Addition and subtraction algorithms and their corresponding flowcharts involve systematic steps for performing arithmetic operations. For addition, the basic algorithm involves adding corresponding digits of two numbers, handling carries when the sum exceeds 9. Subtraction algorithms involve borrowing from higher-order digits when needed. Flowcharts visually represent these steps, making them easy to understand and follow.

Addition Algorithm:

1. **Input:** Two numbers, A and B.
2. **Initialization:** Set carry = 0.
3. **Iterate through the digits:** Starting from the least significant digit (rightmost digit), perform the following for each digit position:

* Add the corresponding digits of A and B, and the current carry.
* If the sum is greater than or equal to 10:
  + Store the remainder of the sum divided by 10 (the units digit) as the current digit of the result.
  + Set carry to the quotient of the sum divided by 10 (the carry-over to the next digit).
* Otherwise:
* Store the sum as the current digit of the result.
* Set carry to 0.

1. **Handle the final carry:** If there's a carry left after the last digit, add it as a new digit to the most significant position of the result.
2. **Output:** The resulting sum.

Subtraction Algorithm:

1. **Input:** Two numbers, A and B, where A >= B.
2. **Initialization:** Set borrow = 0.
3. **Iterate through the digits:** Starting from the least significant digit (rightmost digit), perform the following for each digit position:

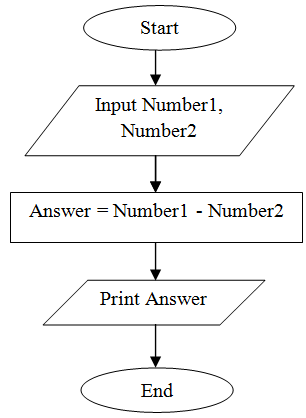
If A's digit is greater than or equal to B's digit plus borrow:

* Subtract B's digit and borrow from A's digit.
* Set the current digit of the result to the difference.
* Set borrow to 0.

Otherwise (borrow needed):

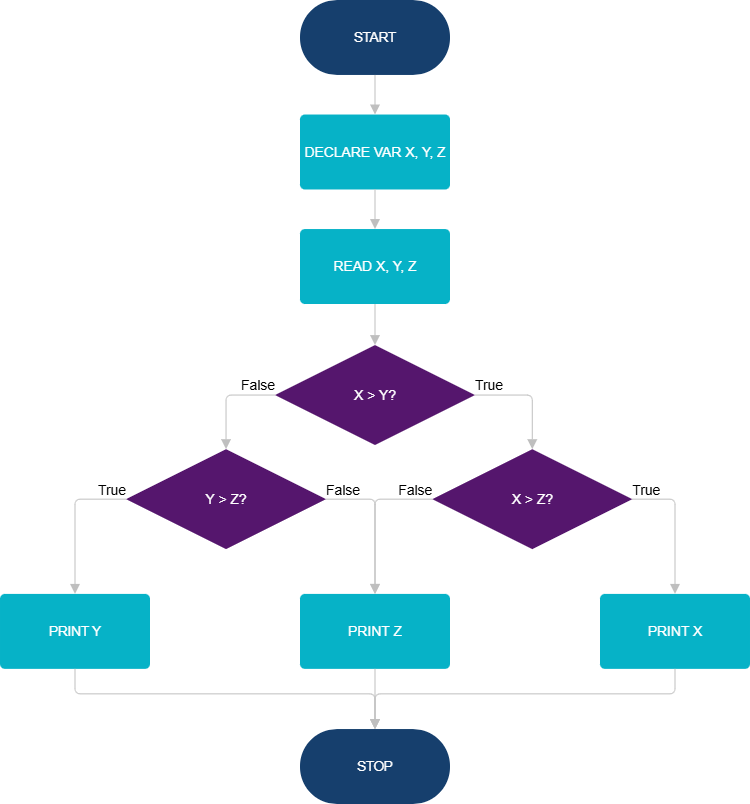
* Subtract B's digit and borrow from A's digit, adding 10 to A's digit (borrowing from the next higher-order digit).
* Set the current digit of the result to the difference.
* Set borrow to 1.

**Outpt:** The resulting difference.



## **Flowchart symbols** Here are some of the common flowchart symbols. For a more comprehensive list, see our full [flowchart symbols page](https://www.lucidchart.com/pages/flowchart-symbols-meaning-explained).

|  |  |
| --- | --- |
| Terminal/Terminator | IMG_256 |
| Process | IMG_257 |
| Decision | IMG_258 |
| Document | IMG_259 |
| Data, or Input/Output | IMG_260 |
| Stored Data | IMG_261 |
| Flow Arrow | IMG_262 |
| Comment or Annotation | IMG_263 |
| Predefined process | IMG_264 |
| On-page connector/reference | IMG_265 |
| Off-page connector/reference | IMG_266 |



<https://app.smartdraw.com/editor.aspx?templateId=c2c78ae9-1616-4b25-85e5-b4ab5f728601>

# Logic Building Problems

Logic building is about creating clear, step-by-step methods to solve problems using simple rules and principles. It’s the heart of coding, enabling programs to think, reason, and arrive at smart solutions just like we do.

Here are some tips for improving your programming logic:

1. **Understand the problem**: Read and understand the problem statement.
2. **Generate Examples**: Create additional input and output cases for each problem.
3. **Draw observations:** Draw observations and patterns based on the examples you created.
4. **Start with Basic**: First, think of the basic way to solve the problem then optimize the solution.

**Basic Problems**

* [Check Even or Odd](https://www.geeksforgeeks.org/check-whether-given-number-even-odd/" \t "https://www.geeksforgeeks.org/logic-building-problems/_blank)
* [Multiplication Table](https://www.geeksforgeeks.org/program-to-print-multiplication-table-of-a-number/" \t "https://www.geeksforgeeks.org/logic-building-problems/_blank)
* [Sum of Naturals](https://www.geeksforgeeks.org/program-find-sum-first-n-natural-numbers/" \t "https://www.geeksforgeeks.org/logic-building-problems/_blank)
* [Sum of Squares of Naturals](https://www.geeksforgeeks.org/sum-of-squares-of-first-n-natural-numbers/" \t "https://www.geeksforgeeks.org/logic-building-problems/_blank)
* [Swap Two Numbers](https://www.geeksforgeeks.org/swap-two-numbers/" \t "https://www.geeksforgeeks.org/logic-building-problems/_blank)
* [Closest Number](https://www.geeksforgeeks.org/find-number-closest-n-divisible-m/" \t "https://www.geeksforgeeks.org/logic-building-problems/_blank)
* [Dice Problem](https://www.geeksforgeeks.org/the-dice-problem/" \t "https://www.geeksforgeeks.org/logic-building-problems/_blank)
* [Nth Term of AP](https://www.geeksforgeeks.org/nth-term-of-ap-from-first-two-terms/" \t "https://www.geeksforgeeks.org/logic-building-problems/_blank)

**Easy Problems**

* [Sum of Digits](https://www.geeksforgeeks.org/program-for-sum-of-the-digits-of-a-given-number/" \t "https://www.geeksforgeeks.org/logic-building-problems/_blank)
* [Reverse Digits](https://www.geeksforgeeks.org/write-a-program-to-reverse-digits-of-a-number/" \t "https://www.geeksforgeeks.org/logic-building-problems/_blank)
* [Prime Testing](https://www.geeksforgeeks.org/introduction-to-primality-test-and-school-method/" \t "https://www.geeksforgeeks.org/logic-building-problems/_blank)
* [Check Power](https://www.geeksforgeeks.org/check-if-a-number-is-power-of-another-number/" \t "https://www.geeksforgeeks.org/logic-building-problems/_blank)
* [Distance between Two Points](https://www.geeksforgeeks.org/program-calculate-distance-two-points/" \t "https://www.geeksforgeeks.org/logic-building-problems/_blank)
* [Valid Triangle](https://www.geeksforgeeks.org/check-whether-triangle-valid-not-sides-given/" \t "https://www.geeksforgeeks.org/logic-building-problems/_blank)
* [Overlapping Rectangles](https://www.geeksforgeeks.org/find-two-rectangles-overlap/" \t "https://www.geeksforgeeks.org/logic-building-problems/_blank)
* [Factorial of a Number](https://www.geeksforgeeks.org/program-for-factorial-of-a-number/" \t "https://www.geeksforgeeks.org/logic-building-problems/_blank)
* [Pair Cube Count](https://www.geeksforgeeks.org/count-pairs-a-b-whose-sum-of-cubes-is-n-a3-b3-n/" \t "https://www.geeksforgeeks.org/logic-building-problems/_blank)
* [GCD or HCF](https://www.geeksforgeeks.org/program-to-find-gcd-or-hcf-of-two-numbers/" \t "https://www.geeksforgeeks.org/logic-building-problems/_blank)
* [LCM of Two Numbers](https://www.geeksforgeeks.org/program-to-find-lcm-of-two-numbers/" \t "https://www.geeksforgeeks.org/logic-building-problems/_blank)
* [Perfect Number](https://www.geeksforgeeks.org/perfect-number/" \t "https://www.geeksforgeeks.org/logic-building-problems/_blank)
* [Add Two Fraction](https://www.geeksforgeeks.org/program-to-add-two-fractions/" \t "https://www.geeksforgeeks.org/logic-building-problems/_blank)
* [Fizz Buzz](https://www.geeksforgeeks.org/fizz-buzz-implementation/" \t "https://www.geeksforgeeks.org/logic-building-problems/_blank)
* [Day of the Week](https://www.geeksforgeeks.org/find-day-of-the-week-for-a-given-date/" \t "https://www.geeksforgeeks.org/logic-building-problems/_blank)
* [Nth Fibonacci Number](https://www.geeksforgeeks.org/program-for-nth-fibonacci-number/" \t "https://www.geeksforgeeks.org/logic-building-problems/_blank)
* [Decimal to Binary](https://www.geeksforgeeks.org/program-decimal-binary-conversion/" \t "https://www.geeksforgeeks.org/logic-building-problems/_blank)
* [N-th term of 1, 3, 6, 10, 15, 21…](https://www.geeksforgeeks.org/find-nth-term-series-136101521/" \t "https://www.geeksforgeeks.org/logic-building-problems/_blank)
* [Armstrong Number](https://www.geeksforgeeks.org/program-for-armstrong-numbers/" \t "https://www.geeksforgeeks.org/logic-building-problems/_blank)
* [Palindrome Number](https://www.geeksforgeeks.org/check-if-a-number-is-palindrome/" \t "https://www.geeksforgeeks.org/logic-building-problems/_blank)
* [Digit Root](https://www.geeksforgeeks.org/digital-rootrepeated-digital-sum-given-integer/" \t "https://www.geeksforgeeks.org/logic-building-problems/_blank)

**Medium Problems**

* [Square Root](https://www.geeksforgeeks.org/square-root-of-an-integer/" \t "https://www.geeksforgeeks.org/logic-building-problems/_blank)
* [3 Divisors](https://www.geeksforgeeks.org/numbers-exactly-3-divisors/" \t "https://www.geeksforgeeks.org/logic-building-problems/_blank)
* [Divisible by 4](https://www.geeksforgeeks.org/check-large-number-divisible-4-not/" \t "https://www.geeksforgeeks.org/logic-building-problems/_blank)
* [Divisibility by 11](https://www.geeksforgeeks.org/check-large-number-divisible-11-not/" \t "https://www.geeksforgeeks.org/logic-building-problems/_blank)
* [Divisibility by 13](https://www.geeksforgeeks.org/check-large-number-divisible-13-not/)
* [K-th Digit in a^b](https://www.geeksforgeeks.org/k-th-digit-raised-power-b/" \t "https://www.geeksforgeeks.org/logic-building-problems/_blank)
* [Fraction to Recurring Decimal](https://www.geeksforgeeks.org/represent-the-fraction-of-two-numbers-in-the-string-format/" \t "https://www.geeksforgeeks.org/logic-building-problems/_blank)
* [Recurring Sequence in a Fraction](https://www.geeksforgeeks.org/find-recurring-sequence-fraction/" \t "https://www.geeksforgeeks.org/logic-building-problems/_blank)
* [Compute nPr](https://www.geeksforgeeks.org/program-to-calculate-the-value-of-npr/" \t "https://www.geeksforgeeks.org/logic-building-problems/_blank)
* [Compute nCr](https://www.geeksforgeeks.org/program-calculate-value-ncr/" \t "https://www.geeksforgeeks.org/logic-building-problems/_blank)
* [Pascal’s Triangle](https://www.geeksforgeeks.org/pascal-triangle/" \t "https://www.geeksforgeeks.org/logic-building-problems/_blank)
* [All Factor (Or Divisors)](https://www.geeksforgeeks.org/find-all-factors-of-a-natural-number/" \t "https://www.geeksforgeeks.org/logic-building-problems/_blank)
* [Prime Factorization](https://www.geeksforgeeks.org/prime-factor/" \t "https://www.geeksforgeeks.org/logic-building-problems/_blank)
* [Largest Prime factor](https://www.geeksforgeeks.org/find-largest-prime-factor-number/" \t "https://www.geeksforgeeks.org/logic-building-problems/_blank)
* [Modular Exponentiation](https://www.geeksforgeeks.org/modular-exponentiation-power-in-modular-arithmetic/" \t "https://www.geeksforgeeks.org/logic-building-problems/_blank)
* [nth Catalan Number](https://www.geeksforgeeks.org/program-nth-catalan-number/" \t "https://www.geeksforgeeks.org/logic-building-problems/_blank)
* [Binomial Coefficient](https://www.geeksforgeeks.org/binomial-coefficient-dp-9/" \t "https://www.geeksforgeeks.org/logic-building-problems/_blank)
* [Power Set](https://www.geeksforgeeks.org/power-set/" \t "https://www.geeksforgeeks.org/logic-building-problems/_blank)
* [Next Permutation](https://www.geeksforgeeks.org/next-permutation/" \t "https://www.geeksforgeeks.org/logic-building-problems/_blank)

**Hard Problems**

* [Sieve of Eratosthenes](https://www.geeksforgeeks.org/sieve-of-eratosthenes/" \t "https://www.geeksforgeeks.org/logic-building-problems/_blank)
* [Clock Angle Problem](https://www.geeksforgeeks.org/calculate-angle-hour-hand-minute-hand/" \t "https://www.geeksforgeeks.org/logic-building-problems/_blank)
* [Tower of Hanoi](https://www.geeksforgeeks.org/c-program-for-tower-of-hanoi/" \t "https://www.geeksforgeeks.org/logic-building-problems/_blank)
* [Rat and Poisoned](https://www.geeksforgeeks.org/rat-and-poisoned-bottle-problem/" \t "https://www.geeksforgeeks.org/logic-building-problems/_blank)
* [8 puzzle Problem](https://www.geeksforgeeks.org/8-puzzle-problem-using-branch-and-bound/" \t "https://www.geeksforgeeks.org/logic-building-problems/_blank)
* [Determinant of a Matrix](https://www.geeksforgeeks.org/determinant-of-a-matrix/" \t "https://www.geeksforgeeks.org/logic-building-problems/_blank)
* [Euler's Totient Function](https://www.geeksforgeeks.org/eulers-totient-function/" \t "https://www.geeksforgeeks.org/logic-building-problems/_blank)
* [Josephus Problem](https://www.geeksforgeeks.org/josephus-problem/" \t "https://www.geeksforgeeks.org/logic-building-problems/_blank)