

PAPER • OPEN ACCESS

Research on the Implementation of Recursive Algorithm Based on C Language

To cite this article: Zhigang Zhu and Wei Sun 2021 *J. Phys.: Conf. Ser.* **2023** 012015

View the [article online](#) for updates and enhancements.



240th ECS Meeting

Digital Meeting, Oct 10-14, 2021

We are going fully digital!

Attendees register for free!

REGISTER NOW



Research on the Implementation of Recursive Algorithm Based on C Language

Zhigang Zhu^{1,*}, Wei Sun²

¹City Institute Dalian University of Technology ,Liaoning, Dalian, 116600, China

*Corresponding author e-mail: zzg@dlut.edu.cn

Abstract. In the utilization of C language, it often appears that the function calls itself directly or indirectly, that is, the function calls itself recursively. The recursive solution of the same kind of problem can be expressed by recursion, and the specific problem can be reduced by recursion. Based on this, this paper first analyses the concept and function of C language recursion, then studies the process description of recursion algorithm, and finally gives the implementation strategy of C language recursion algorithm.

Keywords: Recursive Algorithm, C Language, Implementation

1. Introduction

With the iterative progress and maturity of computer tech, it has been widely and deeply studied and popularized in many fields, especially with the continuous progress of computer software language, and the computer language represented by C language has been popularized rapidly. In the utilization of computer C language, it often appears that the function calls itself directly or indirectly, that is, the function calls itself recursively [1]. As a typical feature of computer C language, recursive function call can effectively reduce the sub-problems of the same kind of problems, and use recursive function call to realize the intuitive expression of the solution of specific problems. Compared with loop statements, C language recursion is more efficient and can solve some practical problems.

The recursion algorithm of C language includes not only recursion of recursion call, but also one-step analysis of statement in recursion process [2]. The difficulty of recursive call is that its algorithm process is abstract, so it is difficult to solve practical problems. The key of C language recursive algorithm lies in the following several key processes as shown in Figure 1, so as to further carry out the specific recursive practical utilization.

Recursive algorithm can effectively solve the calculation and judgment problems with the characteristics of iteration, loop and nesting, so as to effectively carry out the specific practical utilization [3]. Specific problems such as computer language grammar and lexical analysis, mathematical recursive function calculation and database access and other specific issues. Computer C language benefits from its powerful functions, so it can be applied in many environments such as system development, software utilization and control programming [4]. C language code is concise, efficient, flexible and rich, which makes it possible to realize the advanced utilization of recursive program.



In addition, the recursive algorithm of C language can solve many practical problems such as code organization and readability, memory stack overflow and so on, so as to effectively solve the problems of system overhead, space-time complexity control and so on. Reasonable utilization of recursive algorithm can achieve the reliability and readability of the problem solving process [5]. Only by constructing a scientific, reasonable and safe C language recursive algorithm, can we effectively solve the problem of actual algorithm requirements. In a word, as a tool of algorithm description and design, C language recursion is often used to describe and process complex algorithms. Therefore, the research of recursive algorithm based on C language has important practical value.

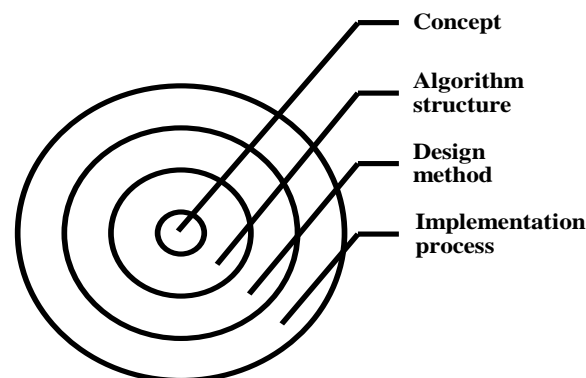


Figure 1. The key process of C language recursive algorithm

2. The concept and function of C language recursion

2.1. The concept of C language recursion

C language allows nested function calls, that is, in the process of calling a function, another function is called. There is a special case of nested function call, that is, recursive call [6]. That is to say, in the process of calling a function, there is a direct or indirect call to the function itself. The concept or function of C language recursion directly or indirectly refers to itself, which plays an important function in computability theory, and recursion can be enumerated. The commonly used important mechanisms include the expression of concepts, the description of data structures and algorithms, and important ways of thinking. In modern programming language, especially C language, it supports recursion.

2.2. Functions and types of recursive algorithms

In recursive algorithm, sort is divided into merge sort and quick sort. Among them, the most typical implementation method of the former is defined by recursion, while the latter is defined directly by recursion [7]. In the process of writing recursive program algorithm, it should first solve the common function call in syntax. Secondly, at the algorithm level, find the recursive form and recursive boundary. In addition, recursive algorithm can be divided into two common types: recursive algorithm based on divide and conquer strategy and recursive algorithm based on Backtracking strategy. Recursive algorithm has many advantages, such as clear concept, easy to understand, simple description, easy implementation, compact code, easy maintenance and so on.

2.3. Difficulties in using recursive algorithm

Although recursive algorithm has many advantages, it still has some problems, such as high computational complexity in some cases, repeated computation caused by improper definition and so on [8]. And in some cases, it takes up more storage space, resource consumption caused by deep recursive call, the cost of function call, and the proportion of function call cost increases when the calculation process is simple. In addition, the difficulties of recursion mainly focus on the basic

thinking method of recursion, the expression of recursion concept, the description method of recursion process, the execution process of recursion, the use condition and environment of recursion.

3. Process description of recursive algorithm

3.1. The basic idea of recursive algorithm process description

The recursive algorithm process turns the problem into the same form but smaller scale problem, which is solved when the problem scale is reduced to a certain extent. Among them, in the recursive description level, by defining the situation and method that can solve the problem directly, the general solving process of the problem is described by self-reference [9]. The basic idea of recursive process description is similar to mathematical induction, and reduces the complexity of the problem in the process of quoting itself, and directly solves the problem when the complexity is reduced to a certain extent.

3.2. Description steps and basic structure of recursive process

The first step to describe the recursive process is to determine the recursive parameters, define the termination conditions and basic calculation of recursion. When the recursive parameter is a certain value, it should be calculated directly. Secondly, in the level of defining recursive call, when the recursive parameter does not meet the termination condition, the calculation is represented as the calculation including the call to itself, and the recursive parameter should be closer to the termination condition when calling to itself. In addition, the basic structure of recursive function mainly includes basic calculation and recursive call, as shown in Table 1 below.

Table 1. The basic structure of recursive function

Recursive entry	Basic structure
Basic calculation	Basic condition of recursion
	Basic calculation of recursion
	Direct or indirect reference to oneself
Recursive call	Modification of recursive control parameters
	Change in the direction of recursive termination condition
	Other calculations in recursive call

3.3. The execution process of recursive function

The execution process of recursive function needs to focus on the execution process of calling itself when the function is not defined, as well as the change of function parameters and local variable values. Secondly, at the execution level of recursive function, it is the same as ordinary function call. The actual call occurs at the execution stage. Each recursive call generates an independent function running instance and has independent variables and parameters with each function running instance. Therefore, the execution process of recursive function can be regarded as calling another function with the same name and code.

3.4. Design and efficiency of recursive function

The main problems of recursive function design and program design are analyzed. The former is mainly to determine the recursive parameters in the problem, to determine the lower limit of the parameter range and its calculation method, and to determine the relationship between the general situation and the simple situation [10]. The latter is mainly to refine the recursive description of the problem, determine the factors to be considered in the implementation of the algorithm and select the appropriate data structure. In terms of the efficiency of recursive function, the extra cost of recursive function mainly includes function call, parameter transfer, saving of call environment and direction of instruction flow. Recursive function is mainly called itself, the number of function calls may be more, and the level of nested function calls may be deeper.

In addition, the main factors that affect the execution efficiency of recursive function include too large proportion of function calling process, too many simple function body parameters, incorrect use of recursion and repeated calculation caused by recursion definition.

4. Implementation of recursive algorithm in C language

4.1. Implementation of recursive algorithm based on divide and conquer strategy

The core idea of recursive algorithm based on divide and conquer strategy is to divide the problem into several sub-problems, deal with each sub-problem separately in the same way, and then synthesize the results of each sub-problem to form the final result. To realize the recursive algorithm program based on divide and conquer strategy, it should first establish the thinking mode of divide and conquer recursion in algorithm analysis. Secondly, it should establish recursion confidence in programming. The basic principle of establishing the thinking mode of divide and conquer recursion is target driven. The recursion process of calling and returning is shown in Figure 2.

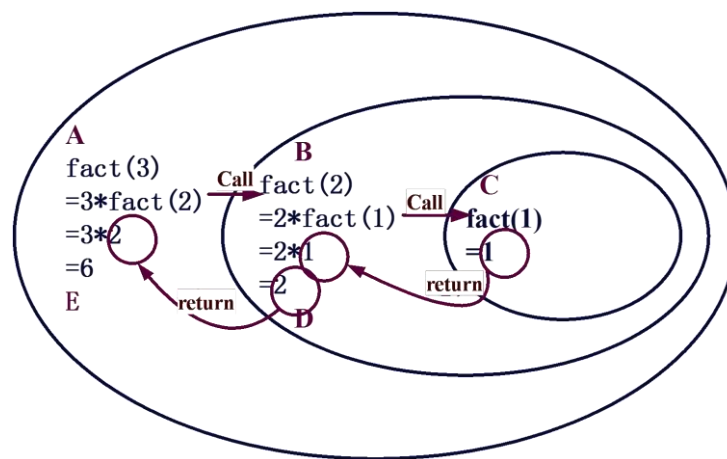


Figure 2. Call and return the recursive process

In addition, the realization of recursive algorithm based on divide and conquers strategy should also find the maximum value; determine the recursive form and recursive boundary of recursive algorithm, so as to write the corresponding recursive function.

4.2. Implementation of recursive algorithm based on backtracking strategy

Recursive algorithm is an important method of recursive algorithm. Recursive algorithm based on Backtracking strategy is mainly used to solve the optimal solution in program design according to the search tech of trial and backtracking. Through the process of traversing the state tree in advance, the state tree is not established in advance, but implied in the process of traversing. The recursive algorithm based on Backtracking strategy is helpful to the analysis and expression of problems, and often occurs in the inductive thinking process. When used in appropriate problems, it is easy to implement, short code, less errors, and can effectively avoid abuse.

4.3. Utilization and efficiency improvement of recursive algorithm

Recursive algorithm is suitable for the operation of recursively defined data structure, such as traversal of tree; and problems described directly by recursive method, such as Hanoi tower. Secondly, it is also applied to the depth first search, so as to solve the problem of recursive algorithm which can meet the efficiency requirements in the conventional computable range. But the recursive algorithm is not suitable for the problem of simple non recursive algorithm, the algorithm with too deep recursive depth and the algorithm with too much repeated calculation and low efficiency. In addition, in order to improve the efficiency of recursive function, we save the calculated value to avoid unnecessary

repeated calculation, and use tail recursive function and simple recursive function to realize non recursive function.

5. Conclusion

In summary, Computer C language benefits from its powerful functions, so it can be applied in many environments such as system development, software utilization and control programming. C language code is concise, efficient, flexible and rich, which makes it possible to realize the advanced utilization of recursive program. These papers studies the concept and function of C language recursion, analyzes the algorithm, function and type of recursion, the design and efficiency of recursion function. By analyzing the process description of recursive algorithm, the description steps and basic structure of recursive process are studied. Through the research on the implementation of recursive algorithm in C language, this paper analyzes the implementation and efficiency optimization of recursive algorithm based on divide and conquers strategy and backtracking strategy.

References

- [1] Chen Baoping. Parameter setting of recursive algorithm [J]. Modern computer monthly, 2011, 13:3-5.
- [2] Chen Dexiang. Discussion on recursive function teaching in C language [J]. China Science and tech information. 2013 (07): 193-195.
- [3] Gao Si, Zhao Bo. Implementation of recursive algorithm in C language [J]. Digital tech and utilization, 2015, 7:140-141.
- [4] Huang Chunyan. Exploration and utilization of recursive method in C language project practice [J]. Computer programming skills and maintenance, 2014, 4:95-96.
- [5] Li Wei. Analysis of C language recursive algorithm [J]. Computer knowledge and tech. 2012 (30): 7229-7235.
- [6] Wang Zhanquan, Wen Xinxiu, Yang Zeping, Liu Jiang, Gu Chunhua. Research and construction of computer teaching in chemical engineering based on Computational Thinking and innovative practice [J]. Chemical higher education, 2017 (3): 31-37.
- [7] Wen Haiying, Chen Youming, Lu Lanlan. Exploration of teaching method of "C language programming" based on the cultivation of Computational Thinking Ability [J]. Computer knowledge and tech, 2019 (32): 139-141.
- [8] Yin Na. Discussion on recursion in C language [J]. China e-commerce. 2012 (6): 189-190.
- [9] Yu Jia. On the utilization of recursion and recursive algorithm in program design [J]. Computer programming skills and maintenance, 2014.12:7-8.
- [10] Zhang Yulin, Zhang Yuan. Research and practice of C programming case teaching method -- Enlightenment of guiding college students to participate in the national software design competition [J]. Computer CD software and utilization, 2019 (8): 288-289.