= 102 again ; the 1 is carried , and 0 is written at the bottom . The third column : 1 + 1 + 1 = 112 . This time , a 1 is carried , and a 1 is written in the bottom row . Proceeding like this gives the final answer 1001002 (36 decimal) .

= = = Computers = = =

Analog computers work directly with physical quantities , so their addition mechanisms depend on the form of the addends . A mechanical adder might represent two addends as the positions of sliding blocks , in which case they can be added with an averaging lever . If the addends are the rotation speeds of two shafts , they can be added with a differential . A hydraulic adder can add the pressures in two chambers by exploiting Newton 's second law to balance forces on an assembly of pistons . The most common situation for a general @-@ purpose analog computer is to add two voltages (referenced to ground) ; this can be accomplished roughly with a resistor network , but a better design exploits an operational amplifier .

Addition is also fundamental to the operation of digital computers , where the efficiency of addition , in particular the carry mechanism , is an important limitation to overall performance .

The abacus , also called a counting frame , is a calculating tool that was in use centuries before the adoption of the written modern numeral system and is still widely used by merchants , traders and clerks in Asia , Africa , and elsewhere ; it dates back to at least 2700 ? 2300 BC , when it was used in Sumer .

Blaise Pascal invented the mechanical calculator in 1642 ; it was the first operational adding machine . It made use of a gravity @-@ assisted carry mechanism . It was the only operational mechanical calculator in the 17th century and the earliest automatic , digital computers . Pascal 's calculator was limited by its carry mechanism , which forced its wheels to only turn one way so it could add . To subtract , the operator had to use the Pascal 's calculator 's complement , which required as many steps as an addition . Giovanni Poleni followed Pascal , building the second functional mechanical calculator in 1709 , a calculating clock made of wood that , once setup , could multiply two numbers automatically .

Adders execute integer addition in electronic digital computers , usually using binary arithmetic . The simplest architecture is the ripple carry adder , which follows the standard multi @-@ digit algorithm . One slight improvement is the carry skip design , again following human intuition ; one does not perform all the carries in computing 999 + 1 , but one bypasses the group of 9s and skips to the answer .

In practice , comutational addition may achieved via XOR and AND bitwise logical operations in conjunction with bitshift operations as shown in the pseudocode below . Both XOR and AND gates are straightforward to realize in digital logic allowing the realization of full adder circuits which in turn may be combined into more complex logical operations . In modern digital computers , integer addition is typically the fastest arithmetic instruction , yet it has the largest impact on performance , since it underlies all floating @-@ point operations as well as such basic tasks as address generation during memory access and fetching instructions during branching . To increase speed , modern designs calculate digits in parallel ; these schemes go by such names as carry select , carry lookahead , and the Ling pseudocarry . Many implementations are , in fact , hybrids of these last three designs . Unlike addition on paper , addition on a computer often changes the addends . On the ancient abacus and adding board , both addends are destroyed , leaving only the sum . The influence of the abacus on mathematical thinking was strong enough that early Latin texts often claimed that in the process of adding " a number to a number " , both numbers vanish . In modern times , the ADD instruction of a microprocessor replaces the augend with the sum but preserves the addend . In a high @-@ level programming language , evaluating a + b does not change either a or b ; if the goal is to replace a with the sum this must be explicitly requested , typically with the statement a = a + b . Some languages such as C or C + + allow this to be abbreviated as a + = b .

On a computer , if the result of an addition is too large to store , an arithmetic overflow occurs , resulting in an incorrect answer . Unanticipated arithmetic overflow is a fairly common cause of program errors . Such overflow bugs may be hard to discover and diagnose because they may manifest themselves only for very large input data sets , which are less likely to be used in validation tests . One especially notable such error was the Y2K bug , where overflow errors due to using a 2 @-@ digit format for years caused significant computer problems in 2000 .

= = Addition of numbers = =

To prove the usual properties of addition , one must first define addition for the context in question . Addition is first defined on the natural numbers . In set theory , addition is then extended to progressively larger sets that include the natural numbers : the integers , the rational numbers , and the real numbers . (In mathematics education , positive fractions are added before negative numbers are even considered ; this is also the historical route .)

= = = Natural numbers = = =

There are two popular ways to define the sum of two natural numbers a and b . If one defines natural numbers to be the cardinalities of finite sets , (the cardinality of a set is the number of elements in the set) , then it is appropriate to define their sum as follows :