直击招聘

533道面试真题

直击招聘

程序员面试笔试
数据结构深度解析

◎ 李春葆 李筱驰 编著

定位准确 面向企业应聘人才，面向编程技术提高者。

答疑解惑 解析相关课程中难点、疑点和热点，涉及目前各大网站上热门讨论的话题。

实战性强 收集近些年笔试和面试题目，涵盖常见考点。

清華大學出版社

直击招聘

直击招聘

程序员面试笔试
数据结构深度解析

◎ 李春葆 李筱驰 编著

清华大学出版社
北京

内 容 简 介

本书汇集国内外众多著名 IT 企业近几年的数据结构面试笔试真题并予以解析，按知识点类型对常见的数据结构难点和疑点进行了系统归纳和透彻剖析，并提供了一定数量的自测题以便于读者自我检验。

全书逻辑清晰、通俗易懂，适合参加 IT 企业校园招聘和面试笔试环节的同学复习，也适合数据结构和算法设计编程爱好者以及在校学生阅读和提高。

图书在版编目（CIP）数据

直击招聘：程序员面试笔试数据结构深度解析 / 李春葆，李筱驰编著. —北京：清华大学出版社，2018
（直击招聘）

ISBN 978-7-302-48881-1

Ⅰ. ①程… Ⅱ. ①李… ②李… Ⅲ. ①数据结构-资格考试-自学参考资料 Ⅳ. ①TP311.12

中国版本图书馆 CIP 数据核字（2017）第 287724 号

责任编辑：魏江江　王冰飞
封面设计：杨　兮
责任校对：李建庄
责任印制：李红英

出版发行：清华大学出版社
网　　址：http://www.tup.com.cn, http://www.wqbook.com
地　　址：北京清华大学学研大厦 A 座　　邮　　编：100084
社 总 机：010-62770175　　邮　　购：010-62786544
投稿与读者服务：010-62776969，c-service@tup.tsinghua.edu.cn
质 量 反 馈：010-62772015，zhiliang@tup.tsinghua.edu.cn
课 件 下 载：http://www.tup.com.cn，62795954
印 装 者：清华大学印刷厂
经　　销：全国新华书店
开　　本：185mm×240mm　　**印　张：**29　　**字　　数：**632 千字
版　　次：2018 年 6 月第 1 版　　**印　　次：**2018 年 6 月第 1 次印刷
印　　数：1～2000
定　　价：89.80 元

产品编号：077573-01

出版说明

国内外许多著名的 IT 企业都采用“企业网申→测评→笔试→面试→发放录用意向书”的招聘流程，如阿里巴巴校园招聘网站“https://campus.alibaba.com/process.htm”发布了详细的招聘信息，申报各种技术类岗位（如研发工程师、算法工程师、前端开发工程师和测试开发工程师等）的同学必须经过笔试和面试环节。

尽管不同 IT 公司的笔试和面试方式存在差异，但考查点基本相同，除了语言表达能力、自我控制能力、人际关系处理能力和生活情趣等非工作技能外，从专业角度看，无非就是如下几点。

❶ 基本知识点掌握情况：面试者至少需要熟练掌握一门计算机语言，常用的有 C/C++、Java 等，这是作为程序员的基本功。所谓“熟练掌握”，就是不仅限于基本语法，还要理解语言的实现与运行时情况，如 C 语言中的指针、变量存储类别和函数执行过程等。

❷ 基本程序设计情况：考查面试者是否具有编程能力，包含编写代码、调试代码和测试代码的整个过程，能够做到“让代码说话”，不能只会夸夸其谈，却不会编程。

❸ 算法设计能力：考查面试者求解问题的算法设计和实现能力，采用的算法策略是否合适，算法是否涵盖所有的测试用例。

❹ 计算逻辑思维能力：考查面试者是否具有一定的分析问题和解决问题的能力，用计算机求解问题时思维是否缜密，能否抓住问题的本质，采用的思路是否得当，条理是否清晰。

本丛书是面向参加笔试和面试的同学编写的，不仅对接大学计算机课程体系，而且结合 IT 企业招聘人才的基本行规，涵盖的内容如下：

- 程序员面试笔试 C 语言深度解析。
- 程序员面试笔试 C++语言深度解析。
- 程序员面试笔试数据结构深度解析。
- 程序员面试笔试算法设计深度解析。

本丛书以 C/C++语言为工具、数据结构为基础、算法设计为目标，其中《直击招聘——程序员面试笔试 C 语言深度解析》和《直击招聘——程序员面试笔试 C++语言深度解析》侧重 C/C++语言的核心概念和深层次用法，《直击招聘——程序员面试笔试数据结构深度解析》侧重以常用数据结构为核心的算法设计，《直击招聘——程序员面试笔试算法设计深度解析》侧重通用的算法设计。本丛书具有如下特点。

❶ 定位准确：面向企业应聘人才，面向编程技术提高者。

❷ 答疑解惑：解析相关课程中的难点、疑点和热点，许多都是目前各大网站上热门讨论的话题。

❸ 实战性强：收集近些年的笔试和面试题目，涵盖常见考点。

前　言

数据结构求解问题的思路是“数据逻辑结构→存储结构→基本算法实现→应用”，这一思路展示了计算逻辑思维，也就是用计算机求解问题的基本过程。

编程的第一步需要理解问题本身，提炼出数据逻辑结构和相关运算；然后实现数据的机内表示，也就是数据的存储结构设计，好的存储结构设计会达到事半功倍的效果；最后在存储结构上实现数据的运算，即算法实现。

常用的数据结构有线性表、栈、队列、串、树、二叉树和图等，除了围绕这些数据结构的基本运算算法设计外，还包含查找和排序算法设计。

在面试笔试中数据结构的考点主要包含两个方面：一是常用数据结构的基本知识点，包括各种数据结构的逻辑特点、存储方式和运算算法，如一个城市图的存储、在城市图中查找两个城市之间的最短路径等；二是常用数据结构的应用知识点，能够熟练地利用数据结构解决问题，如用栈或者队列求解迷宫问题，用栈求解皇后问题等。

很多数据结构都是递归数据结构，递归也是求解问题的基本方法，所以面试者必须具有递归算法设计能力，掌握从递归模型、递归算法执行过程到递归算法设计的一般方法，为二叉树、图等复杂数据结构算法设计打下坚实的基础。

本书系统归纳了数据结构常见的知识要点，汇集国内外众多著名 IT 企业近几年的数据结构面试笔试真题并予以解析，透彻地剖析了难点和疑点，每道面试题给出了难度标识，从一星到五星难度依次递增。

在本书的编写过程中参考了众多网站和博客，无法一一列出，在此编者表示衷心感谢。

限于编者水平，书中难免存在遗漏，恳请读者批评指正，编者的联系邮箱是licb1964@126.com。

编　者

2018 年 3 月

目 录 ▷▷▷

·第 1 章·
数据结构基础

常见考点

- 数据结构的相关概念。
- 算法的 5 个特性。
- 算法描述方法。
- 算法分析方法。
- 递归的定义和递归模型。
- 递归算法设计的一般方法。
- 灵活运用递归算法解决一些较复杂的应用问题。

1.1 数据结构的概念

1.1.1 要点归纳

数据结构是指带结构的数据集合，即数据结构=(D,R)，其中，D 是数据元素的集合，R 是 D 中元素的关系的集合，同一个 D 中可能有多种关系。

数据结构中讨论的数据通常是结构化数据，所有 D 中元素的类型相同，其类型通过数据项来描述。相关概念如下：

- 一种数据结构包含逻辑结构、存储结构和数据运算 3 个方面。
- 逻辑结构是 D 中元素逻辑关系的整体，是面向用户的，与计算机无关。
- 存储结构是 D 和 R 在计算机中的表示，是面向程序员的，与计算机相关。
- 数据运算是对 D 实施的操作，分为运算描述和运算实现。
- 抽象数据类型=D 的逻辑结构+运算描述，用于精确地表述求解问题。

从逻辑结构看，所有数据按元素的逻辑关系分为线性结构、树结构和图结构（通常不考虑集合，集合是指 D 中数据元素之间没有关系）。树结构和图结构统称为非线性结构。每种逻辑结构类型具有不同的逻辑结构特点。

从存储结构看，常用的存储结构分为顺序、链式、索引和哈希表。每种存储结构类型具有不同的性能特点。

重要的数据结构观：数据逻辑结构（含运算描述） ⇨ 存储结构 ⇨ 运算实现。

一个数据结构总会提供基本运算。在它实现好后，可以使用它存放数据，也可以使用它的基本运算完成更复杂的功能。实际上，C/C++中的 int 就是一个实现好的数据结构，它可以用于存放整数，还可以使用其基本运算（如+、-、*、/、%）求解问题。

算法是对特定问题求解步骤的一种描述，它是指令的有限序列。在数据结构课程中通常把数据运算的实现称为算法，算法实现一般是在存储结构的基础上给出运算的具体实现过程。算法具有有穷性、确定性、可行性和输入/输出特性。

算法采用计算机语言表述就是程序，但程序不一定满足有穷性。

1.1.2 面试题解析

【面试题 1-1 ★】下列叙述中正确的是（　　）。

A．有一个以上根结点的数据结构不一定是非线性结构

B．只有一个根结点的数据结构不一定是线性结构

C．循环链表是非线性结构

D．双向链表是非线性结构

答：根结点是指没有前驱结点的结点，而开始结点，有一个以上的根结点表示该数据结构的开始结点不唯一，它是一种非线性数据结构。而只有一个根结点的数据结构不一定是线性结构，如树只有一个根结点，属于非线性结构。答案为 B。

【面试题 1-2 ★】下列数据结构能够支持随机的插入和删除操作并具有较好的性能的是（　　）。

A．数组和链表　　B．链表和哈希表　　C．哈希表和队列

D．队列和堆栈　　E．堆栈和双向队列　　F．双向队列和数组

答：这里随机的含义是指任意位置。通常，链式存储结构更好地支持随机插入和删除操作，顺序存储结构在进行插入和删除操作时需要大量移动元素，而数组是一种顺序存储结构，栈和队列（含双向队列）不提供随机插入和删除操作。答案为 B。

【面试题 1-3 ★★★★】已知计算机有以下原子操作：

❶ 赋值操作：b = a

❷ 自增 1：++a

❸ for ( ) { *** }有限循环

❹ 操作数只能为 0 或者正整数

利用这些原子操作定义函数实现加、减和乘操作。

解：本题的设计算法及其测试数据如下（根据题目要求，减法操作的结果是一个非负数）。

```
#include <stdio.h>
int add(int a,int b)          //求 a+b
{   int result=a;
```

```
    for(int i=0;i<b;++i)
        ++result;
    return result;
}
int decone(int n)              //求 n-1
{   int tmp=0;
    int result=0;
    for (int i=0;i<n;++i)
    {   result=tmp;
        ++tmp;
    }
    return result;
}
int sub(int a,int b)           //求 a-b: 只适合 a≥b 的情况
{   int result=a;
    for(int i=0;i<b;++i)
        result=decone(result);
    return result;
}
int mult(int a,int b)          //求 a*b
{   int result=0;
    for (int i=0;i<a;++i)
        for (int j=0;j<b;++j)
            ++result;
    return result;
}
void main()
{   int a=6,b=3;
    printf("%d+%d=%d\n",a,b,add(a,b));      //输出: 6+3=9
    printf("%d-%d=%d\n",a,b,sub(a,b));      //输出: 6-3=3
    printf("%d*%d=%d\n",a,b,mult(a,b));     //输出: 6*3=18
}
```

【面试题 1-4 ★★★★】只允许使用+、赋值语句、判断语句和有限循环实现两个整数的减法、乘法和除法（整除）运算。

解：由提供的操作设计好改变整数符号的 negate()函数和求整数绝对值的 abs()函数。各功能运算设计如下：

❶ a-b=a+negate(b)。

❷ 若 a≥0，a*b=b 累加 a 次；若 a<0，a*b=-(b 累加 abs(a)次)。

❸ x=a/b，即 a=bx，在 a>0 并且 b>0 时，x=累加 b 小于或等于 a 的最多次数。再考虑 a、

b 为负数的其他情况。

对应的算法及其测试数据如下：

```
#include <stdio.h>
int negate(int a)                    //改变整数 a 的符号
{   int neg=0;
    int d=(a<0?1:-1);
    while(a!=0)
    {   neg=neg+d;
        a=a+d;
    }
    return neg;
}
int abs(int a)                       //求 a 的绝对值
{   if(a<0) return negate(a);
    else return a;
}
int sub(int a,int b)                 //求 a-b
{
    return a+negate(b);
}
int mult(int a,int b)                //求 a*b
{   int result=0;
    for(int i=abs(a);i>0;i--)
        result=result+b;
    if(a>0) return result;
    else return negate(result);
}
int div(int a,int b)                 //求 a/b
{   int a1=abs(a);
    int b1=abs(b);
    int product=b1;
    int x=0;
    while(product<=a1)
    {
        product=product+b1;
        x=x+1;
    }
    if((a<0 && b<0) || (a>0 && b>0))
        return x;                    //a、b 符号相同返回 x
    else
```

```
        return negate(x);                          //a、b 符号不同返回-x
}
void main()
{   int a=-5,b=3;
    printf("%d-%d=%d\n",a,b,sub(a, b));      //输出：-5-3=-8
    printf("%d*%d=%d\n",a,b,mult(a, b));     //输出：-5*3=-18
    printf("%d/%d=%d\n",a,b,div(a, b));      //输出：-5/3=-1
}
```

1.2 算法描述和分析

1.2.1 要点归纳

⋙ 1. 算法描述

一个算法用于实现一个运算功能，包含有输入和输出，可以采用 C/C++中的函数来描述，其中输入和输出作为函数的参数，分别称为输入型参数和输出型参数。

如果需要对算法输入进行正确性判断，通常将函数的返回值类型设计成 bool；如果不需要对算法输入进行正确性判断，通常将函数的返回值类型设计成 void。

输出型参数需要将形参值回传给对应的实参，通常设计成引用型参数，这是最简单的方法，采用指针方式也可以将形参回传给实参，但相对复杂一些。有时也可以将算法输出作为返回值，此时将函数类型设计成该返回值的类型。

一些考生在设计算法时经常出现不将算法的输出结果回传给实参的情况，这就像网上答题，辛苦做完题目最后却没有提交一样。认清算法参数，确定哪些是输入型参数，哪些是输出型参数，在输出型参数名称前面加上&，特别是一些参数既是输入型又是输出型的情况，一般像输出型参数那样处理，这是程序员的基本功。

算法设计的一般过程如下：

❶ 分析求解问题的功能，如果是面对面试官，要多提问以理清问题，展示你的沟通能力。

❷ 选择合适的数据结构，展示你的运用数据结构求解问题的能力。

❸ 设计尽可能高效的算法，展示你的算法设计能力。

❹ 编写相应的程序，运行并调试程序直至得到正确的结果，展示你的算法实现能力。

⋙ 2. 算法分析

算法分析是为了改进算法，包括时间分析和空间分析，主要是采用事前估计法，即不

是求算法执行的时间和空间的绝对数量，而是认为算法的时间和空间量是问题规模 n 的函数。

时间分析的基本方法是求出算法中所有原操作的执行次数 T(n)，并认为所有原操作的执行时间相同（实际上，+和*都是原操作，在计算机组成原理课程中可以看出它们的执行时间相差很大，但在这里将它们的执行时间看成是相同的，这说明数据结构课程是从更高抽象层次讨论算法性能的），这样就可以将 T(n)看成是算法的执行时间。

再将 T(n)用复杂度形式表示，通常用 O 表示，即 T(n)=O(f(n))，其含义是存在正常数 c 和 n_0 使得所有 $n \geq n_0$ 有 $0 \leq T(n) \leq cf(n)$。例如 $T(n)=2n^2-5n+1000=O(n^2)$。所以复杂度是一种大致的非精确的数量级分析，也是一种趋势分析。

实际上，如果仅仅选择算法中循环最深层的原操作（基本操作）来统计 T(n)，发现求出的时间复杂度是相同的，所以算法分析在大多数情况下采用这种简化的方法。

算法空间分析也是采用复杂度的方法，只是需要注意这里所指的空间是临时空间，即函数体内部分配的空间数量。

1.2.2 面试题解析

【面试题 1-5 ★】若一个算法的时间复杂度用 T(n)表示，其中 n 的含义是（　　）。

A．问题规模　　B．语句条数　　C．循环层数　　D．函数数量

答：算法的时间复杂度是问题规模 n 的函数。答案为 A。

【面试题 1-6 ★★】算法的时间复杂度取决于（　　）。

A．待处理数据的状态　　B．处理器的速度

C．问题的规模　　D．程序所占的空间

答：这道题有争议，一般认为算法的时间复杂度是问题规模 n 的函数，那么答案只有 C。算法的时间复杂度又分为最好、最坏和平均时间复杂度，而最好和最坏时间复杂度是与待处理数据的状态有关的，如果这里考虑了它们，答案就应该为 A、C。

【面试题 1-7 ★★】下列记号 O 的定义正确的是（　　）。

A．$O(g(n))=\{f(n)|$存在正常数 c 和 n_0 使得所有 $n \geq n_0$ 有 $0 \leq f(n) \leq cg(n)\}$

B．$O(g(n))=\{f(n)|$对于任何正常数 $c>0$，存在 $n_0>0$ 使得对所有 $n \geq n_0$ 有 $0 \leq cg(n)<f(n)\}$

C．$O(g(n))=\{f(n)|$对于任何正常数 $c>0$，存在 $n_0>0$ 使得对所有 $n \geq n_0$ 有 $0 \leq f(n)<cg(n)\}$

D．$O(g(n))=\{f(n)|$存在正常数 c 和 n_0 使得所有 $n \geq n_0$ 有 $0 \leq cg(n) \leq f(n)\}$

答：f(n)=O(g(n))表示存在正常量 c 和 n_0，当 $n \geq n_0$ 时有 $0 \leq f(n) \leq cg(n)$。答案为 A。

【面试题 1-8 ★★】以下关于渐进记号的性质正确的是（　　）。

A．$f(n)=O(g(n)) \Leftrightarrow g(n)=O(f(n))$

B．$f(n)=O(g(n)), g(n)=O(h(n)) \Rightarrow h(n)=O(f(n))$

C．$f(n)=O(g(n)), g(n)=O(h(n)) \Rightarrow f(n)=O(h(n))$