

南京大学面试方式很独特，一般将学生分称 ABCDE 组，每组有三个老师面试，每个老师出 5 道题，其中 4 个是专业知识题，1 个是英语能力测试题，每题 10 分，总共 150 分。

复试报道时候告之面试分组情况和面试是上午还是下午。面试之前统一开会，告诉面试中注意事项，并在此时告之 ABCDE 组分别由哪三个老师负责，然后学生根据分组情况，拿着记录本去各个老师办公室面试。也就是说，与其他学校很多老师统一在一个房间面试一个学生不一样，南大面试是一对一的，老师身边只有个做记录的研究生，学生可以选择先面哪个老师后面哪个老师。面试时记录本交给研究生，老师面试研究生记录，打分在老师另外本子上，记录本上没有打分，所以学生不知道自己面试分数。但是记录本上会记录你的面试题。所以后进去同学可以知道前面同学的面试题，这些面试题有的是重复的，但是重复的大多是不能叙述出来的或是实际应用的题。英语题一般是准备很多个，随机抽选一个。比如郭延文老师的英语翻译题是全部放在一个电脑上，有很多，选好了让你做一个。面试时候还有两个细节在这里说一下：第一，进去时候有的老师会问你是哪个学校学生、学什么专业等，可能对非重点学校学生还是有一些偏见。第二，如果你做过什么项目，面试前一定要准备怎样说明突出你所做的项目，这将在面试时很有帮助。

8:30，满怀期望略带紧张的走进蒙民伟 109，抽到的是下午面试，紧张的心情略有放松。和 HSj、PMx 在 109 聊了聊，看了看书。后面打听到了上午的面试的一些情况。今年面试和往年情况有所不同，往年都是比较年轻的老师，比如赵建华、周志华、陶先平、陈贵海等等，今年都是我们系资历很老的老师，陈道蓄、费祥林、还有张福炎。果然，面试的风格差别也很大，至少和我开始时与期待的差得很远。着实很紧张。

费祥林

我们知道 Linux/Unix 里面有系统进程在用户进程中的使用，解释一下。知道“面包店算法”吗？解释系统调用 syscall 和库函数的区别。最后一个问题，在计算机领域有很多典型问题，比如在操作系统中的“生产者消费者问题”，你能不能列举一下在其他领域的典型问题，比如在人工智能方面、图形学、算法等等？人工智能方面有一个比较著名的图灵测试，你知道吗？

张福炎

（多媒体技术及应用，计算机图形学，中文信息处理等）

张老的每一个问题都是由浅入深根据你的回答，一步一步深入下去的一大堆小问题组成的。

第一个问题：关于 Cache，问到了原理和思想、评价标准、改进方案、计算机软硬件中其他用到这个思想的地方。

第二个问题：关于操作系统中的文件管理和组织，后来抓着我问 FAT。

第三个问题：数据结构中的线性表，栈是不是线性表，还有什么用到了栈的机制。

第四个问题：电子邮件的协议，收发机制，其它协议等。

最后一题：编程能力怎样？用过那些环境？控件是什么？说一下 MFC 中的类库。

C 头。

第一题，基础知识：解释“抽象数据类型”，解释“优先队列”这一抽象数据类型及其实现方法。

第二题，专业知识：电子邮件收发到接收的过程。

第三题，新技术：先是位置敏感，不懂，请求换了一个，变成了关于 P2P 技术。

第四题，综合应用：握手问题，一个实际应用题，就不细说了。

陈贵海

（并行处理与分布式计算，网络计算，高性能计算机体系结构，计算机软硬件协同设计）

图灵奖获得者；英语陈述题（最喜欢的课程之类的）；图论题（离散数学）；王国维，学习的三个境界

王国维《人间词话》第二十六：古今之成大事业、大学问者，必经过三种之境界："昨夜西风凋碧树。独上高楼，望尽天涯路。"此第一境也。"衣带渐宽终不悔，为伊消得人憔悴。"此第二境也。"众里寻他千百度，蓦然回首，那人却在，灯火阑珊处。"此第三境也。此等语皆非大词人不能道。然遽以此意解释诸词，恐为晏欧诸公所不许也。

陶先平

网络的七层协议名称；罗素悖论；图灵机；软件方法的问题

武岗山：图形图像相关

马晓星

（软件体系结构，软件 Agent，软件协同。）

什么是计算机，计算，语法，语义，语用？群论中 Lagrange 定理及其证明；设 G 是有限群， H 是 G 的子群，则 $|G| = |H| * [G : H]$ （即 G 中元素个数等于 H 中元素个数乘以 H 在 G 中的陪集个数。

谈谈你自己的一个程序设计的成果；做过什么项目？用什么邮件系统？你平时的一个喜好是什么？为什么喜欢？怎么样向其他人推荐或者影响的？

窦万春

(知识网格、认知科学、知识流、E - Science 支撑技术)

操作系统的三态，五态，七态分别是什么？如何表示小数点这一概念，在计算机体系中？FTP 和 HTTP 中文件传输的异同？翻译一篇关于 workflow 的文章的摘要；集合，关系，图和树之间的关系；“君子怀德，小人怀土；君子怀刑，小人怀惠。”

陈道蓄

给定序列 123，通过栈操作和打印操作，不给其他的存储器，能都打印 123 的任何一种排列；FP Protocol (about mobile agent)；数据库中的查询优化有哪些？MS 的 Sql server 中有哪些措施？k 条直线能最多把一个平面分几份？折线(夹一个锐角)呢？连续函数，一致连续，N 个连续函数相加还是连续函数么？一致连续呢？计算机中如何表示数据，如何表示知识，谈谈数据结构；解释一下平台无关性(英文)

一老板买光碟，第一天卖出去 60 张，10 元 3 张卖出去 30 张，10 元 2 张卖出去 30 张。第二天也卖出去 60 张，20 元 5 张，问，两天营业额有没有变化，为什么？

黄皓

进程和线程的区别，然后提到了一个切换的时候的栈的问题；路由协议；带冲突检测的载波侦听的英文全称 CSMA/CD，还有什么指数回退的原因；智力题，就是什么一条船，每次只能过两个人，要多长时间；编程：a.实现简单的客户与服务器交互的程序，b.RSA 算法的模拟，c.编程计算 24 点。三题选一题完成。

陈桂海

ACM 全称，是哪国的；IEEE 全称；What is the definition of OS? What is dead lock? (用英文回答)；死锁的条件；一个编译原理的题目。

茅兵

一道智力题，就是给出一堆的家庭关系，从中找出谁是医生；过什么项目，具体介绍一下？对哪些科目感兴趣？然后问 Windows 有哪些漏洞？为什么 Windows 容易被攻击？茅老是搞信息安全的，说 Unix 的漏洞也很多，只是攻击的价值没有 Windows 大，所以人们关注的比较少。

一道小学奥林匹克信息学竞赛的题目，是一段 Pascal 写的程序，让说出程序的功能。

刘奇志

刘是搞多媒体和图形图像的。数据库的恢复方法有哪些？说说数据库的接口；有没有做过什么项目有用到数据库的？多媒体显示方法的问题。问研究人工智能最重要的是什么？专家系统的知识，专家系统中说专家知识获取更重要还是效率更重要？还问了一些人工智能算法；怎么样编程把一个整数变成字符串？英语叙述，"Have you attended any contest?"

郭延文

C++中析构函数和构造函数哪个可以进行重载？C++中有哪些方法可以实现多态？；C++中的类和 C 语言中的 struct 有什么区别？线性代数问题，给出一个矩阵，求矩阵的特征值和特征向量。

有一个质地非均匀的香，可以烧一个小时，问如何确定 30 分钟？问如果有两个质地非均匀的香，每个都可以少一个小时，如何确定 45 分钟？英译汉，关于算法的优点。

申富饶

特征向量和特征值的定义，特征向量是不是唯一的？写出贝叶斯的公式；英语自我介绍；C 语言的指针概念和内容；C++中的类和对象的关系；毕业设计做的什么。

谢俊元

英语介绍母校；毕业设计做什么；什么是图灵测试；什么是操作系统；pv 操作是什么意思，死锁概念，怎么解决死锁；排序有哪几种，快排是不是最快的。

LT

什么是虚拟存储器；虚拟存储器，内存，cache 三者之间的关系；什么是瀑布模型；页表是用什么实现的，里面存储的是什；读一篇英语文章。

虚函数主要作用；介绍模板；关于继承的问题；数据库关系模型；给出一段论语，让说出它的意思；图形图像类；嵌入式的问题；做过什么项目？（貌似每个老师都问）

给出一段代码，找错。（貌似错在类嵌套定义，类的成员函数的形参是类本身，具体的和复试指定那本 c++类那章一道题差不多）

排序问题，从 n 个整数中找出最大的四个数，用尽可能多的方法。（我只会冒泡，选择，但复杂度都很高，貌似应该用堆来做）

n 个科学家，对应有 n 顶帽子，现把帽子放在一起，每个科学家拿一顶，问每个科学家都拿到不是自己的帽子的概率有多大，当 n 趋于无穷时，这个概率是否为 0？

c++和 C 比，最特别的地方是什么

tcp 和 udp 的有无连接问题。

网络的七层协议是什么。

是想做毕业后直接工作还是搞科研，这个问题 3 个老师都问了

为什么跨考计算机，具体地说说将来想从事的方向

快排的时间、空间复杂度，包括一般和最坏情况

设计一个实现搜索引擎的算法，概要地描述一下

指针传递与值传递的不同

说说平时接触过的软件中用到 malloc 的地方

算法题，上台阶问题：一共 n 级台阶，你一次只能上一级或者两级，请问有几种不同的走法？提示，用递归实现。

介绍一下图灵机

介绍一下图像处理方面的几个相关算法

数据库 ACID

一个方格，从左上角走到右下角，只能向下或者向右，问有几种走法

如何在 100W 个，或者更多的数据里面找到前 50 个；

说说大数据下的关系数据库的弊端

进程死锁与饿死的区别

进程同步于互斥的共同点与不同点（不是单个的概念解释）

进程间通信的几种方式，以及实际的经验？

设计模式有哪些种？说出名字等

有四个装有粒状药瓶的瓶子，其中三瓶中每一粒药瓶的重量均为 x，另一瓶药由于受潮每一粒变为 x+1. 现有一电子称，问如何只称一次确定哪一瓶药受潮；

什么是归并排序？它的时间复杂度是多少？这是它的平均复杂度还是最差复杂度？如果是平均复杂度，那么它的最差复杂度是多少，为什么；

有一单链表，如果它的最后一个节点的 next 并不是指向空而是指向前面的某一个节点则会构成一个环，请设计一个算法来判断一个单链表中是否有环；

说说如果让你设计一个“智能书架（或者智能冰箱之类的）”，你会使用什么样的技术。为什么？没有其他提示，问多了他还不让再问，我就是不懂他的意思啊！

const char *p 与 char const *p 的区别；

int *p; *p = 0 ; 请问上述代码有错吗；

请解释你学过的专业课中的一个名词，我讲了 RSA，然后老师就紧追不舍，一直问到密码学，数论，计算机网络，网络安全，还要说明这几门课的联系；

你最熟悉哪门语言，我选了 C++，老师就问我 C++ 内存有几个区，他是怎样进行动态地址分配的，请举一个你平时所用的软件中用到动态分配的例子；

学过数据库没，我说学过，他就要我用 C 语言实现一个简单的 select 语句功能；

说说数据库中的一些名词。我就说了模式、视图、触发器、范式等等。视图的作用；

最熟悉什么语言，我回答 C++，老师就问我在编程的时候遇到问题是怎么调试的；

最喜欢的业余爱好，我说单机游戏，老师就要我用描述一下所玩的单机游戏；

一个硬币可抛出正反面，可抛若干次，请设计一个等概率到 1-6 的映射；

计算 X 的 N 次方，要求时间复杂度最小

求最长公共子串，比如 ABCDEFG 和 QAWBG 的最长公共子串是 ABG

const 跟#define 相比，有何优点

面向对象的特征？

什么是红黑树？红黑树是用来干什么的？

图形与图像有什么区别？怎么存储的？

MapReduce 的英文，映射-规约

并行计算与集群有神马区别？有哪些并行计算。我答了网格计算、云计算，貌似不是很对。

什么是 MVC? 为什么要用 MVC?

C++与 Java 的区别? 我说了指针, 和编译

看镜子, 为啥上下不颠倒, 左右颠倒?

一个有向图 G , 设每个顶点 w 对应一个值 $d(w)$, 对任意边 (u, v) 属于 G , 则 $d(u) - d(v) \equiv 1 \pmod{m}$, 任意给定一个有向图 ($d(w)$ 未知), 判定 m 是否存在, 若存在如何求出 m 的值。($d(w)$ 未知是重点, 各种不会)

引用和指针的区别

数 $1-2n$ 中取任意 $n+1$ 个数, 是否一定有两个数互素, 我说 $n+1$ 个数中必有两个相邻, 这两个互素

是否知道哥德巴赫猜想, 陈景润证明 $1+2$

有 3 把椅子, 其中一把后边藏了奖品, 让你只能选一把, 中奖概率是多少? 如果你选了一把后(还没看有没有奖品), 有人帮你排除了剩下两把中的一把没奖品的, 问你要不要改变你的选择?

有一个随机数生成函数以等概率 $1/2$ 产生数字 1 和 2, 问如何用它以等概率 $1/3$ 产生 1、2、3 三个数字? 答上来后又问要产生序列 “123” 需要调用这个函数多少次?

拉普拉斯变换(到复频域, 原函数限制少)、傅立叶变换、Z 变换(离散序列)为什么要有三个变换? z 变换则是将拉普拉斯的复平面投影到 z 平面, 将虚轴变为一个圆环

给一道几何题, 长为 n 段线段、宽为 m 段线段的网格, 里面有多少个矩形?

事务的四个性

哪些算法用到贪心, 问迪杰斯特拉算法是贪心的还是动态规划的; 翻译一段英文(关于数据库的)

大数定理的证明

数据库的 E-R 图, 为什么有 E-R 图等等

英文详细描述你最善长的一件事, 或今天早上发生的事, 描述你未来的职业规划, 描述为什么你报计算机专业, 描述你大学期间做的最自豪的事情, 描述今天的天气, 描述印象最深刻的老师, 包括事件和感想, 介绍一门比较熟悉的语言, 介绍你最喜欢的课程(会追着你问这个课程的问题), 叙述一下你在研究生期间有什么打算, 准备从事哪个方向的研究, 说一下你最喜欢哪个计算机方向; 假设我现在是一名大一新生, 你是一位老师, 请用英语给我讲授一门课程; 最喜欢的 web application 是什么

图灵测试：又称“图灵判断”，是图灵提出的一个关于机器人的著名判断原则。所谓图灵测试是一种测试机器是不是具备人类智能的方法。被测试的有一个人，另一个是声称自己有人类智力的机器。图灵测试是测试人在与被测试者(一个人和一台机器)隔开的情况下，通过一些装置（如键盘）向被测试者随意提问。问过一些问题后，如果测试人不能确认被测试者 30%的答复哪个是人、哪个是机器的回答，那么这台机器就通过了测试，并被认为具有人类智能。目前，还没有一台机器能够通过图灵测试。

ACM： Association for Computing Machinery 美国计算机协会。

图灵机：指一个抽象的机器，它有一条无限长的纸带，纸带分成了一个一个小方格，每个方格有不同的颜色。有一个机器头在纸带上移来移去。机器头有一组内部状态，还有一些固定的程序。在每个时刻，机器头都要从当前纸带上读入一个方格信息，然后结合自己的内部状态查找程序表，根据程序输出信息到纸带方格上，并转换自己的内部状态，然后进行移动。

抽屉原理（鸽巢原理）：桌上有十个苹果，要把这十个苹果放到九个抽屉里，无论怎样放，我们会发现至少会有一个抽屉里面至少放两个苹果。这一现象就是我们所说的“抽屉原理”。抽屉原理的一般含义为：“如果每个抽屉代表一个集合，每一个苹果就可以代表一个元素，假如有 $n+1$ 或更多于 $n+1$ 个元素放到 n 个集合中去，其中必定至少有一个集合里有二个元素。” 抽屉原理有时也被称为鸽巢原理（“如果有五个鸽子笼，养鸽人养了 6 只鸽子，那么当鸽子飞回笼中后，至少有一个笼子里装有 2 只鸽子”）。它是组合数学中一个重要的原理。

预测股票走势的软件：用 BP 神经网络对股票走势进行分析和预测的可行性。BP 网络通过对以往历史数据的学习，找出股市发展的内在规律，并将其存储在网络具体的权值、阈值中，用以预测未来的走势，并在 matlab 中得以实现。针对 BP 算法在股市预测中存在的学习速度慢、容易陷入局部极小值、预测结果精度不高等问题，提出一种改进的 BP 神经网络算法。通过重新选取神经元的激活函数，对输出层和隐层中神经元转换函数的权值、缩放系数和位移参数进行调整，减少隐层节点数，加快 BP 网络的收敛速度。根据 BP 网络进行股市预测的原理，建立基于 BP 网络的股市预测模型，采用改进后的 BP 算法进行股市预测，并通过 MATLAB 软件对其预测过程进行仿真实验。在仿真过程中对 BP 算法和改进后的 BP 算法在预测股票中的收敛性能进行比较，达到了预测效果。

智能书架：读写器是智能书架信息控制和处理中心，实现图书标签信息的采集和数据通信。天线是磁场能量辐射的单元，书架天线采用阵列形式，对书架检测范围提供有效范围控制，并提高了检测成功率。为使书架各层之间定位准确，实现各层天线组的分时切换，多路器的应用一方面使多个天线组可共用一个读写器，同时也降低了应用成本。智能书架系统与图书馆管理软件系统需要实时数据通信，并受图书馆管理系统的控制，RFID 模块采集到图书标签信息后，将大量的图书标签信息传送给图书馆管理系统。智能书架电气系统基本组成包括：读写器、控制器、天线。读写器：是智能书架的核心设备，完成智能书架图书标签识别、通信与书架控制功能。其中通信支持 TCP/IP 以太网协议，书架单元与图书馆系统数据通信。控制器：完成智能书架各层天线控制和定位工作。天线：天线为智能书架检测电子标签关键单元，决定了检测标签范围。

一次性遍历出一个单链表的中间节点：解析：设立两个指针，比如 $*p$ 和 $*q$ 。 p 每次移动两个位置，即 $p=p->next->next$ ， q 每次移动一个位置，即 $q=q->next$ 。当 p 到达最后一个节点时候， q 就是中间节点了。类似的可以用一次遍历方法得到三分之一、四分之一、前几个节点。

嵌入式系统：是一种专用的计算机系统，作为装置或设备的一部分。通常，嵌入式系统是一个控制程序存储在 ROM 中的嵌入式处理器控制板。事实上，所有带有数字接口的设备，如手表、微波炉、录像机、汽车等，都使用嵌入式系统，有些嵌入式系统还包含操作系统，但大多数嵌入式系统都是由单个程序实现整个控制逻辑。嵌入式系统（Embedded system），是一种“完全嵌入受控器件内部，为特定应用而设计的专用计算机系统”，根据英国电器工程师协会（U.K. IEE）的定义，嵌入式系统为控制、监视或辅助设备、机器或用于工厂运作的设备。与个人计算机这样的通用计算机系统不同，嵌入式系统通常执行的是带有特定要求的预先定义的任务。由于嵌入式系统只针对一项特殊的任务，设计人员能够对它进行优化，减小尺寸降低成本。由于嵌入式系统通常进行大量生产。所以单个的成本节约，能够随着产量进行成百上千的放大。

静态与动态多态：静多态是发散式的，让相同的实现代码应用于不同的场合。动多态是收敛式的，让不同的实现代码应用于相同的场合。思维方式上：静多态是泛型式编程风格，它看重的是算法的普适性。动多态是对象式编程风格，它看重的是接口和实现的分离度。

联编：就是将模块或者函数合并在一起生成可执行代码的处理过程，同时对每个模块或者函数调用分配内存地址，并且对外部访问也分配正确的内存地址，它是计算机程序彼此关联的过程。按照联编所进行的阶段不同，可分为两种不同的联编方法：静态联编和动态联编。静态联编是指在编译阶段就将函数实现和函数调用关联起来，因此静态联编也叫早绑定，在编译阶段就必须了解所有的函数或模块执行所需要检测的信息，它对函数的选择是基于指向对象的指针（或者引用）的类型，C 语言中，所有的联编都是静态联编，据我所知道的，任何一种编译器都支持静态联编（废话）。动态联编是指在程序执行的时候才将函数实现和函数调用关联，因此也叫运行时绑定或者晚绑定，动态联编对函数的选择不是基于指针或者引用，而是基于对象类型，不同的对象类型将做出不同的编译结果。C++中一般情况下联编也是静态联编，但是一旦涉及到多态和虚拟函数就必须使用动态联编了。

多态：字面的含义是具有多种形式或形态。C++多态有两种形式，动态多态和静态多态；动态多态是指一般的多态，是通过类继承和虚函数机制实现的多态；静态多态是通过模板来实现，因为这种多态实在编译时而非运行时，所以称为静态多态。

一个实现搜索引擎的算法：现有的搜索引擎排序算法，以基于网页链接结构的算法为主，主要的两种代表性算法是PageRank 算法和 HITS 算法，HITS 算法是由康奈尔大学(Cornell University) 的 Jon Kleinberg 博士于 1998 年首先提出。Kleinberg 认为既然搜索是开始于用户的检索提问，那么每个页面的重要性也就依赖于用户的检索提问。他将用户检索提问分为如下三种：特指主题检索提问（specific queries，也称窄主题检索提问）、泛指主题检索提问（Broad-topic queries，也称宽主题检索提问）和相似网页检索提问（Similar-page queries）。HITS 算法专注于改善泛指主题检索的结果。

零一律：是概率论中的一个定律，它是安德雷·柯尔莫哥洛夫发现的，因此有时也叫柯尔莫哥洛夫零一律。其内容是：有些事件发生的概率不是几乎一（肯定发生），就是几乎零（肯定不发生）。这样的事件被称为“尾事件”。尾事件是由无限多的随机变量的序列来定义的。比如它不是与 X_1 的值无关。比如假如我们扔无限多次银币，则连续 100 次数字面向上的事件是一个尾事件。

白盒测试：是通过程序的源代码进行测试而不使用用户界面。这种类型的测试需要从代码句法发现内部代码在算法，溢出，路径，条件等等中的缺点或者错误，进而加以修正。黑盒测试：是通过使用整个软件或某种软件功能来严格地测试，而并没有通过检查程序的源代码或者很清楚地了解该软件的源代码程序具体是怎样设计的。测试人员通过输入他们的数据然后看输出的结果从而了解软件怎样工作。在测试时，把程序看作一个不能打开的黑盒子，在完全不考虑程序内部结构和内部特性的情况下，测试者在程序接口进行测试，它只检查程序功能是否按照需求规格说明书的规定正常使用，程序是否能适当地接收和正确的输出。逻辑覆盖是以程序内部的逻辑结构为基础的设计测试用例的技术。属白盒测试。这一方法要求测试人员对程序的逻辑结构有清楚的了解，甚至要能掌握源程序的所有细节。由于覆盖测试的目标不同，逻辑覆盖又可分为：语句覆盖、判定覆盖、判定一条条件覆盖、条件组合覆盖及路径覆盖。

边界值分析法（BVA, Boundary Value Analysis）：是用于对输入或输出的边界值进行测试的一种黑盒测试方法。在测试过程中，边界值分析法是作为对等价类划分法的补充，专注于每个等价类的边界值，两者的区别在于前者在等价类中随机选取一个测试点。边界值分析法采用一到多个测试用例来测试一个边界，不仅重视输入条件边界值，而且重视输出域中导出的测试用例。边界值分析法比较简单，仅用于考察正处于等价划分边界或边界附近的状态，考虑输出域边界产生的测试情况，针对各种边界情况设计测试用例，发现更多的错误。边界值分析法的测试用例是由等价类的边界值产生的，根据输入输出等价类，选取稍高于边界值或稍低于边界值等特定情况作为测试用例

云计算(Cloud Computing)：是分布式处理(Distributed Computing)、并行处理(Parallel Computing)和网络计算(Grid Computing)的发展。（一）云计算的基本原理。通过使计算分布在大量的分布式计算机上，而非本地计算机或远程服务器中，企业数据中心的运行将更与互联网相似。这使得企业能够将资源切换到需要的应用上，根据需求访问计算机和存储系统。这就好比是从古老的单台发电机模式转向了电厂集中供电的模式。它意味着计算能力也可以作为一种商品进行流通，就像煤气、水电一样，取用方便，费用低廉。最大的不同在于，它是通过互联网进行传输的。云计算的蓝图已经呼之欲出：在未来，只需要一台笔记本或者一个手机，就可以通过网络服务来实现我们需要的一切，甚至包括超级计算这样的任务。从这个角度而言，最终用户才是云计算的真正拥有者。云计算的应用包含这样的一种思想，把力量联合起来，给其中的每一个成员使用。从最根本的意义来说，云计算就是利用互联网上的软件和数据的能力。

面包店算法：该算法的基本思想源于顾客在面包店中购买面包时的排队原理。顾客在进入面包店前，首先抓一个号，然后按照号码由小到大的次序依次进入面包店购买面包。这里，面包店发放的号码是由小到大的，但是两个或两个以上的顾客却有可能得到相同的号码(使所抓号码不同需要互斥)，如果多个顾客抓到相同的号码，则规定按照顾客名字的字典次序进行排序，这里假定顾客是没有重名的。在计算机系统中，顾客就相当于进程，每个进程有一个唯一的标识，我们用 P 的下面加一个下标来表示。例如：对于 P_i 和 P_j ，如果有 $i < j$ ，则先为 P_i 服务，即 P_i 先进入临界区

人工智能的典型算法： A^* (A-Star)算法是一种静态路网中求解最短路，是启发式算法式图搜索算法。企图寻找从起始状态至目标状态的耗散最小的路径 A^* 算法在静态路网中的应用有效的方法。公式表示为： $f(n)=g(n)+h(n)$ ，其中 $f(n)$ 是从初始点经由节点 n 到目标点的估价函数， $g(n)$ 是在状态空间中从初始节点到 n 节点的实际代价， $h(n)$ 是从 n 到目标节点最佳路径的估计代价。保证找到最短路径（最优解的）条件，关键在于估价函数 $h(n)$ 的选取：估价值 $h(n) \leq n$ 到目标节点的距离实际值，这种情况下，搜索的点数多，搜索范围大，效率低。但能得到最优解。如果估价值 $>$ 实际值，搜索的点数少，搜索范围小，效率高，但不能保证得到最优解

图像处理：中点画线，中点画圆，多变形裁剪，多边形区域填充 Bezier 曲线生成等

Cache 的工作原理：基于程序访问的局部性。对大量典型程序运行情况的分析结果表明，在一个较短的时间间隔内，由程序产生的地址往往集中在存储器逻辑地址空间的很小范围内。指令地址的分布本来就是连续的，再加上循环程序段和子程序段要重复执行多次。因此，对这些地址的访问就自然地具有时间上集中分布的倾向。Cache 的设计有在片一级分离和统一设计两种方案。Intel、AMD、原 DEC 等公司将 L1 Cache 设计成指令 Cache 与数据 Cache 分离型。因为这种双路高速缓存结构减少了争用高速缓存所造成的冲突，改进了处理器效能，以便数据访问和指令调用在同一时钟周期内进行。但是，仅依靠增加在片一级 Cache 的容量，并不能使微处理器性能随之成正比例地提高，还需设置二级 Cache。在 L1 Cache 结构方面，一般采用回写式静态随机存储器（SRAM）。目前，L1 Cache 容量有加大的趋势。L2 Cache 的设计分芯片内置和外置两种设计。

一些图灵奖(由 ACM 颁发)获得者

1971 年 约翰·麦卡锡 John McCarthy	人工智能 (Lisp)
1972 年 艾兹格·迪科斯彻 Edsger Dijkstra	程序设计语言
1974 年 高德纳 Donald E. Knuth	算法分析、程序设计语言、程序设计
1977 年 约翰·巴克斯 John Backus	程序设计语言规范的形式化定义
1981 年 埃德加·科德 Edgar F. Codd	关系型数据库
1983 年 肯·汤普逊 & 丹尼斯·里奇 K & R	UNIX 和 C 语言
1984 年 尼古拉斯·沃斯 Niklaus Wirth	PASCAL 程序设计语言设计、程序设计

OSI：是一个开放性的通行系统互连参考模型，他是一个定义的非常好的协议规范。OSI 模型有 7 层结构，每层都可以有几个子层。OSI 的 7 层从上到下分别是 7 应用层对应应用程序的通信服务 6 表示层 定义数据格式及加密 5 会话层 它定义了如何开始、控制和结束一个会话，包括对多个双向消息的控制和管理，以便在只完成连续消息的一部分时可以通知应用，从而使表示层看到的数据是连续的 4 传输层 是否选择差错恢复协议还是无差错恢复协议，及在同一主机上对不同应用的数据流的输入进行复用，还包括对收到的顺序不对的数据包的重新排序功能 3 网络层 2 数据链路层 在单个链路上如何传输数据 1 物理层 其中高层，既 7、6、5、4 层定义了应用程序的功能，下面 3 层，既 3、2、1 层主要面向通过网络的端到端的数据流。

罗素悖论：把所有集合分为 2 类，第一类中的集合以其自身为元素，第二类中的集合不以自身为元素，假令第一类集合所组成的集合为 P ，第二类所组成的集合为 Q ，于是有： $P=\{A \mid A \in A\}$ $Q=\{A \mid A \notin A\}$ 问， $Q \in P$ 还是 $Q \in Q$ ？若 $Q \in P$ ，那么根据第一类集合的定义，必有 $Q \in Q$ ，但是 Q 中任何集合都有 $A \notin A$ 的性质，因为 $Q \in Q$ ，所以 $Q \notin Q$ ，引出矛盾。若 $Q \in Q$ ，根据第一类集合的定义，必有 $Q \in P$ ，而显然 $P \cap Q = \emptyset$ ，所以 $Q \notin Q$ ，还是矛盾。这就是著名的“罗素悖论”。罗素悖论还有一些较为通俗的版本，如理发师悖论等。1900 年前后，在数学的集合论中出现了三个著名悖论，理发师悖论就是罗素悖论的一种通俗表达方式。此外罗素悖论还有康托尔悖论、布拉利—福尔蒂悖论。这些悖论特别是罗素悖论，在当时的数学界与逻辑界内引起了极大震动。触发了第三次数学危机。

计算机 (Computer)：俗称电脑，是一种能够按照程序运行，自动、高速处理海量数据的现代化智能电子设备。由硬件系统和软件系统所组成，没有安装任何软件的计算机称为裸机。

操作系统的三态五态七态：

运行，阻塞，就绪

运行，阻塞，就绪 创建，终止

创建，运行，静止就绪，活动就绪，静止阻塞，活动阻塞，结束

SQL 查询优化: 1.对查询进行优化,应尽量避免全表扫描,首先应考虑在 **where** 及 **order by** 涉及的列上建立索引。2 应尽量避免在 **where** 子句中对字段进行 **null** 值判断,否则将导致引擎放弃使用索引而进行全表扫描 3.应尽量避免在 **where** 子句中使用 **!=**或 **<>**操作符,否则将引擎放弃使用索引而进行全表扫描。应尽量避免在 **where** 子句中使用 **or** 来连接条件,否则将导致引擎放弃使用索引而进行全表扫描。**.in** 和 **not in** 也要慎用,否则会导致全表扫描。应尽量避免在 **where** 子句中对字段进行表达式操作,这将导致引擎放弃使用索引而进行全表扫描。应尽量避免在 **where** 子句中对字段进行函数操作,这将导致引擎放弃使用索引而进行全表扫描

回退算法: 该方法包括以下步骤:

- 节点启动后进行初始化;
- 当节点没有数据包需要发送时,进入空闲等待状态;
- 当节点有数据包需要发送时,检测信道是否空闲;
- 选取回退计数初始计数值,进行回退计数;
- 当计数器的计数值减为 0 时,节点发送数据包;
- 将该节点当前的竞争窗口值加倍,并将该值与最大竞争窗口值相比较,取其中最小值作为重新发送时的竞争窗口值,回到步骤 c。使得无线节点只需要对数据包发送情况进行统计,就可以实现自适应的改变回退窗口。

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。

数据库的恢复方法: 备份和日志

多媒体装置中文件列表显示方法: 第 1 步骤,通过菜单界面上的输入窗口输入一个以上的单词,并在装置内进行储存和管理; 第 2 步骤,在要求显示文件列表时,则检索并参照所储存的一个以上的单词,并用文件列表显示除包含有上述单词的文件名以外的文件名。

贝叶斯定理: 英国数学家贝叶斯发展,用来描述两个条件概率之间的关系,比如 $P(A|B)$ 和 $P(B|A)$ 。按照乘法法则: $P(A \cap B) = P(A) * P(B|A) = P(B) * P(A|B)$, 可立刻导出贝叶斯定理公式: $P(A|B) = P(B|A) * P(A) / P(B)$ 如上公式也可变形为: $P(B|A) = P(A|B) * P(B) / P(A)$ 贝叶斯公式

瀑布模型 (Waterfall Model): 是一个项目开发架构,开发过程是通过设计一系列阶段顺序展开的,从系统需求分析开始直到产品发布和维护,每个阶段都会产生循环反馈,因此,如果有信息未被覆盖或者发现了问题,那么最好“返回”上一个阶段并进行适当的修改,项目开发进程从一个阶段“流动”到下一个阶段,这也是瀑布模型名称的由来。包括软件工程开发、企业项目开发、产品生产以及市场销售等构造瀑布模型。

n 条直线最多将平面分成 : $1+1+2+3+\cdots+n=1+n(n+1)/2$ 部分。 N 条折线, 分成 $(n^2+n+2)/2$ 个平面

抽象数据类型(Abstract Data Type 简称 ADT): 是指一个数学模型以及定义在此数学模型上的一组操作。抽象数据类型需要通过固有数据类型(高级编程语言中已实现的数据类型)来实现。抽象数据类型是与表示无关的数据类型,是一个数据模型及定义在该模型上的一组运算。对一个抽象数据类型进行定义时,必须给出它的名字及各运算的运算符名,即函数名,并且规定这些函数的参数性质。一旦定义了一个抽象数据类型及具体实现,程序设计中就可以像使用基本数据类型那样,十分方便地使用抽象数据类型。

控件: 是对数据和方法的封装。控件可以有自己的属性和方法。属性是控件数据的简单访问者。方法则是控件 的一些简单而可见的功能。