it is more efficient than the traditional long multiplication algorithm for all but the simplest multidigit problems. Students are urged to

experiment with various methods for each operation in order to become proficient at using at least one alternative.

The alternative-algorithms phase of algorithm development has significant advantages:

- A key belief in *Everyday Mathematics* is that problems can (and should) be solved in more than one way. This belief in multiple solutions is supported by the alternative-algorithms approach to developing computational proficiency.
- Providing several alternative algorithms for each operation affords flexibility. A one-size-fits-all approach may work for many students, but the goal in *Everyday Mathematics* is to reach *all* students. One algorithm may work well for one student, but another algorithm may be better for another student.
- Different algorithms are often based on different concepts, so studying several algorithms for an operation can help students understand the operation.
- Presenting several alternative algorithms gives the message that mathematics is a creative field. In today's rapidly changing world, people who can break away from traditional ways of thinking are especially valuable.

Teaching multiple algorithms for important operations is common in mathematics outside the elementary school. In computer science, for example, alternative algorithms for fundamental operations are always included in textbooks. An entire volume of Donald Knuth's monumental work, *The Art of Computer Programming* (1998), is devoted exclusively to sorting and searching. Knuth presents many inefficient sorting algorithms because they are instructive.

Focus Algorithms

The authors of *Everyday Mathematics* believe that the invented-procedures/alternative-algorithms approach described above is a radical improvement over the traditional approach to developing computational proficiency. The *Everyday Mathematics* approach is based on decades of research and was refined during extensive fieldtesting. Student achievement studies indicate, moreover, that when the approach is properly implemented, students do achieve high levels of computational proficiency (Carroll, 1996, 1997; Carroll & Porter, 1997, 1998; Fuson, Carroll, & Drucek, 2000; Carroll, Fuson, & Diamond, 2000; Carroll & Isaacs, in press).

In the second edition of *Everyday Mathematics*, the approach described above is extended in one significant way: For each operation, one of the several alternative algorithms is identified as a focus algorithm. All students are expected to learn the focus algorithms eventually, although, as usual in *Everyday Mathematics*, proficiency is expected only after multiple exposures over several years. Students are also not required to use the focus algorithms in solving problems if they have alternatives they prefer. For addition, the focus algorithm is partial-sums; for subtraction, trade-first; for multiplication, partial-products; and for division, partial-quotients. (See the *Everyday Mathematics Teacher's Reference Manual* for details about these and other algorithms.)

The focus algorithms are powerful, relatively efficient, and easy to understand and learn, but they are not meant to short-circuit the invented-procedures/alternative-algorithms approach described above. Students still need to grapple with problems on their own and explore alternative algorithms. The focus algorithms have been introduced for two specific reasons. One

is that they provide reliable alternatives for students who do not develop effective procedures on their own. The focus algorithm for subtraction, for example, is introduced in second grade. Second grade students are not expected to be proficient with the method, though they are expected to be able to solve multidigit subtraction problems in some way, by using counting, number grids, manipulatives, or some other method. A fourth grade student, however, who does not have a reliable method for subtraction despite several years of work with invented procedures and alternative algorithms should focus on the trade-first method so that he or she will have at least one reliable way to subtract with paper and pencil. One aim of the focus-algorithm approach is to promote flexibility while ensuring that all students know at least one reliable method for each operation.

Another reason for introducing focus algorithms is to provide a common ground for the further development of mathematical ideas. Most algorithms for operations with whole numbers, for example, can be extended to decimals. This is easier to show in a class at least one whole-number algorithm for each operation is known by every student. Focus algorithms provide a common language that facilitates classroom discussion.

Focus algorithms were introduced in response to teachers' concerns. However, a teacher who has developed an effective strategy for teaching algorithms, and who feels that the focus-algorithm approach is unnecessary or compromises that strategy, is not obliged to adopt the focus-algorithm approach.

Algorithmic Thinking

Mathematics advances in part through the development of efficient procedures that reduce difficult tasks to routine exercises that can be carried out without effort of thought. Alfred North Whitehead expressed this idea memorably in his book, *An Introduction to Mathematics* (1911): "It is a profoundly erroneous truism, repeated by all copy books and by eminent people when they are making speeches, that we should cultivate the habit of thinking of what we are doing. The precise opposite is the case. Civilization advances by extending the number of important operations which we can perform without thinking about them" (p. 61).

An effective algorithm can be used to efficiently solve an entire class of problems, without having to think through each problem from first principles. Knowing algorithms increases students' mathematical power, which is a principal goal of school mathematics (NCTM, 1989). The approach described in this paper — invented procedures followed by alternative algorithms, with focus algorithms as a backup and a basis for further work — will produce students who understand their methods and can carry them out proficiently so that they can think about more important things, such as why they are doing what they are doing and what their results mean. The approach improves students' mental arithmetic skills, helps them understand the operations, and develops sound number sense, including a good understanding of place value. The emphasis on multiple solutions, including both inventing new procedures and making sense of others' inventions, encourages the belief that mathematics is creative and sensible. In *Everyday Mathematics*, accordingly, an increase in mathematical power through algorithmic proficiency is achieved at the same time that other important objectives are being met.

The authors of *Everyday Mathematics* have also found that the study of paper-and-pencil computational algorithms can be valuable for developing algorithmic thinking in general. For

this reason, explicit discussions of algorithms occur in lessons devoted to computation. Algorithmic and procedural thinking includes:

- understanding specific algorithms or procedures provided by other people,
- applying known algorithms to everyday problems,
- adapting known algorithms to fit new situations,
- developing new algorithms and procedures when necessary, and
- recognizing the limitations of algorithms and procedures so they are not used inappropriately.

By studying computational algorithms, students can learn things that will carry over to other areas of their lives. More and more, people need to apply algorithmic and procedural thinking in order to operate technologically advanced devices. Algorithms beyond arithmetic are increasingly important in theoretical mathematics, in applications of mathematics, in computer science, and in many areas outside of mathematics.

References

- Baroody, A. J., & Ginsburg, H. P. (1986). The relationship between initial meaning and mechanical knowledge of arithmetic. In J. Hiebert (Ed.), *Conceptual and procedural knowledge: The case of mathematics*. Hillsdale, NJ: Erlbaum.
- Carpenter, T. P., Fennema, E., & Franke, M. L. (1992). Cognitively guided instruction: Building the primary mathematics curriculum on children's informal mathematical knowledge. A paper presented at the annual meeting of the American Educational Research Association, April 1992, San Francisco.
- Carroll, W. M. (1996). Use of invented algorithms by second graders in a reform mathematics curriculum. *Journal of Mathematical Behavior*, 15: 137-150.
- Carroll, W. M. (1997). Mental and written computation: Abilities of students in a reform-based curriculum. *The Mathematics Educator*, 2 (1): 18-32.
- Carroll, W., & Isaacs, A. (in press). Achievement of students using the University of Chicago School Mathematics Project's Everyday Mathematics. In S. Senk & D. Thompson, *Student outcomes in Standards-oriented school mathematics curriculum projects*. Hillsdale, NJ: Erlbaum.
- Carroll, W., & Porter, D. (1997). Invented algorithms can develop meaningful mathematical procedures. *Teaching Children Mathematics* 3(7): 370-74.
- Carroll, W., & Porter, D. (1998). Alternative algorithms for whole-number operations. In L. J. Morrow (Ed.), *The teaching and learning of algorithms in school mathematics: 1998 yearbook* (pp. 106-114). Reston, VA: National Council of Teachers of Mathematics.
- Carroll, W., Fuson, K., & Diamond, A. (2000). Use of student-constructed number stories in a reform-based curriculum. *Journal of Mathematical Behavior*, 19: 49-62.
- Cobb, P. (1985). Two children's anticipations, beliefs, and motivations. *Educational Studies in Mathematics*, 16: 111-126.

- Cobb, P., & Merkel, G. (1989). Thinking strategies: Teaching arithmetic through problem solving. In P. Trafton (Ed.), *New directions for elementary school mathematics: 1989 yearbook*. Reston, VA: National Council of Teachers of Mathematics.
- Fuson, K., Carroll, W. M., & Drucek, J. V. (2000) Achievement results for second and third graders using the Standards-based curriculum Everyday Mathematics. *Journal for Research in Mathematics Education* 31 (3): 277-295.
- Greenwood, I. (1729). *Arithmeticks*. Boston: T. Hancock.
- Hiebert, J. (1984). Children's mathematical learning: The struggle to link form and understanding. *Elementary School Journal*, 84 (5): 497-513.
- Kamii, C., & Joseph, L. (1988). Teaching place value and double-column addition. *Arithmetic Teacher*, 35 (6), pp. 48-52.
- Knuth, D. E. (1998). *The art of computer programming, volume 3: Sorting and searching*. Reading, Massachusetts: Addison-Wesley.
- Madell, R. (1985). Children's natural processes. *Arithmetic Teacher*, 32 (7), pp. 20-22.
- National Council of Teachers of Mathematics. (1989). *Curriculum and evaluation standards for school mathematics*. Reston, VA: Author.
- Resnick, L. B., Lesgold, S., & Bill, V. (1990). From protoquantities to number sense. A paper prepared for the Psychology of Mathematics Education Conference, Mexico City.
- Whitehead, A. N. (1911). *An introduction to mathematics*. New York: Holt.