

我们眼中的数据结构与算法

目录

CONTENTS

-
01. 数据结构与算法的重要性
 02. 数据结构与算法的发展史
 03. 数据结构与算法的名人轶事
 04. 对数据结构与算法课程的期望

1. 数据结构与算法的重要性

瑞典计算机科学家，图灵奖得主 Niklaus Wirth 在 1976 年写了一本书，叫作《Algorithms + Data Structures = Programs》（算法+数据结构=程序）。时至今日，这句话依旧被许多程序员奉为圭臬，一阵见血的点出了数据结构与算法对于编程的重要性。人们研究数据结构和算法的目的，是为了学会编写效率更高的程序，换句话说高效率程序的设计是基于良好的信息组织和优秀的算法。

从历史上看，因研究数据结构和算法，做出卓越贡献而获得图灵奖的科学家也不在少数，大约有21位图灵奖得主在数据结构与算法领域做出重大突破，可见数据结构与算法层面的突破对于计算机行业的发展有着不可或缺贡献。

从现实上看，程序员在求职时，也总是避免不了被问及是否熟练掌握数据结构与算法的相关知识，对于数据结构和算法的理解，能体现出一个人是否能在编程工作中灵活选用合适的数据结构和算法，有效降低代码的复杂度并提升任务的完成效率，能够直接体现出这个人对于编程这项工作的理解深度和实力。

22989-客户端开发高级工程师（深圳）

 分享  收藏

CSIG | 深圳 | 技术 | 2021年03月01日

工作职责

腾讯会议音频和视觉相关算法工程优化；
腾讯会议整体工程性能优化；
相关前沿技术探索。

工作要求

计算机、电子等相关专业硕士以上学历；
具备扎实的C/C++、**数据结构与算法基础知识**，良好的编程动手能力、逻辑思维和快速学习能力；
对以下一种或多种异构计算硬件体系结构、并行计算有深入研究和实践经验：
ARM/x86等CPU上算法性能优化经验，熟悉汇编指令（如ARMv7/v8-A、多线程编程，各种内存管理技术；
Nvidia/AMD/Adreno/Mali等GPU上算法性能优化经验，熟悉GPU架构编程；
CEVA/Cadence/Movidius/Hexagon等DSP上算法性能优化经验，熟悉DMA操作；
熟悉机器学习算法和模型，熟悉主流机器学习平台框架（如Caffe/pytorch/Tensorflow），熟悉主流机器学习推理框架（如ArmNN/MNN/ncnn/Tensorflow Lite）；
良好的团队合作意识，优秀的沟通和学习能力。

15579-游戏高级C++开发工程师（深圳）

IEG | 深圳 | 技术 | 2021年03月01日

工作职责

负责精品手游C++GameCore基础框架机制的开发和维护；
负责精品手游帧同步框架的机制的开发扩展和维护；
负责精品手游多线程技术方案的扩展和优化；
负责精品手游其他技术方案的开发和维护。

工作要求

具备较好的C++/C#、**数据结构、算法基础**；
具备一定的架构机制设计能力；
游戏行业4年以上客户端开发经验，具备底层公共机制开发经验优先；
熟悉Unity引擎，具备Unity游戏项目开发经验优先；
具备较好的学习能力，解决问题能力优先。

腾讯相关职位要求均点出数据结构与算法的重要性

2. 数据结构与算法的发展史

- 数据结构发展史（部分引用自张抒菲的《数据结构发展史》一文）

根据百度百科，我们不难得到数据结构如下的定义，数据结构是计算机存储、组织数据的方式。数据结构是指相互之间存在一种或多种特定关系的数据元素的集合。通常情况下，精心选择的数据结构可以带来更高的运行或者存储效率。数据结构往往同高效的检索算法和索引技术有关。那么数据结构是由何产生的？我们需要把指针拨回到1968年，随着计算机科学技术的不断发展，日趋庞大的数据给程序设计带来了挑战，如何处理好庞大数据之间的关系成了关键，而恰好在这一年，克努思教授开创了数据结构的最初体系，他所著的《计算机程序设计艺术》第一卷《基本算法》是第一本较系统地阐述数据的逻辑结构和存储结构及其操作的著作。70年代初，数据结构作为一门独立的课程开始进入大学课堂。

而数据结构随着程序设计的发展而发展。程序设计经历了三个阶段：无结构阶段、结构化阶段和面向对象阶段，相应地，数据结构的发展也经历了三个阶段：

(1)无结构阶段。40~60年代，计算机的应用主要针对科学计算，程序设计技术以机器语言/汇编语言为主，程序处理的数据是纯粹的数值，数据之间的关系主要是数学公式或数学模型。这一阶段，在人类的自然语言与计算机编程语言之间存在着巨大的鸿沟，程序设计属于面向计算机的程序设计，设计人员关注的重心是使程序尽可能地被计算机接受并按指令正确执行，至于程序能否让人理解并不重要。

(2)结构化阶段。60~80 年代，计算机开始广泛应用于非数值处理领域，数据表示成为程序设计的重要问题，人们认识到程序设计规范化的重要性，提出了程序结构模块化，并开始注意数据表示与操作的结构化。数据结构及抽象数据类型就是在这种情况下形成的。数据结构概念的引入，对程序设计的规范化起到了重大作用。图灵奖获得者沃思给出了一个著名的公式：数据结构+ 算法= 程序。从这个公式可以看到，数据结构和算法是构成程序的两个重要的组成部分，一个软件系统通常是以一个或几个关键数据结构为核心而组织的。随着软件系统的规模越来越大、复杂性不断增加，人们不得不对结构化技术重新评价。由于软件系统的实现依赖于关键数据结构，如果这些关键数据结构的一个或几个有所改变，则涉及到整个系统，甚至导致整个系统彻底崩溃。

(3)面向对象阶段。面向对象技术（首先是面向对象程序设计）始于80 年代初，是目前最流行的程序设计技术。在面向对象技术中，问题世界的相关实体被视为一个对象，对象由属性和方法构成，属性用以描述实体的状态或特征，方法用以改变实体的状态或描述实体的行为。一组具有相同属性和方法的对象的集合抽象为类，每个具体的对象都是类的一个实例。例如，“教师”是一个类，“王老师”、“李老师”等对象都是“教师”类的实例。由于对象（类）将密切相关的属性（数据）和方法（操作）定义为一个整体，从而实现了封装和信息隐藏。使用类时，无需了解其内部的实现细节，一旦数据（结构）修改了，只需修改类内部的局部代码，软件系统的其余部分无需修改。

● 算法的发展史（部分引用自维基百科）

在维基百科上，算法是这样定义的：算法，在数学和计算机科学之中，一个被定义好的、计算机可施行之指示的有限步骤或次序，常用于计算、数据处理和自动推理。作为一个有效方法，算法被用于计算函数，它包含了一系列定义清晰的指令，并可于有限的时间及空间内清楚的表述出来。

而高德纳在他的著作《计算机程序设计艺术》里对算法的特征进行了归纳：输入，输出，明确性，有限性，有效性。由此我们可以追溯算法的起源与发展，早在中国古代，算法在文献中称为“术”，最早出现在《周髀算经》、《九章算术》。特别是《九章算术》，给出四则运算、最大公约数、最小公倍数、开平方根、开立方根、求素数的埃拉托斯特尼筛法，线性方程组求解的算法。三国时代的刘徽给出求圆周率的算法：刘徽割圆术。

而英文名称“algorithm”来自于9世纪波斯数学家花拉子米，因为花拉子米在数学上提出了算法这个概念。

“算法”原为“algorism”，即“al-Khwarizmi”的音转，意思是“花拉子米”的运算法则，在18世纪演变为“algorithm”。而欧几里得算法被人们认为是史上第一个算法，即辗转相除法，是指用于计算两个非负整数 a ， b 的最大公约数。第一次编写程序是爱达·勒芙蕾丝于1842年为巴贝奇分析机编写求解伯努利微分方程的程序，因此爱达·勒芙蕾丝被大多数人认为是世界上第一位程序员。但由于查尔斯·巴贝奇未能完成他的巴贝奇分析机，这个算法未能在巴贝奇分析机上执行。因为“well-defined procedure”（良定义）缺少数学上精确的定义，19世纪和20世纪早期的数学家、逻辑学家在定义算法上出现了困难，因此算法发展进入困境。20世纪的英国数学家图灵提出了著名的图灵论题，并提出一种假想的计算机的抽象模型，这个模型被称为图灵机。图灵机的出现解决了算法定义的难题，图灵的思想对算法的发展起到了重要的作用。而当“ENIAC”于1946年在美国宾夕法尼亚大学诞生，硬件层面的不断升级，需求不断多元化，也促使算法不断发展。

3. 数据结构与算法的名人轶事



李开复

李开复在《算法的力量》一书写到：“还记得1988年贝尔实验室副总裁亲自来访问我的学校，目的就是为了想了解为什么他们的语音识别系统比我开发的慢几十倍，而且，在扩大至大词汇系统后，速度差异更有几百倍之多。他们虽然买了几台超级计算机，勉强让系统跑了起来，但这么贵的计算资源让他们的产品部门很反感，因为“昂贵”的技术是没有应用前景的。在与他们探讨的过程中，我惊讶地发现一个 $O(n*m)$ 的动态规划(dynamic programming)居然被他们做成了 $O(n*n*m)$ 。更惊讶的是，他们还为此发表了不少文章，甚至为自己的算法起了一个很特别的名字，并将算法提名到一个科学会议里，希望能得到大奖。当时，贝尔实验室的研究员当然绝顶聪明，但他们全都是学数学、物理或电机出身，从未学过计算机科学或算法，才犯了这么基本的错误。我想那些人以后再也不用嘲笑学计算机科学的人了吧！”

4.对数据结构与算法课程的期望

- 正如《数据结构与算法分析》前言中写到那般，“程序高效率的要求不会也不应该与合理的设计和简明清晰的编码相矛盾。高效率程序的设计基于良好的信息组织和优秀的算法，而不是基于“编程小伎俩”。一名程序员如果没有掌握设计简明清晰程序的基本原理，就不可能编写出有效的程序。”所以我希望自己能够在数据结构与算法这门课程上能熟练掌握与运用数据结构与算法，通过大量的编程训练以达成目的。当然单纯的编码不一定是快乐的，一路上也必定有许多坎坷和挫折，希望自己能苦中作乐，从课程中汲取更多经验与知识，同时也希望课程能相对有趣生动，将枯燥的知识以有趣的内容呈现。