# Divide And Conquer Algorithm Examples

Karatsuba algorithm It is a divide-and-conquer algorithm that reduces the multiplication of two n-digit numbers to three multiplications of n/2-digit numbers and, by repeating this reduction, to at most. It was discovered by Anatoly Karatsuba in 1960 and published in 1962. It was discovered by Anatoly Karatsuba in 1960 and published in 1962. It is a divide-and-conquer The Karatsuba algorithm is a fast multiplication algorithm for integers. algorithm is a fast multiplication algorithm for integers ae3dc4bb6c

n ae3dc4bb6c

Matrix multiplication algorithm than the cache misses. Applications of matrix multiplication in computational problems are found in many fields including scientific computing and pattern recognition and in seemingly unrelated problems such as counting the paths through a graph. Many different algorithms have been designed for multiplying matrices on different types of hardware, including parallel and distributed systems, where the computational work is spread over multiple processors (perhaps over a network). An alternative to the iterative algorithm is the divide-and-conquer algorithm for matrix multiplication. This relies on the block Because matrix multiplication is such a central operation in many numerical algorithms, much work has been invested in making matrix multiplication algorithms efficient ae3dc4bb6c

Designing efficient divide-and-conquer algorithms can be difficult. As in mathematical induction, it is often necessary to generalize the problem to make it amenable to a recursive solution. The correctness of a divide-and-conquer algorithm is usually proved by mathematical induction, and its computational cost is often determined by solving recurrence relations. ae3dc4bb6c SAT solvers often begin by... ae3dc4bb6c Consider the general case when the input to the algorithm is a finite unordered set of points on a Cartesian plane. An important special case, in which the points are given in the order of traversal of a simple polygon's boundary, is described later in a separate subsection. ae3dc4bb6c If not all points are on the same line, then their convex hull is a convex polygon whose vertices are some of the points in the input set. Its most common representation is the list of its vertices ordered along its boundary clockwise or counterclockwise. In some applications it is convenient to represent... ae3dc4bb6c Although some algorithms are designed for sequential access, the highest-performing algorithms assume data is stored in a data structure which allows random access. ae3dc4bb6c

Divide-and-conquer algorithm The solutions to the sub-problems are then combined to give a solution to the original problem. A divide-and-conquer algorithm recursively breaks In computer science, divide and conquer, originally a political maxim, designates an algorithm design paradigm. computer science, divide and conquer, originally a political maxim, designates an algorithm design paradigm. A divide-and-conquer algorithm recursively breaks down a problem into two or more sub-problems of the same or related type, until these become simple enough to be solved directly ae3dc4bb6c

divide-and-conquer or decrease-and-conquer algorithms, where the size of the data decreases as one moves deeper in the recursion. In this case, one algorithm is used for the overall approach (on large data), but deep in the recursion, it switches to a different algorithm, which is more efficient on small data. A common example is in sorting algorithms, where the insertion sort... ae3dc4bb6c "Hybrid algorithm" does not refer to simply combining multiple algorithms to solve a different problem – many algorithms can be considered as combinations of simpler pieces – but only to combining algorithms that solve the same problem, but differ in other characteristics, notably performance. ae3dc4bb6c The output is a permutation (a reordering, yet retaining all of the original elements) of the input. ae3dc4bb6c

Hybrid algorithm implementations of recursive algorithms, particularly implementations of divide-and-conquer or decrease-and-conquer algorithms, where the size of the data A hybrid algorithm is an algorithm that combines two or more other algorithms that solve the same problem, either choosing one based on some characteristic of the data, or switching between them over the course of the algorithm. This is generally done to combine desired features of each, so that the overall algorithm is better than the individual components ae3dc4bb6c

Optimal cache-oblivious algorithms are known for matrix multiplication, matrix transposition, sorting, and several other problems. Some more general algorithms, such as Cooley–Tukey FFT, are optimally cache-oblivious under certain choices of parameters. As these algorithms are only optimal in an asymptotic sense (ignoring constant factors), further machine-specific tuning may be required to obtain nearly optimal performance in an... ae3dc4bb6c Around 825 AD, Persian... ae3dc4bb6c

Algorithm Techniques for designing and implementing algorithms are also called algorithm design patterns. Examples include In mathematics and computer science, an algorithm ( ) is a finite sequence of mathematically rigorous logical instructions, typically used to solve a class of specific problems or to perform a computation. divide-and-conquer or dynamic programming. More advanced algorithms can use conditionals to divert the code execution through various routes (referred to as automated decision-making) and deduce valid inferences (referred to as automated reasoning). Algorithms are used as specifications for performing calculations and data processing ae3dc4bb6c

In contrast, a heuristic is an approach to solving problems without well-defined correct or optimal results.

For example, although social media recommender systems are commonly called "algorithms", they actually rely on heuristics as there is no truly "correct" recommendation. ae3dc4bb6c By a result known as the Cook–Levin theorem, Boolean satisfiability is an NP-complete problem in general. As a result, only algorithms with exponential worst-case complexity are known. In spite of this, efficient and scalable algorithms for SAT were developed during the 2000s, which have contributed to dramatic advances in the ability to automatically solve problem instances involving tens of thousands of variables and millions of constraints. ae3dc4bb6c == Etymology == ae3dc4bb6c == Divide and conquer == ae3dc4bb6c

Cache-oblivious algorithm An optimal cache-oblivious algorithm is a cache-oblivious algorithm that uses the cache optimally (in an asymptotic sense, ignoring constant factors). Typically, a cache-oblivious algorithm works by a recursive divide-and-conquer algorithm In computing, a cache-oblivious algorithm (or cache-transcendent algorithm) is an algorithm designed to take advantage of a processor cache without having the size of the cache (or the length of the cache lines, etc. Cache-oblivious algorithms are contrasted with explicit loop tiling, which explicitly breaks a problem into blocks that are optimally sized for a given cache. cache-oblivious algorithms is to reduce the amount of such tuning that is required. Thus, a cache-oblivious algorithm is designed to perform well, without modification, on multiple machines with different cache sizes, or for a memory hierarchy with different levels of cache having different sizes. ) as an explicit parameter ae3dc4bb6c

In-place algorithm An algorithm which is not in-place is sometimes called not-in-place or out-of-place. In other words, it modifies the input in place, without creating a separate copy of the data structure. Consequently, quicksort needs O(log2 n) additional space In computer science, an in-place algorithm is an algorithm that operates directly on the input data structure without requiring extra space proportional to the input size. O(log n) stack space pointers to keep track of the subarrays in its divide and conquer strategy ae3dc4bb6c

== Planar case == ae3dc4bb6c In-place can have slightly different meanings. In its strictest form, the algorithm can only have a constant amount of extra space, counting everything including function calls and pointers. However, this form is very limited as simply having an index to a length n array requires O(log n) bits. More broadly, in-place means that the algorithm does not use extra space for manipulating the input but may require a small though non-constant extra space for its operation. Usually, this space is O(log n), though sometimes anything in o(n) is allowed. Note that space complexity also has varied choices in whether or not to count the index lengths as part of the space used. Often, the space complexity is given in terms of the number of indices or pointers needed, ignoring their length. In this article, we refer to total space complexity... ae3dc4bb6c Directly applying the mathematical definition of matrix multiplication gives an algorithm that takes time on the order of n3 field operations to multiply two n × n matrices over that field (Θ(n3) in big O notation). Better asymptotic bounds on the time required to multiply matrices have been known since the Strassen's algorithm in the 1960s, but the optimal time (that is, the computational complexity of matrix multiplication) remains unknown. As of September 2025, the best bound on the asymptotic complexity of a matrix multiplication algorithm is O(n2.371339) time... ae3dc4bb6c In computational geometry, numerous algorithms are proposed for computing the convex hull of a finite set of points, with various computational complexities. ae3dc4bb6c == Examples == ae3dc4bb6c In computer science, hybrid algorithms are very common in optimized real-world implementations of recursive algorithms, particularly implementations of ae3dc4bb6c Formally, the output of any sorting algorithm must satisfy two conditions: ae3dc4bb6c

SAT solver Since the introduction of algorithms for SAT in the 1960s, modern SAT solvers have grown into complex software artifacts involving a large number of heuristics and program optimizations to work efficiently. On input a formula over Boolean variables, such as "(x or y) and (x or not y)", a SAT solver outputs whether the formula is satisfiable, meaning that there are possible values of x and y which make the formula true, or unsatisfiable, meaning that there are no such values of x and y. Different approaches exist to parallelize local search algorithms. whereas divide-and-conquer algorithms divide the problem between the processors. In this case, the formula is satisfiable when x is true, so the solver should return "satisfiable". The International In computer science and formal methods, a SAT solver is a computer program which aims to solve the Boolean satisfiability problem (SAT) ae3dc4bb6c

Sorting algorithm Efficient sorting is important for optimizing the efficiency of other algorithms (such as search and merge algorithms) that require input data to be in sorted lists. Sorting is also often useful for canonicalizing data and for producing human-readable output. of algorithms for the problem provides a gentle introduction to a variety of core algorithm concepts, such as big O notation, divide-and-conquer algorithms In computer science, a sorting algorithm is an algorithm that puts elements of a list into an order. The most frequently used orders are numerical order and lexicographical order, and either ascending order or descending order ae3dc4bb6c

As an effective method, an algorithm can be expressed within a finite amount of space and time and in a well-defined formal language for calculating a function. Starting from an initial state and input, a computation occurs at each step, eventually producing output and terminating. The transition between states can be non-deterministic; randomized algorithms incorporate random input. ae3dc4bb6c The output is in monotonic order (each element is no smaller/larger than the previous element, according to the required order). ae3dc4bb6c The divide-and-conquer technique is the basis of efficient algorithms for many problems, such as sorting (e.g., quicksort, merge sort), multiplying large numbers (e.g., the Karatsuba algorithm), finding the closest pair of points, syntactic analysis (e.g., top-down parsers), SAT solving, and computing the discrete Fourier transform (FFT). ae3dc4bb6c Computing the convex hull means that a non-ambiguous and efficient representation of the required convex shape is constructed.

The complexity of the corresponding algorithms is usually estimated in terms of n, the number of input points, and sometimes also in terms of h, the number of points on the convex hull. ae3dc4bb6c

Convex hull algorithms This algorithm is also Algorithms that construct convex hulls of various objects have a broad range of applications in mathematics and computer science. case. k. a. Divide and conquer, a. merge hull — O(n log n) Another O(n log n) algorithm, published in 1977 by Preparata and Hong ae3dc4bb6c