

NOIP 初赛复习指南

By.史添能 2014-08-1

初赛考的知识点就是计算机基本常识、基本操作和程序设计基础知识。其中选择题考查的是知识,而问题解决类型的题目更加重视能力的考查。一般说来,选择题只要多用心积累就可以了。问题解决题目的模式比较固定,大家应当做做以前的题目。写运行结果和程序填空也需要多做题目,并且培养良好的程序阅读和分析能力,就像语文的阅读理解一样。近几年来,初赛的考查范围有了很大的变化,越来越紧跟潮流了。这就需要大家有比较广泛的知识,包括计算机硬件、软件、网络、简单的数据结构(例如栈、队列、树和图等)和简单的算法(例如排序、查找和搜索等),程序设计语言以及一些基本的数学知识和技巧。

知识点复习:

第一部分 计算机基础知识

1. 计算机的发展 知识点:1>.计算机的发展阶段(4代,标志及主要特点)

2>.ENIAC,图灵,冯·诺依曼,Ada Lovelace(第一个程序员)

2. 计算机系统

1>.计算机硬件

- 组成:运算器,控制器,存储器,IO 设备;
- CPU:字长,主频(时钟频率),总线;
- 存储器:内(RAM,ROM),外存储器,种类,单位,存取速度;
- 输入输出设备:扫描仪,数字化仪,绘图仪,打印机(种类)

2>.计算机软件:

- BIOS (功能);
- 系统软件(包括操作系统:DOS,LINUX,UNIX,WINDOWS,OS/2,MAC/OS 和语言的解释或编译程序);

解释程序:高级语言翻译的一种,它将源语言(如 basic)书写的源程序作为输入,解释一句后就提交计算机执行一句,并不形成目标程序。

翻译程序:(编译程序)一类很重要的语言处理程序,它把高级语言(如 FORTRAN,COBOL,pascal,c 等)源程序作为输入,进行翻译转换,产生出机器语言的目标程序,然后再让计算机去执行这个目标程序,得到计算结果。

语言:机器语言 汇编语言 高级语言(面向对象,面向过程) c.应用软件

数据库管理软件:Foxpro, Access, Oracle, Sybase, DB2 和 Informix 等。字处理软件: WPS, word

3>.计算机的主要性能指标

- 字长
- 速度
- 存储系统容量(bit,B,KB,MB,GB,TB)

3. 数据在计算机中的表示

1>.数值的表示:二进制,八进制,十六进制,十进制(包括小数部分的转化)

原码,反码,补码的表示

2>.字符的表示: ASCII 码(128 个)

‘0’---48 ‘A’---65 ‘a’---97

汉字的表示: 2 个字节(Byte) :机内码,输入码,字型码

3>.图像的表示 4>.声音的表示

4.计算机的维护与使用安全

1>. 计算机的维护与安全使用常识

(电源,温度,湿度,开关机)

2>. 计算机病毒的预防与消除

(何谓病毒,病毒的特点,杀毒方式及软件)

第二部分 计算机网络

1.计算机网络的定义:

计算机网络,就是把分布在不同地理区域的计算机与专门的外部设备用通信线路互连成一个规模大、功能强的网络系统,从而使众多的计算机可以方便地互相传递信息,共享信息资源。

2.计算机网络名词:

ISP: 因特网服务提供商,能提供拨号上网服务、网上浏览、下载文件、收发电子邮件等服务。即为用户提供 Internet 接人和(或)Internet 信息服务的公司和机构。如“中国电信”等;

DNS: 域名服务器;

FTP: 文件传输协议; **HTTP:** 超文本传输协议; **SMTP:** 简单邮件系统传输协议;

WWW: 万维网;

POP3: 邮件传输协议

ARP: 地址解析协议

3.两种网络参考模型

OSI 开放式系统互联模型参考模型: (七层)

由下到上:物理层、数据链路层、网络层、传输层、会话层、表示层、应用层;

TCP/IP 参考模型(五层)

由下到上:物理层、数据链路层, 互联网层、传输层、应用层

4.网络软件

1>.计算机协议: (TCP/IP)

- a. **TCP : Transfer Control Protocol**,传输控制协议
- b. **IP: Internet Protocol**,网际协议
- c. 三类 IP 地址: IPV4

2>.应用软件:

5.网络硬件

(网卡, **MODEM**,光纤,双绞线,同轴电缆,无线信道)

6.网络分类

计算机网络的类型有很多, 而且有不同的分类依据。

按拓扑结构:总线型、星型、环形、树形 按地域:局域网、城域网、广域网和网间网

7.域名的表示

<http://stneng.com>

<http://open.jloj.cn>

第三部分 数据结构

1.简单数据类型:

- a 数值: **integer, real, longint**
- b 字符: **char**
- c 布尔类型: **Boolean**
- d 数组:一维,二维
- e 字符串: **string**

2.线性表 栈、 队列

3.树

二叉树、哈弗曼树

4.图

图的最小生成树、最短路径

第四部分 基本及常用算法

第五部分 问题求解

队列、栈、二叉树等数据结构、数学问题、归纳法、数列和逻辑推理、排列组合等

题型归类:

第一部分:选择题(30 分=20*1.5)

类型 1:计算机原理 类型 2:操作系统与应用软件 [HYPERLINK \l "page7"](#) 类型 3:多媒体与网络 [HYPERLINK \l "page12"](#) 类型 4:数据结构与 算法 类型 5:数学与逻辑运算

一般是比较容易得分的,不可错过!

程序设计方面的知识多是平时计算机课堂教学或课外活动中学到的,建议大家找全国计算机等级考试(一、二级)的题目做做,一般不超过二级的知识点,知识要复习的系统一些。新大纲和最近两年的考试不再考 DOS,但有 DOS 经验的选手可能会占一点便宜,因为有些题目可以根据经验判断。另外,往更高层次发展的过程中,必要的 DOS 知识和命令还是必须的。

类型 1:计算机原理:

NOIP1999:

- 1、微机内的存储器的地址是以(C)编址的。
A. 二进制位 B. 字长 C. 字节 D. 微处理器的型号
- 2、下列诸因素中,对微机工作影响最小的是(B)
A. 尘土 B. 噪声 C. 温度 D. 湿度
- 3、在 24*24 点阵的字库中,汉字“一”与“编”的字模占用字节数分别是(C)
A. 32、32 B. 32、72 C. 72、72 D. 72、32
- 7、计算机能直接执行的指令包括两部分,它们是(B)
A. 源操作数与目标操作数 B. 操作码与操作数
C. ASC II 码与汉字代码 D. 数字与字符
- 8、在微机中,通用寄存器的位数是(C)
A. 8 位 B. 16 位 C. 计算机字长 D. 32 位
- 9、在计算机,字符编码通常采用(C)
A. 原码 B. 反码 C. ASCII 码 D. 补码
- 13、已知小写字母“M”的十六进制的 ASC II 码值是 6D, 则小写字母“C”的十六进制数的 ASC II 码值是(D)
A. 98 B. 62 C. 99 D. 63
- 14、计算机中的数有浮点与定点数两种,其中用浮点数表示的数,通常由(C)这两部分组成。
A. 指数与基数 B. 尾数与小数 C. 阶码与尾数 D. 整数与小数
- 16、启动计算机引导 DOS 是将操作系统(D)
A. 从磁盘调入中央处理器 B. 从内存调入高速缓冲存储器
C. 从软盘调入硬盘 D. 从系统盘调入内存
- 18、组成“教授”(JIAO SHOU),“副教授”(FU JIAO SHOU)与“讲师”(JIANG SHI)这三个词的汉字,在 GB2312-80 字符集中都是一级汉字,对这三个词排序的结果是(D)
A. 教授、副教授、讲师 B. 副教授、教授、讲师
C. 讲师、副教授、教授 D. 副教授、讲师、教授
- 19、不同的计算机,其指令系统也不相同,这主要取决于(C)
A. 所用的操作系统 B. 系统的总体结构
C. 所用的 CPU D. 所用的程序设计语言

NOIP2000:

- 8.计算机系统总线上传送的信号有(B) A.地址信号与控制信号 B.数据信号、控制信号与地址信号

C.控制信号与数据信号 D. 数据信号与地址信号

9.计算机的运算速度取决于给定的时间内, 它的处理器所能处理的数据量。处理器一次能处理的数据量叫字长。已知 64 位的奔腾处理器一次能处理 64 个信息位, 相当于(A)字节。

A.8 个 B.1 个 C.16 个 D. 2 个

14.不同类型的存储器组成了多层次结构的存储器体系, 按存取速度从快到慢的排列是(C) A.快存/辅存/主存 B. 外存/主存/辅存 C. 快存/主存/辅存 D. 主存/辅存/外存

NOIP2001:

1、中央处理器 CPU 能访问的最大存储器容量取决于(A)

A)地址总线 B)数据总线 C)控制总线 D)内存容量

7、若我们说一个微机的 CPU 是用的 PII300, 此处的 300 确切指的是(A)

A)CPU 的主时钟频率 B)CPU 产品的系列号

C)每秒执行 300 百万条指令 D)此种 CPU 允许最大内存容量

NOIP2002:

1. 微型计算机的问世是由于(C)的出现。

A)中小规模集成电路 B)晶体管电路 C)(超)大规模集成电路 D)电子管电路

2. 中央处理器(CPU)能访问的最大存储器容量取决于(A)。

A)地址总线 B)数据总线 C)控制总线 D)实际内存容量

11. 微型计算机中, (C)的存取速度最快。

A)高速缓存 B)外存储器 C)寄存器 D)内存储器 14. 一个向量第一个元素的存储地址是 100, 每个元素的长度是 2, 则地 5 个元素的地址是(B)。

A)110 B)108 C)100 D)109

NOIP2003:

1. 图灵 (Alan Turing) 是(B)。

A) 美国人 B) 英国人 C) 德国人 D) 匈牙利人 E) 法国人

2. 第一个给计算机写程序的人是(B)。

A) Alan Mathison Turing B) Ada Lovelace C) John von Neumann

D) John Mc-Carthy E) Edsger Wybe Dijkstra

11. 下列分辨率的显示器显示出的图像, 最清晰的是(D)。

A) 800*600 B) 1024*768 C) 640*480 D) 1280*1024 E) 800*1000

12. 下列说法中, 哪个(些)是错误的(BDE)。

A)程序是指令的序列, 它有三种结构:顺序、分支和循环。

B)数据总线决定了中央处理器 CPU 所能访问的最大内存空间的大小。

C)中央处理器 CPU 内部有寄存器组, 用来储存数据。

D)不同厂家生产的 CPU 所能处理的指令集是相同的。

E)数据传输过程中可能会出错, 奇偶校验法可以检测出数据中那一为在传输中出了差错。

17. 下列哪个(些)不是个人计算机的硬件组成部分(B)。

A)主板 B)虚拟内存 C)电源 D)硬盘 E)总线

NOIP2004:

7. 下面哪个部件对于个人桌面电脑的正常运行不是必需的(C)。

A. CPU B. 图形卡(显卡) C. 光驱 D. 主板 E. 内存

11. 美籍匈牙利数学家冯·诺依曼对计算机科学发展所做出的贡献包括(BC)。

A. 提出理想计算机的数学模型, 成为计算机科学的理论基础。

B. 提出存储程序工作原理, 对现代电子计算机的发展产生深远影响。

C. 设计出第一台具有存储程序功能的计算机 EDVAC。

D. 采用集成电路作为计算机的主要功能部件。

E. 指出计算机性能将以每两年翻一番的速度向前发展。

12. 下列哪个(些)是 64 位处理器(ACDE)。

A. Intel Itanium B. Intel Pentium III C. AMD Athlon64

D. AMD Opteron E. IBM Power 5

15. 下列哪个(些)不是计算机的存储设备(AC)。

A. 文件管理器 B. 内存 C. 显卡 D. 硬盘 E. U 盘

18. 彩色显示器所显示的五彩斑斓的色彩, 是由哪三色混合而成的(ACD)。

A. 红 B. 白 C. 蓝 D. 绿 E. 橙

NOIP2005:

7. Intel 的首颗 64 位处理器是(E)。

A. 8088 B. 8086 C. 80386 D. 80486 E. Pentium

18. 以下断电之后将不能保存数据的有(BCDE)。

A. 硬盘 B. 寄存器 C. 显存 D. 内存 E. 高速缓存

20. 下列关于高级语言的说法正确的有(BDE)。

A. Ada 是历史上的第一个高级语言

B. Pascal 和 C 都是编译执行的高级语言

C. C++是历史上的第一个支持面向对象的语言

D. 编译器将高级语言程序转变为目标代码

E. 高级语言程序比汇编语言程序更容易从一种计算机移植到另一种计算机上

NOIP2006:

1. 在以下各项中。(E)不是 CPU 的组成部分。

A. 控制器 B. 运算器 C. 寄存器 D. ALU E. RAM

2. BIOS(基本输入输出系统)是一组固化在计算机内(C)上一个 ROM 芯片上的程序。

A. 控制器 B. CPU C. 主板 D. 内存条 E. 硬盘

18. 在下列关于计算机语言的说法中, 正确的有(AB)。

A. Pascal 和 C 都是编译执行的高级语言

B. 高级语言程序比汇编语言程序更容易从一种计算机移植到另一种计算机上

C. C++是历史上的第一个支持面向对象的计算机语言

D. 高级语言比汇编语言更高级, 是因为它的程序的运行效率更高

NOIP2007:

1. 在以下各项中。(D)不是 CPU 的组成部分。

A. 控制器 B. 运算器 C. 寄存器 D. 主板 E. 算术逻辑单元(ALU)

3. 在下列各项中, 只有(D)不是计算机存储容量的常用单位。

A. Byte B. KB C. MB D. UB E. TB

4. ASCII 码的含义是(B)。

A. 二—十进制转换码

B. 美国信息交换标准代码

C. 数字的二进制数码

D. 计算机可处理字符的唯一编码

E. 常用字符的二进制编码

20. 近 20 年来, 许多计算机专家都大力推崇递归算法, 认为它是解决较复杂问题的强有力的工具. 在下列关于递归的说法中, 正确的是(AC)。

A. 在 1977 年前后形成标准的计算机高级语言"FORTRAN77"禁止在程序使用递归, 原因之一是该方法可能会占用更多的内存空间。

B. 和非递归算法相比, 解决同一个问题, 递归算法一般运行得更快一些

C. 对于较复杂的问题, 用递归方式编程往往比非递归方式更容易一些

D. 对于已定义好的标准数学函数 $\sin(x)$, 应用程序中的语句“ $y=\sin(\sin(x));$ ”就是一种递归调用

NOIP2008:

1. 在以下各项中, (C)不是操作系统软件。

A. Solaris B. Linux C. Sybase D. Windows Vista E. Symbian

11. 在下列关于图灵奖的说法中, 正确的有(ABD)。

A. 图灵奖是美国计算机协会于 1966 年设立的, 专门奖励那些对计算机事业作出重要贡献的个人

B. 图灵奖有“计算机界诺贝尔奖”之称

C. 迄今为止, 还没有华裔计算机科学家获此殊荣

D. 图灵奖的名称取自计算机科学的先驱、英国科学家阿兰·图灵

NOIP2009:

2、关于 BIOS 下面的说法哪个是正确的:A

- A) BIOS 是计算机基本输入输出系统软件的简称。
- B) BIOS 里包含了键盘、鼠标、声卡、图形界面显示器等常用输入输出设备的驱动程序。
- C) BIOS 一般由操作系统厂商来开发完成。
- D) BIOS 能提供各种文件拷贝、复制、删除以及目录维护等文件管理功能。

3、已知大写字母 A 的 ASCII 编码为 65(十进制), 则大写字母 J 的 十六进制 ASCII 编码为:D A)
48 B) 49 C) 50 D) 以上都不是

类型 2:操作系统与应用软件:

NOIP1999:

10、计算机的软件系统通常分为(A)

- A. 系统软件与应用软件
- B. 高级软件与一般软件
- C. 军用软件与民用软件
- D. 管理软件与控制软件

NOIP2000:

4.计算机病毒的特点是(C)

- A. 传播性、潜伏性、易读性与隐蔽性
- B. 破坏性、传播性、潜伏性与安全性
- C. 传播性、潜伏性、破坏性与隐蔽性
- D. 传播性、潜伏性、破坏性与易读性

5.WINDOWS 9X 是一种(D)操作系统

- A. 单任务字符方式
- B. 单任务图形方式
- C. 多任务字符方式
- D. 多任务图形方式

7.计算机网络是一个(D)系统

- A.管理信息系统
- B.管理数据系统
- C.编译系统
- D.在协议控制下的多机互连系统

NOIP2001:

4、在树型目录结构中, 不允许两个文件名相同主要指的是(D) A)

- 同一个磁盘的不同目录下 B)不同磁盘的同一个目录下
- C)不同磁盘的不同目录下 C)同一个磁盘的同一个目录下

10、以下对 Windows 的叙述中, 正确的是(A) A)

- 从软盘上删除的文件和文件夹, 不送到回收站
- B)在同一个文件夹中, 可以创建两个同类、同名的文件
- C)删除了某个应用程序的快捷方式, 将删除该应用程序对应的文件
- D)不能打开两个写字板应用程序

NOIP2002:

7. 计算机病毒传染的必要条件是:(B)。

- A)在内存中运行病毒程序
- B)对磁盘进行读写操作
- C)在内存中运行含有病毒的可执行的程序 D)复制文件

8. 在磁盘上建立子目录有许多优点, 下列描述中不属于建立子目录优点的是(D)。

- A)便于文件管理
- B)解决根目录中目录项个数有限问题
- C)加快文件查找速度 D)节省磁盘使用空间

12. 资源管理器的目录前图标中增加“+”号, 这个符号的意思是(B)。

- A)该目录下的子目录已经展开 B)该目录下还有子目录未展开
- C)该目录下没有子目录 D)该目录为空目录

13. 在 WORD 文档编辑中实现图文混合排版时, 关于文本框的下列叙述正确的是(C)。

- A)文本框中的图形没有办法和文档中输入文字叠加在一起, 只能在文档的不同位置
- B)文本框中的图形不可以衬于文档中输入的文字的下方
- C)通过文本框, 可以实现图形和文档中输入的文字的叠加, 也可以实现文字环绕
- D)将图形放入文本框后, 文档中输入的文字不能环绕图形

NOIP2004:

14. 下列哪个(些)不是数据库软件的名称(D)。

A. MySQL B. SQL Server C. Oracle D. Outlook E. Foxpro

16. 下列哪个(些)软件属于操作系统软件(BE)。

A. Microsoft Word B. Windows XPC. Foxmail D. 金山影霸 E. Red Hat Linux

19. 下列哪个(些)程序设计语言支持面向对象程序设计方法(ABDE)。

A. C++ B. Object Pascal C. CD. Smalltalk E. Java

NOIP2006:

15. 下列外设接口中可以通过无线连接的方式连接设备的是(ABCD)。

A. USB 2.0 高速版 B. 红外 C. 蓝牙 D. 串口 E. IEEE 802.11g 无线网卡

类型 3: 多媒体与网络:

NOIP2000:

11. 下面哪些计算机网络不是按覆盖地域划分的(A) A. 局域网 B. 都市网 C. 广域网 D. 星型网

NOIP2001:

12. TCP/IP 协议共有(C)层协议

A)3 B)4 C)5 D)6

NOIP2002:

9. 在使用 E-mail 前, 需要对 Outlook 进行设置, 其中 ISP 接收电子邮件的服务器称为(A)服务器。

A)POP3 B)SMTP C)DNS D)FTP 10. 多媒体计算机是指(D)计算机。

A)专供家庭使用的 B)装有 CD-ROM 的

NOIP2004:

8. 下列哪个网络上常用的名字缩写是错误的(D)。

A. WWW(World Wide Web)

B. URL(Uniform Resource Locator)

C. HTTP(Hypertext Transfer Protocol)

D. FTP(Fast Transfer Protocol)

E. TCP(Transfer Control Protocol)。

10. 一台计算机如果要利用电话线上网, 就必须配置能够对数字信号和模拟信号进行相互转换的设备, 这种设备是(A)。

A. 调制解调器 B. 路由器 C. 网卡 D. 网关 E. 网桥

NOIP2005:

8. 常见的邮件传输服务器使用(B)协议发送邮件。

A. HTTP B. SMTP C. TCP D. FTP E. POP3

9. 不能在 Linux 上使用的网页浏览器是(A)。

A. Internet Explore B. Netscape C. Opera D. Firefox E. Mozilla

NOIP2008:

14. Web2.0 是近年来互联网的热门概念之一, 其核心思想是互动与分享。下列网站中, (B)是典型的 Web2.0 应用。

A. Sina B. Flickr C. Yahoo D. Google 4. 关于计算机网络, 下面的说法哪些是正确的: C

A) 网络协议之所以有很多层主要是由于新技术需要兼容过去老的实现方案。

B) 新一代互联网使用的 IPv6 标准是 IPv5 标准的升级与补充。

C) TCP/IP 是互联网的基础协议簇, 包含有 TCP 和 IP 等网络与传输层的通讯协议。

D) 互联网上每一台入网主机通常都需要使用一个唯一的 IP 地址, 否则就必须注册一个固定的域名来标明其地址。

5. 关于 HTML 下面哪些说法是正确的: BD

- A) HTML 全称超文本标记语言, 实现了文本、图形、声音乃至视频信息的统一编码。
- B) HTML 不单包含有网页内容信息的描述, 同时也包含对网页格式信息的定义。
- C) 网页上的超链接只能指向外部的网络资源, 本网站网页间的联系通过设置标签来实现。
- D) 点击网页上的超链接从本质上就是按照该链接所隐含的统一资源定位符(URL)请求网络资源或网络服务。

类型 4: 数据结构与算法:

NOIP2000:

12. 在有 N 个叶子节点的哈夫曼树中, 其节点总数为(B) A. 不确定 B. $2N-1$ C. $2N+1$ D. $2N$

解法一: 设叶子节点个数为 n , 度为 1 的节点个数为 m , 度为 2 的节点个数为 l .

显然易知: 一颗二叉树的节点数 = 这个树的度加 1 (因为每个节点都是前一个节点的度, 根节点除外, 所以要加 1)

$$\text{故有 } l + m + n = 2l + m + 1$$

$$\text{----> } n = l + 1$$

由于哈夫曼树没有度为 1 的节点, 在 $m = 0$

$$\text{总节点} = n + m + l = 2n - 1$$

解法二: 第 1 次必定是 2 个叶子组成二叉树, 产生 1 新结点, 接下来有 2 种情况:

1. 此新结点与原剩下的叶子再组成二叉树又产生 1 新结点, 这样就只有第 1 次时由 2 个叶子产生 1 新结点, 以后每次由 1 叶子与新结点产生新结点, 故 n 个叶子共有 $2n-1$ 个结点。

2. 剩下的叶子中又有 2 个叶子(比第 1 次产生的新结点权小)结合产生新结点, 其它类似, 那么必然会由 2 个都是新结点再产生新结点, 所以实际上数量与第 1 种一样, 共有 $2n-1$ 个。

具体证明用一个构造哈夫曼树的算法。

13. 已知数组中 A 中, 每个元素 $A(I, J)$ 在存贮时要占 3 个字节, 设 I 从 1 变化到 8, J 从 1 变化到 10, 分配内存时是从地址 SA 开始连续按行存贮分配的。

试问: $A(5, 8)$ 的起始地址为(A)

- A. $SA+141$ B. $SA+180$ C. $SA+222$ D. $SA+225$

15. 某数列有 1000 个各不相同的单元, 由低至高按序排列; 现要对该数列进行二分法检索(binary-search), 在最坏的情况下, 需检视(B)个单元。

- A. 1000 B. 10 C. 100 D. 500

NOIP2001:

13. 若已知一个栈的入栈顺序是 1, 2, 3, ..., n , 其输出序列为 $P_1, P_2, P_3, \dots, P_n$, 若 P_1 是 n , 则 P_i 是(C) A) i B) $n-1$ C) $n-i+1$ D) 不确定

15. 下面关于算法的错误说法是(B)

- A) 算法必须有输出 B) 算法必须在计算机上用某种语言实现
- C) 算法不一定有输入 D) 算法必须在有限步执行后能结束

17. 以下哪一个不是栈的基本运算(B) A)

- 删除栈顶元素 B) 删除栈底的元素
- C) 判断栈是否为空 D) 将栈置为空栈

18. 在顺序表(2, 5, 7, 10, 14, 15, 18, 23, 35, 41, 52)中, 用二分法查找 12, 所需的关键码比较的次数为(C) A) 2 B) 3 C) 4 D) 5

19. 一棵二叉树的高度为 h , 所有结点的度为 0, 或为 2, 则此树最少有(B)个结点

- A) $2h-1$ B) $2h-1$ C) $2h+1$ D) $h+1$

20. 无向图 $G=(V, E)$, 其中 $V=\{a, b, c, d, e, f\}$ $E=\{(a, b), (a, e), (a, c), (b, e), (c, f), (f, d), (e, d)\}$

对该图进行深度优先遍历, 得到的顶点序列正确的是(D)

- A) a, b, e, c, d, f B) a, c, f, e, b, d C) a, e, b, c, f, d D) a, b, e, d, f, c

NOIP2002:

17. 按照二叉数的定义, 具有 3 个结点的二叉树有(C)种。

A)3 B)4 C)5 D)6 18. 在一个有向图中, 所有顶点的入度之和等于所有顶点的出度之和的(B)倍。

A) 1/2 B)1 C)2 D)4

解析: 在有向图的邻接表中, 从一顶点出发的弧链接在同一链表中, 邻接表中结点的个数恰为图中弧的数目, 所以顶点入度之和为弧数和的一倍, 若为无向图, 同一条边有两个结点, 分别出现在和它相关的两个顶点的链表中, 因此无向图的邻接表中结点个数的边数的 2 倍

19. 要使 1 . . . 8 号格字的访问顺序为:8、2、6、5、7、3、1、4, 则下图中的空格中应填入(C)。

1 2 3 4 5 6 7 8
4 6 1 -1 7 3 2

A)6 B)0 C)5 D)3

NOIP2003:

5. 一个高度为 h 的二叉树最小元素数目是(B)。

A) $2h+1$ B) h C) $2h-1$ D) $2h$ E) $2h-1$

6. 已知队列(13, 2, 11, 34, 41, 77, 5, 7, 18, 26, 15), 第一个进入队列的元素是 13, 则第五个出队列的元素是(B)。

A) 5 B) 41 C) 77 D) 13 E) 18

19. 已知元素(8, 25, 14, 87, 51, 90, 6, 19, 20), 问这些元素以怎样的顺序进入栈, 才能使出栈的顺序满足:8 在 51 前面;90 在 87 的后面;20 在 14 的后面;25 在 6 的前面;19 在 90 的后面。(D)。

A)20, 6, 8, 51, 90, 25, 14, 19, 87

B)51, 6, 19, 20, 14, 8, 87, 90, 25

C)19, 20, 90, 8, 6, 25, 51, 14, 87

D)6, 25, 51, 8, 20, 19, 90, 87, 14

E)25, 6, 8, 51, 87, 90, 19, 14, 20

20. 假设我们用 $d=(a_1, a_2, \dots, a_5)$ 表示无向图 G 的 5 个顶点的度数, 下面给出的哪(些)组 d 值合理(BE)。

A) {5, 4, 4, 3, 1} B) {4, 2, 2, 1, 1} C) {3, 3, 3, 2, 2} D) {5, 4, 3, 2, 1} E) {2, 2, 2, 2, 2}

NOIP2004:

3. 某个车站呈狭长形, 宽度只能容下一台车, 并且只有一个出入口。已知某时刻该车站状态为空, 从这一时刻开始的出入记录为:“进, 出, 进, 进, 出, 进, 进, 进, 出, 出, 进, 出”。假设车辆入站的顺序为 1, 2, 3, ..., 则车辆出站的顺序为(E)。

A. 1, 2, 3, 4, 5 B. 1, 2, 4, 5, 7 C. 1, 3, 5, 4, 6 D. 1, 3, 5, 6, 7 E. 1, 3, 6, 5, 7

4. 满二叉树的叶结点个数为 N, 则它的结点总数为(C)。

A. N B. $2 * N$ C. $2 * N - 1$ D. $2 * N + 1$ E. $2N - 1$

5. 二叉树 T, 已知其前序遍历序列为 1 2 4 3 5 7 6, 中序遍历序列为 4 2 1 5 7 3 6, 则其后序遍历序列为(B)。

A. 4 2 5 7 6 3 1 B. 4 2 7 5 6 3 1 C. 4 2 7 5 3 6 1 D. 4 7 2 3 5 6 1 E. 4 5 2 6 3 7 1

20. 某大学计算机专业的必修课及其先修课程如下表所示:

课程代号	C_0	C_1	C_2	C_3	C_4	C_5	C_6	C_7
课程名称	高等数学	程序设计语言	离散数学	数据结构	编译技术	操作系统	普通物理	计算机原理
先修课程			C_0, C_1	C_1, C_2	C_3	C_3, C_7	C_0	C_6

请你判断下列课程安排方案哪个(些)是合理的(BCE)。

A. $C_0, C_1, C_2, C_3, C_4, C_5, C_6, C_7$ B. $C_0, C_1, C_2, C_3, C_4, C_6, C_7, C_5$

C. $C_0, C_1, C_6, C_7, C_2, C_3, C_4, C_5$ D. $C_0, C_1, C_6, C_7, C_5, C_2, C_3, C_4$

E. $C_0, C_1, C_2, C_3, C_6, C_7, C_5, C_4$

NOIP2005:

4. 完全二叉树的结点个数为 $4 * N + 3$, 则它的叶结点个数为(E)。

A. $2 * N$ B. $2 * N - 1$ C. $2 * N + 1$ D. $2 * N - 2$ E. $2 * N + 2$

5. 平面上有五个点 A(5, 3), B(3, 5), C(2, 1), D(3, 3), E(5, 1)。以这五点作为完全图 G 的顶点, 每两点之间的直线距离是图 G 中对应边的权值。图 G 的最小生成树中的所有边的权值 综合为(D)。

A. 8 B. 7+ 5 C. 9 D. 6+ 5 E. 4+2 2 + 5

13. 二叉树 T 的宽度优先遍历序列为 A B C D E F G H I, 已知 A 是 C 的父结点, D 是 G 的父结点, F 是 I 的父结点, 树中所有结点的最大深度为 3(根结点深度设为 0), 可知 E 的父结点可能是(BC)。

A. A B. B C. C D. D E. F

14. 设栈 S 的初始状态为空, 元素 a, b, c, d, e, f, g 依次入栈, 以下出栈序列不可能出现的有(CE)。

A. a, b, c, e, d, f, g B. b, c, a, f, e, g, d C. a, e, c, b, d, f, g

D. d, c, f, e, b, a, g E. g, e, f, d, c, b, a

NOIP2006:

4. 在编程时(使用任一种高级语言, 不一定是 Pascal), 如果需要从磁盘文件中输入一个很大的二维 数组(例如 1000*1000 的 double 型数组), 按行读(即外层循环是关于行的)与按列读(即外层循 环是关于列的)相比, 在输入效率上(E)。

A. 没有区别 B. 有一些区别, 但机器处理速度很快, 可忽略不计

C. 按行读的方式要高一些 D. 按列读的方式要高一些 E. 取决于数组的存储方式。

7. 某个车站呈狭长形, 宽度只能容下一台车, 并且只有一个出入口。已知某时刻该车站状态为空, 从 这一时 刻开始的出入记录为:“进, 出, 进, 进, 进, 出, 出, 进, 进, 进, 出, 出”。假设车辆入站的 顺序为 1, 2, 3, ..., 则车辆出站的顺序为(C)。

A. 1, 2, 3, 4, 5 B. 1, 2, 4, 5, 7 C. 1, 4, 3, 7, 6

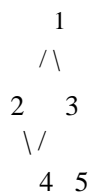
D. 1, 4, 3, 7, 2 E. 1, 4, 3, 7, 5

8. 高度为 n 的均衡的二叉树是指: 如果去掉叶结点及相应的树枝, 它应该是高度为 n-1 的满二叉树。在这里, 树高等于叶结点的最大深度, 根结点的深度为 0, 如果某个均衡的二叉树共有 2381 个结点, 则该树的树高为(B)。

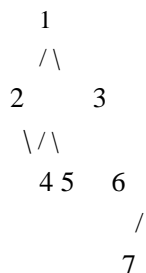
A. 10 B. 11 C. 12 D. 13 E. $2^{10} - 1$

解析: 均衡二叉树就是: 任意两个度不为 2 的节点的深度之差 不大于 1

例如:



是均衡二叉树
而



就不是, 2 和 7 的深度差 2.

因为 $2^{11} = 2048$; 所以一颗满二叉树从深度为 0(根节点)到深度 10 的总节点数是 2047, 剩下 $2381 - 2047 = 334$ 个节点, 这剩下的节点的深度都是 11。

所以答案为 B

10. 将 5 个数的序列排序, 不论原先的顺序如何, 最少都可以通过(B)次比较, 完成从小到大的排序。

A. 6 B. 7 C. 8 D. 9 E. 10

13. 设栈 S 的初始状态为空, 元素 a, b, c, d, e 依次入栈, 以下出栈序列不可能出现的有(C)。

A. a, b, c, e, d B. b, c, a, e, d

C. a, e, c, b, d D. d, c, e, b, a

14. 已知 6 个结点的二叉树的先根遍历是 1 2 3 4 5 6(数字为结点的编号, 以下同), 后根遍历是

3 2 5 6 4 1, 则该二叉树的可能的中根遍历是(BC)

A. 3 2 1 4 6 5 B. 3 2 1 5 4 6

C. 2 3 1 5 4 6 D. 2 3 1 4 6 5

NOIP2007:

2. 在关系数据库中, 存放在数据库中的数据逻辑结构以(E)为主。

A. 二叉树 B. 多叉树 C. 哈希表 D. B+树 E. 二维表

9. 欧拉图 G 是指可以构成一个闭回路的图, 且图 G 的每一条边恰好在这个闭回路上出现一次(即一笔画成)。

在以下各个描述中, 不一定是欧拉图的是:(D)。

A. 图 G 中没有度为奇数的顶点

B. 包括欧拉环游的图(欧拉环游是指通过图中每边恰好一次的闭路径)

C. 包括欧拉闭迹的图(欧拉迹是指通过图中每边恰好一次的路径)

D. 存在一条回路, 通过每个顶点恰好一次

E. 本身为闭迹的图

14. 已知 7 个节点的二叉树的先根遍历是 1 2 4 5 6 3 7(数字为结点的编号, 以下同), 后根遍历是 4 6 5 2 7 3 1,

则该二叉树的可能的中根遍历是(ABD)

A. 4 2 6 5 1 7 3

B. 4 2 5 6 1 3 7

C. 4 2 3 1 5 4 7

D. 4 2 5 6 1 7 3

19. 在下列关于算法复杂性的说法中, 正确的有(BC)。

A. 算法的时间复杂度, 是指它在某台计算机上具体实现时的运行时间

B. 算法的时间复杂度, 是指对于该算法的一种或几种主要的运算, 运算的次数与问题的规模之间的函数关系

C. 一个问题如果是 NPC 类的, 就意味着在解决该问题时, 不存在一个具有多项式时间复杂度的算法. 但这一点还没有得到理论上证实, 也没有被否定

D. 一个问题如果是 NP 类的, 与 C 有相同的结论 由 X2Studio.Net 收集

5. 将数组{8, 23, 4, 16, 77, -5, 53, 100}中的元素按从大到小的顺序排列, 每次可以交换任意两个元素, 最少需要交换(B)次。

A. 4

B. 5

C. 6

D. 7

E. 8

6. 设栈 S 的初始状态为空, 元素 a, b, c, d, e, f 依次入栈 S, 出栈的序列为 b, d, c, f, e, a, 则栈 S 的容量至少应该是(D)。

A. 6

B. 5

C. 4

D. 3

E. 2

18. 设 T 是一棵有 n 个顶点的树, 下列说法正确的是(ABC)。

A. T 是连通的、无环的

B. T 是连通的, 有 n-1 条边

C. T 是无环的, 有 n-1 条边

D. 以上都不对

NOIP2009:

5. 一个包含 n 个分支结点(非叶结点)的非空满 k 叉树, $k \geq 1$, 它的叶结点数目为: D A)

$nk + 1$ B) $nk - 1$ C) $(k+1)n - 1$ D. $(k-1)n + 1$

7. 最优前缀编码, 也称 Huffman 编码。这种编码组合的特点是对于较频繁使用的元素给与较短的唯一编码, 以提高通讯的效率。下面编码组合哪一组不是合法的前缀编码。 B

A) (00, 01, 10, 11) B)

(0, 1, 00, 11) C) (0, 10,

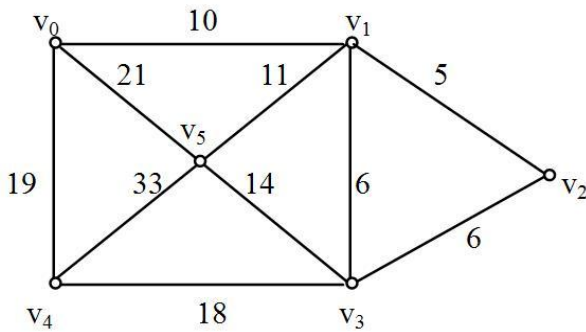
110, 111) D) (1, 01,

000, 001)

8. 快速排序平均情况和最坏情况下的算法时间复杂度分别为: A

A) 平均情况 $O(n \log_2 n)$, 最坏情况 $O(n^2)$

- B) 平均情况 $O(n)$, 最坏情况 $O(n^2)$
 C) 平均情况 $O(n)$, 最坏情况 $O(n \log 2n)$
 D) 平均情况 $O(\log 2n)$, 最坏情况 $O(n^2)$



9、左图给出了一个加权无向图，从顶点 V_0 开始用 prim 算法求最小生成树。则依次加入最小生成树的顶点集合的顶点序列为:A

- A) $V_0, V_1, V_2, V_3, V_5, V_4$
 B) $V_0, V_1, V_5, V_4, V_3, V_2$
 C) $V_1, V_2, V_3, V_0, V_5, V_4$
 D) $V_1, V_2, V_3, V_0, V_4, V_5$

6、若 3 个顶点的无权图 G 的邻接矩阵用数组存储为 $\{\{0, 1, 1\}, \{1, 0, 1\}, \{0, 1, 0\}\}$ ，假定在具体存储中 顶点依次为: v_1, v_2, v_3 。关于该图，下面的说法哪些是正确的:ABD

- A) 该图是有向图。
 B) 该图是强连通的。
 C) 该图所有顶点的入度之和减所有顶点的出度之和等于 1。
 D) 从 v_1 开始的深度优先遍历所经过的顶点序列与广度优先的顶点序列是相同的。

8、散列表的地址区间为 0-10,散列函数为 $H(K)=K \bmod 11$ 。采用开地址法的线性探查法处理冲突，并将关键字序列 26, 25, 72, 38, 8, 18, 59 存储到散列表中，这些元素存入散列表的顺序并不确定。假定之前散列表为 空，则元素 59 存放在散列表中的可能地址有:ABC

- A) 5 B) 7 C) 9 D) 10

类型 5: 数学与逻辑运算:

NOIP1999:

17、十进制算术表达式 $:3*512 + 7*64 + 4*8 + 5$ 的运算结果，用二进制表示为(B)

- A. 10111100101B. 11111100101 C. 11110100101D. 11111101101

NOIP2000:

1. 下列无符号数中，最小的数是(C)

- A. (11011001)₂ B. (75)₁₀ C. (37)₈ D. (2A)₁₆

10. 某种计算机的内存容量是 640K，这里的 640K 容量是指(C)个字节

- A. 640B. 640*1000C. 640*1024 D. 640*1024*1024

NOIP2001:

3、64KB 的存储器用十六进制表示，它的最大的地址码是(B)

- A) 10000 B) FFFF C) 1FFFF D) EFFFF

9、2KB 的内存能存储()个汉字的机内码 A

- A) 1024 B) 516 C) 2048 D) 218

11、运算式 $(2047)_{10} - (3FF)_{16} + (2000)_8$ 的结果是(A)

- A) (2048)₁₀ B) (2049)₁₀ C) (3746)₈ D) (1AF7)₁₆

16. $[x]$ 补码 = 10011000，其原码为(B)

- A) 011001111 B) 11101000 C) 11100110 D) 01100101

NOIP2002:

3. 十进制书 11/128 可用二进制数码序列表示为: (D)。

- A) 1011/1000000 B) 1011/100000000 C) 0.001011 D) 0.0001011

4. 算式 $(2047)_{10} - (3FF)_{16} + (2000)_8$ 的结果是(A)。

- A) (2048)₁₀ B) (2049)₁₀ C) (3746)₈ D) (1AF7)₁₆

5. 已知 $x = (0.1011010)_2$ ，则 $[x/2]$ 补 = (C)₂。

- A) 0.1011101 B) 11110110 C) 0.0101101 D) 0.100110

C)连接在网络上的高级 D)具有处理文字、图形、声音、影像等信息的

15. 已知 $A = 35H$, $A \wedge 05H \vee A \wedge 30H$ 的结果是: (C)。 A) 30H B) 05H C) 35H D) 53H

NOIP2003:

3. 十进制数 2003 等值于二进制数 (D)。
A) 0100000111 B) 10000011 C) 110000111 D) 1111010011 E) 1111010011
4. 假设 $A=true, B=false, C=true, D=true$, 逻辑运算表达式 $A \wedge B \vee C \wedge D$ 的值是 (A)。
A) true B) false C) 0 D) 1 E) NULL
8. 设全集 $E=\{1, 2, 3, 4, 5\}$, 集合 $A=\{1, 4\}$, $B=\{1, 2, 5\}$, $C=\{2, 4\}$, 则集合 $(A \cap B) \cup \sim C$ 为 (E)。
A) 空集 B) $\{1\}$ C) $\{3, 5\}$ D) $\{1, 5\}$ E) $\{1, 3, 5\}$
18. 运算 $(2008)_{10} - (3723)_8$ 的结果是 (BCD)。
A) $(-1715)_{10}$ B) $(5)_{10}$ C) $(5)_{16}$ D) $(101)_2$ E) $(3263)_8$

NOIP2004:

1. 设全集 $I = \{a, b, c, d, e, f, g\}$, 集合 $A = \{a, b, c\}$, $B = \{b, d, e\}$, $C = \{e, f, g\}$, 那么集合 为 (A)。
A. $\{a, b, c, d\}$ B. $\{a, b, d, e\}$ C. $\{b, d, e\}$ D. $\{b, c, d, e\}$ E. $\{d, f, g\}$
2. 由 3 个 a, 5 个 b 和 2 个 c 构成的所有字符串中, 包含子串“abc”的共有 (D) 个。
A. 40320 B. 39600 C. 840 D. 780 E. 60
13. $(2004)_{10} + (32)_{16}$ 的结果是 (BCD)。
A. $(2036)_{16}$ B. $(2054)_{10}$ C. $(4006)_8$ D. $(100000000110)_2$ E. $(2036)_{10}$

NOIP2005:

1. 字符串“ababacbab”和字符串“abcba”的最长公共子串是 (B)。
A. abcba B. cba C. abc D. ab E. bcba
2. 设全集 $I = \{a, b, c, d, e, f, g, h\}$, 集合 $BA = \{a, b, c, d, e, f\}$, $CA = \{c, d, e\}$, $BA \sim = \{a, d\}$, 那么集合 $CB A$ 为 (A)。
A. $\{c, e\}$ B. $\{d, e\}$ C. $\{e\}$ D. $\{c, d, e\}$ E. $\{d, f\}$
3. 以下二进制数的值与十进制数 23.456 的值最接近的是 (D)。
A. 10111.0101 B. 11011.1111 C. 11011.0111 D. 10111.0111 E. 10111.1111
11. 设 $A = true, B = false, C = false, D = true$, 以下逻辑运算表达式值为真的有 (BC)。
A. $(A \wedge B) \vee (C \wedge D)$ B. $((A \wedge B) \vee C) \wedge D$ C. $A \wedge ((B \vee C) \vee D)$ E. $D \wedge (A \vee (B \vee C))$ D. $(A \wedge (B \vee C)) \wedge D \wedge E$

NOIP2006:

5. 在 Pascal 语言中, 表达式 $(21 \text{ xor } 2)$ 的值是 (C)
A. 441 B. 42 C. 23 D. 24 E. 25
11. 设 $A=B=D=true, C=E=false$, 以下逻辑运算表达式值为真的有 (ABC)。
A. $(A \wedge B) \vee (C \wedge D) \vee E$ B. $((A \wedge B) \vee C) \wedge D \wedge E$
C. $A \wedge (B \vee C \vee D \vee E)$ D. $(A \wedge (B \vee C)) \wedge D \wedge E$
12. $(2010)_{16} + (32)_8$ 的结果是 (AB)。
A. $(8234)_{10}$ B. $(202A)_{16}$
C. $(100000000110)_2$ D. $(2042)_{16}$

NOIP2007:

6. 在 Pascal 语言中, 判断整数 a 等于 0 或 b 等于 0 或 c 等于 0 的正确的条件表达式是 (B)
A. $\text{not } ((a <> 0) \text{ or } (b <> 0) \text{ or } (c <> 0))$
B. $\text{not } ((a <> 0) \text{ and } (b <> 0) \text{ and } (c <> 0))$
C. $\text{not } ((a=0) \text{ and } (b=0)) \text{ or } (c=0)$
D. $(a=0) \text{ and } (b=0) \text{ and } (c=0)$
E. $\text{not } ((a=0) \text{ or } (b=0) \text{ or } (c=0))$
8. 与十进制数 17.5625 相对应的 8 进制数是 (B)。
A. 21.5625 B. 21.44 C. 21.73

D. 21.731

E. 前 4 个答案都不对

11. 设 $A=B=\text{true}$, $C=D=\text{false}$, 以下逻辑运算表达式值为真的有(ABC)。

A. $(\neg A \wedge B) \vee (C \wedge D \vee A)$

B. $\neg(((A \wedge B) \vee C) \wedge D)$

C. $A \wedge (B \vee C \vee D) \vee D$

D. $(A \wedge (D \vee C)) \wedge B$

12. 命题“ $P \rightarrow Q$ ”可读做 P 蕴含 Q, 其中 P、Q 是两个独立的命题. 只有当命题 P 成立而命题 Q 不成立时, 命题“ $P \rightarrow Q$ ”的值为 false, 其它情况均为 true. 与命题“ $P \rightarrow Q$ ”等价的逻辑关系式是(AD)。

A. $\neg P \vee Q$

B. $P \wedge Q$

C. $\neg(P \vee Q)$

D. $\neg(\neg Q \wedge P)$

13. $(2070)_{16} + (34)_8$ 的结果是(ABD)。

A. $(8332)_{10}$

B. $(208C)_{16}$

C. $(100000000110)_2$

D. $(20214)_8$

NOIP2008:

3. 设字符串 $S = \text{"Olympic"}$, S 的非空子串的数目是(B)。

A. 29

B. 28

C. 16

D. 17

E. 7

13. 设 $A=\text{true}$, $B=\text{false}$, $C=\text{true}$, $D=\text{false}$, 以下逻辑运算表达式值为真的有(BC)。

A. $(A \wedge B) \vee (C \wedge D \vee A)$

B. $((A \wedge B) \vee C) \wedge D$

C. $(B \vee C \vee D) \vee D \wedge A$

D. $A \wedge (D \vee C) \wedge B$

15. $(2008)_{10} + (5B)_{16}$ 的结果是(ABC)。

A. $(833)_{16}$

B. $(2099)_{10}$

C. $(4063)_8$

D. $(100001100011)_2$

NOIP2009:

4. 在字长为 16 位的系统环境下, 一个 16 位带符号整数的二进制补码为 111111111101101。其对应的十进制整数应该是: B

A) 19

B) -19

C) 18

D) -18

第二部分: 问题求解(2*5=10 分)

这部分题目对数学要求要高一点, 往往考查的是数学问题(如代数变形、组合、概论统计等), 数列(一般是考递推、递归、归纳法等), 逻辑推理; 也考查一些算法和数据结构知识(如队列、栈、二叉树)。建议大家多花一点时间做, 尽量做对。

1. 数组 $A[30..100, 20..100]$ 以行优先的方式存储, 每个元素占 8 个字节, 且已知 $A[40, 30]$ 的地址为 20000, 则 $A[60, 90]$ 的地址为: _____。如果以列优先存储, 则为: _____。

解答: 设数组首地址为 X, 则: $X + ((40-30) * (100-20+1) + (30-20)) * 8 = 20000$

前面的行数 每行的列数 本行中序号 每个元素的基

则: $X = 13340$, 用上述的式子, 不难算出:

$A[60, 90]$ 的地址: 33340

列优先存储, 只要稍微改动上述公式, 结果略;

➤ 考查了数据结构中数组存储方式。还可以考数组基类型为记录的情况, 可以问你同样的问题; 或者问你共 占用多少空间!

2. (1998 年初中组) 设栈 S 的初始状态为空, 现有 5 个元素组成的序列 {1, 2, 3, 4, 5}, 对该序列在 S 栈上依次进行如下操作(从序列中的 1 开始, 出栈后不在进栈): 进栈, 进栈, 进栈, 出栈, 进栈, 出栈, 进栈, 问出栈的元素序列是: _____, 栈顶指针的值为 _____, 栈顶元素为: _____。

解答: 出栈序列为 {3, 4}, 栈顶指针值为 3, 栈顶元素为 5。

➤ 考查了数据结构中的栈。还可以把栈和队列结合起来考! 如下题:

3. 如 2002 年高中组: 设栈 S 和队列 Q 初始状态为空, 元素 $e_1, e_2, e_3, e_4, e_5, e_6$ 依次通过栈 S, 一个元素出栈后即进入队列 Q, 若出队顺序为 $e_2, e_4, e_3, e_6, e_5, e_1$, 则栈 S 的容量至少应该为 _____。

解答: 为 3。

4. (2000 年初中组) 设循环队列中数组的下标范围是 $1..n$, 其头尾指针分别为 f 和 r , 则其元素个数为:

解答: $(r-f+n) \bmod n$

➤ 考查了数据结构中的队列。

5. 中缀表达式、前缀表达式、后缀表达式 (1997 年初中组)

(1) 已知中缀表达式: $A+B*C/D$

求它的前缀表达式和后缀表达式?

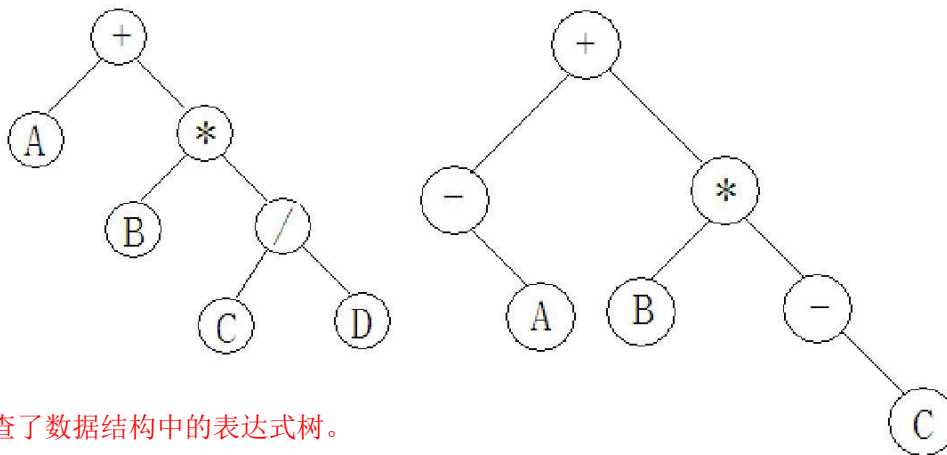
(2) 已知前缀表达式: $+\Delta A*B\Delta C$ {注 Δ 表示一元运算符负号, 即 ΔA 表示 $-A$ }

求它的中缀表达式和后缀表达式?

解答: 画(二叉)表达式树。

(1) 的结果: $+A*B/CD$; $ABCD/*+$

(2) 的结果: $(-A)+B*(-C)$; $A\Delta BC\Delta *+$



➤ 考查了数据结构中的表达式树。

6. (1998 年初中组) 已知一个数列 $U_1, U_2, U_3, \dots, U_n, \dots$, 往往可以找到一个最小的 K 值和 K 个数 a_1, a_2, \dots, a_k , 使得数列从某项开始都满足: $U(n+k) = a_1 * U(n+k-1) + a_2 * U(n+k-2) + \dots + a_k * U(n)$ (式 A)

例如数列 $1, 1, 2, 3, 5, \dots$ 可以发现: 当 $K=2, a_1=1, a_2=1$ 时, 从第 3 项起 ($N \geq 1$) 满足:

$$U(n+2) = U(n+1) + U_n$$

试对数列 $1^3, 2^3, 3^3, \dots, N^3, \dots$, 求 K 和 a_1, a_2, \dots, a_k , 使得式 A 成立。解答: 解

方程, 先设 $K=2$, 列出方程组:

$$a_1 * 1^2 + a_2 * 2^2 = 3^2$$

$$a_1 * 2^2 + a_2 * 3^2 = 4^2$$

以上方程组无整数解。再设 $K=3$, 列出方程组:

$$a_1 * 0^2 + a_2 * 1^2 + a_3 * 2^2 = 3^2$$

$$a_1 * 1^2 + a_2 * 2^2 + a_3 * 3^2 = 4^2$$

$$a_1 * 2^2 + a_2 * 3^2 + a_3 * 4^2 = 5^2$$

以上方程的整数解为: $a_1=1, a_2=-3, a_3=3$, 此时 $K=3$ 。

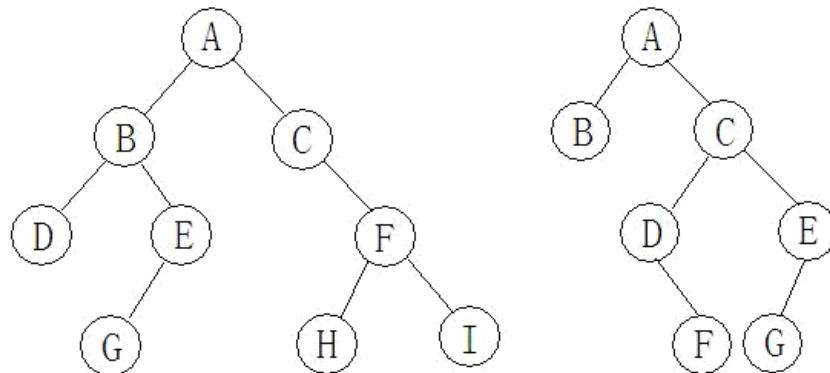
➤ 实质是考数学。

7. (1998 年高中组) 给出一棵二叉树的中序遍历: DBGEACHFI 与后序遍历: DGEBHIFCA, 画出此二叉树。

8. (1996 年高中组) 下面是一个利用完全二叉树特性, 用顺序表来存储的一个二叉树, 结点数据为字符型(结点层次从小到大, 同一层从左到右顺序存储, #表示空结点, @表示存储数据结束)。

结点	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
数据	A	B	C	#	#	D	E	#	#	#	#	#	G	F	@

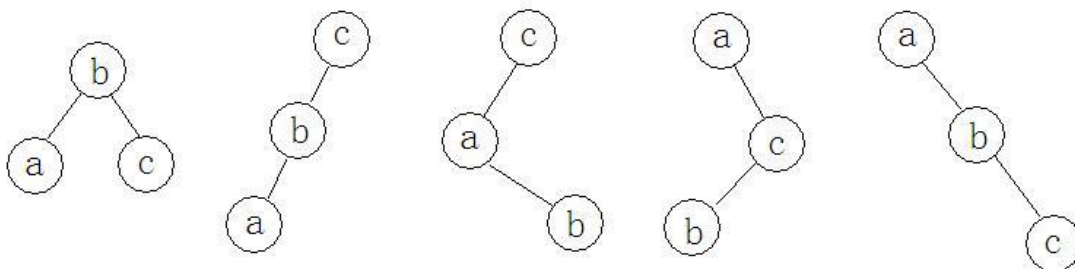
请画出对应的二叉树。解答: 以上两题的图分别如下:



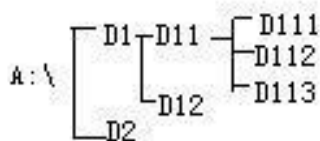
➤ 以上两题实质是考数据结构中的二叉树。还经常考二叉树的计数！如下题：

9. 如：(2000 年初中组) 已知按中序遍历二叉树的结果为:abc, 问: 有多少种不同形态的二叉树可以得到这一遍历结果, 并画出这些二叉树。

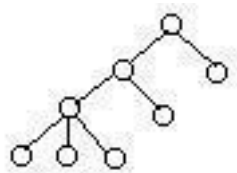
解答: 5 种, 形态如下:



10. (1999 年初中组) 在磁盘的目录结构中, 我们将与某个子目录有关联的目录数称为度。例如下图



该图表达了 A 盘的目录结构: D1, D11, ..., D2 均表示子目录的名字。在这里, 根目录的度为 2, D1 子目录的度为 3, D11 子目录的度为 4, D12, D2, D111, D112, D113 的度均为 1。不考虑子目录的名字, 则可简单的图示为如下所示的树结构:



若知道一个磁盘的目录结构中, 度为 2 的子目录有 2 个, 度为 3 的子目录有 1 个, 度为 4 的子目录有 3 个。

试问: 度为 1 的子目录有几个? 解答: 一种方法是画图; 另外, 可以根据整棵树的入度=出度(因为任一根关联边连接两个结点)这一性质

推导, 除根结点外的每个结点入度都是 1, 所以总的入度=1*x+1*2+1*1+1*3-1; 每个叶结点的出度为 0, 分支结点的出度为度数-1, 根结点的出度就是它的度, 所以总的出度=0*x+(2-1)*2+(3-1)*1+(4-1)*3-1; 算出:

x=9。

➤ 考查了计算机中的目录结构和树结构中的“度”的概念和性质。

11. (1998 年高中组) 用邻接矩阵/邻接表表示下面的无向/有向图(略)。

➤ 考查了数据结构中的图的表示。

12. (1999 年初中) 根据 Nocomachns 定理, 任何一个正整数 n 的立方一定可以表示成 n 个连续的奇数的和。

例如:

$$1^3=1$$

$$2^3=3+5$$

$$3^3=7+9+11$$

$$4^3=13+15+17+19$$

在这里, 若将每一个式中的最小奇数称为 X , 那么当给出 n 之后, 请写出 X 与 n 之间的关系表达式:

解答: $X=n*(n-1)+1$

➤ 考查代数和递推能力!

13. (2000 年高中组) 设有一个共有 n 级的楼梯, 某人每步可走 1 级, 也可走 2 级, 也可走 3 级, 用递推公式给出某人从底层开始走完全部楼梯的走法。例如: 当 $n=3$ 时, 共有 4 种走法, 即 $1+1+1, 1+2, 2+1, 3$ 。

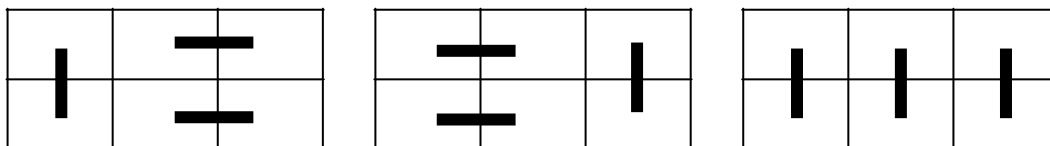
解答: 两种方法, 一是“猜”+“凑”, 从具体的 $n=1, 2, 3, \dots$ 算起, 只能算比较简单的, 容易错; 二是用组合数学和归纳法进行推导, 一般先假设 $F(n)=a*F(n-1)+b*F(n-2)+c*F(n-3)+\dots$, 然后算 a, b, c, \dots

直到某个系数=0 就结束, 再代入式子中。

$$\begin{aligned} F(1) &= 1 & F(2) &= 2 & F(3) &= 4 \\ F(N) &= F(N-3) + F(N-2) + F(N-1) & (N \geq 4) \end{aligned}$$

14. (2000 年初中组) 有 $2 \times n$ 的一个长方形方格, 用一个 1×2 的骨牌铺满方格。例如 $n=3$ 时, 为 2×3 方格。

此时用一个 1×2 的骨牌铺满方格, 共有 3 种铺法:



试对给出的任意一个 $n(n>0)$, 求出铺法总数的递推公式。

$$\begin{aligned} \text{解答: } F(1) &= 1 & F(2) &= 2 \\ F(n) &= F(n-2) + F(n-1) \quad (n \geq 3) \end{aligned}$$

➤ 以上两题, 考察了归纳+加法原理+乘法原理+递归关系式! 这是近年来的常见题。

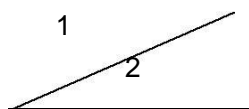
15. (2002 年初中组) 将 N 个红球和 M 个黄球排成一行。例如: $N=2, M=3$ 可得到以下 6 种排法: 红红黄黄黄 红黄红黄黄 红黄黄红黄 黄红红黄黄 黄红黄红黄 黄黄黄红红。

问题: 当 $N=4, M=3$ 时有多少种不同排法?(不用列出每种排法)

解答: 35

➤ 排列组合和概率统计!

16. (1999 年高中组) 将 L_n 定义为求在一个平面中用 n 条直线所能确定的最大区域数目。例如: 当 $n=1$ 时, $L_1=2$, 进一步考虑, 用 n 条折成角的直线(角度任意), 放在平面上, 能确定的最大区域数目 Z_n 是多少? 例如: 当 $n=1$ 时, $Z_1=2$ (如下图所示)



当给出 n 后, 请写出以下的表达式:

$$\begin{aligned} L_n &= \underline{\hspace{2cm}} \\ Z_n &= \underline{\hspace{2cm}} \end{aligned}$$

解答: 本题实质是求直线或折线将一个平面分成的最大区域数, 从两个方面考虑:

(1) 求在一个平面中用 n 条直线所能确定的最大区域数;

$$\begin{aligned} n=1, L_1 &= 2, & F(1) &= 2 \\ n=2, L_2 &= 4, & F(2) &= F(1) + 2 \\ n=3, L_3 &= 7, & F(3) &= F(2) + 3 \\ n=4, L_4 &= 11, & F(4) &= F(3) + 4 \end{aligned}$$

””

所以, $F(n) = F(n-1) + n$

把上面的 n 个等式左右相加, 化简得出: $F(n) = 2 + 2 + 3 + 4 + \dots + n$

即: $L(n) = n(n+1)/2 + 1$

(2) 求在一个平面中用 n 条折线所能确定的最大区域数;

$n=1, Z_1=2, \quad F(1)=0+2$

$n=2, Z_2=7, \quad F(2)=1*(2*2-1)+4$

$n=3, Z_3=16, \quad F(3)=2*(2*3-1)+6$

$n=4, Z_4=29, \quad F(4)=3*(2*4-1)+8$

””

所以, $F(n) = (n-1)*(2*n-1) + 2*n$

即: $Z(n) = (n-1)*(2*n-1) + 2*n$

➤ 几何+归纳+组合数学!

17. (1998 年初中组) 某班有 50 名学生, 每位学生发一张调查卡, 上写 a, b, c 三本书的书名, 将读过的书打 ✓, 结果统计数字如下: 只读 a 者 8 人; 只读 b 者 4 人; 只读 c 者 3 人; 全部读过的有 2 人; 读过 a, b 两本书的有 4 人; 读过 a, c 两本书的有 2 人; 读过 b, c 两本书的有 3 人; 问:

(1) 读过 a 的人数是 _____ (2) 一本书也没有读过的人数是 _____。

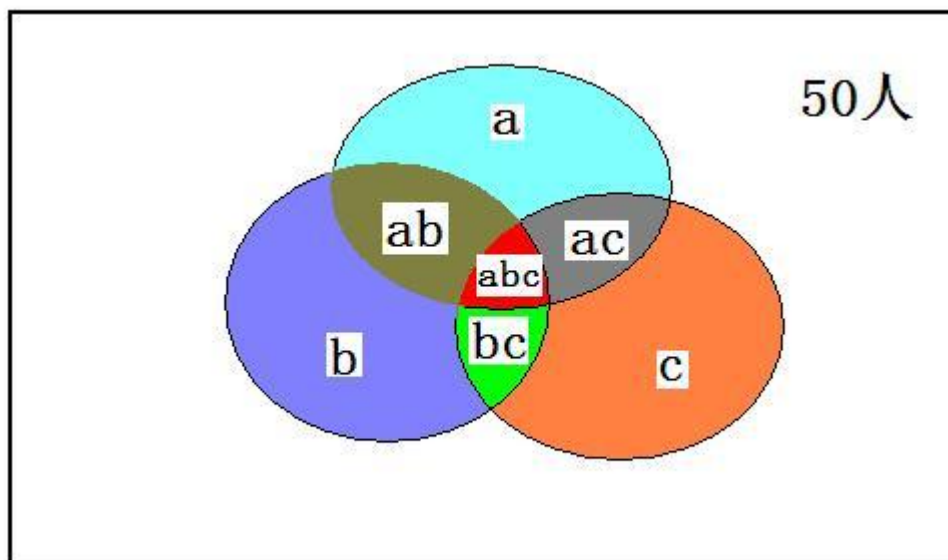
解答: (1) 12 人 (2) 30 人

方法: 推理或集合表示如下:

下图中: $a=8; b=4; c=3; abc=2; ab=4-abc=2; ac=2-abc=0; bc=3-abc=1;$

读过 a 的人数为: $a+ab+abc+ac=8+2+2+0=12$

一本未读过的人: $50-a-b-c-abc-ab-ac-bc=30$



➤ 逻辑推理+集合运算及变形!

第三部分: 阅读程序(4*8=32 分)

其实很容易, 目的几乎是送分, 而且占的分数很多, 但得分率却不见得高。很容易不明不白的就把分(全)丢了!!!

这部分程序考 3 个方面:

1. 程序设计语言本身, 如循环、递归、值型参和变量型参数、跟踪变量等;
2. 归纳和数学运算能力;
3. 是否掌握了一些常用算法(程序段)的框架;
4. 细心、耐心等心理品质; 灵感+编程的量等;

一般做这类题目的核心是找程序目的:

即这个程序想干什么。迄今为止考过的题目还没有“乱写”的, 总有一点“写作目的”的。抓住了它, 不仅得出答案变得很容易了, 而且对自己的结果也会比较有信心。

一般的解题步骤如下：

1. 从总体上通读程序，大致把握程序的目的和算法；
2. 猜测变量的作用，跟踪主要变量值的变化(列表)，找出规律；
3. 将程序分段，理清每一小段的作用和目的(灵感+关键表达式和语句的领会)；
4. 看清输入、按照输出格式，写出结果；
5. 带着得到的结果回到程序进行检查；

下面举几个例子。



1. 基本题(考语言本身，尤其是循环嵌套。1999 年初中组)

Program excpl;

var

x,y,y1,jk,j1,g,e:Integer;

a:array[1..20] of 0..9;

begin

x:=3465; y:=264; jk:=20; for

j1:=1 to 20 do a[j1]:=0; while

y<>0 do

begin

y1:=y mod 10;

y:=y div 10;

while y1<>0 do

begin

g:=x;

for e:=jk downto 1 do

begin

g:=g+a[e];

a[e]:=g mod 10;

g:=g div 10

end;

y1:=y1-1

end;

jk:=jk-1

end;

j1:=1;

while a[j1]=0 do j1:=j1+1;

for jk:=j1 to 20 do write(a[jk]:4); writeln

end.

程序运行结果为:_____。

解答：

程序不长，但是有一定的难度。高手最多半分钟就看懂了程序的意思，但初学者往往算了很久得出了结果却是错的。下面我们还是先以一个初学者的身份分析一下这个程序。记住，不要一开始就模拟电脑来一个个语句“执行”-----你算过自己是多少 Hz 的 CPU 没有？！

首先是看看变量的名字，可惜分区联赛题目中的变量不是 i 就是 j，很讨厌。i 和 j 一般作为循环计数器，没有什么意思，因此不要管它了。然后要看变量在程序的哪里出现过，着重看它与其它变量的相互引用关系，猜测它的作用。例如上题：x 只在 g:=x 中出现，暂时不要管它，因为它很可能只是一个初始数据。y 有三处：

1) while y<>0 do

2) $y1 := y \bmod 10$;

3) $y := y \div 10$;

很明显, y 每次少了最后一位数字,把这位数字给了 $y1$ 。有经验的选手应该体会到了什么, 不过我们继续。

现在我们知道了: y 对程序的作用是:每次提供最后一位给 $y1$,即 $y1$ 每次的值依次是:4,6,2

那么再看 $y1$, 它出现在两个地方:

1)while $y1 \neq 0$ do

2) $y1 := y1 - 1$

很明显就是一个循环次数嘛! 循环 $y1$ 次!

再看 jk :

1)for $e := jk$ downto 1 do

2) $jk := jk - 1$

jk 作为循环初始值, 居然一次比一次少...其原因有待进一步分析。

再看 $j1$:

1)for $j1 := 1$ to 20 do $a[j1] := 0$;

2) $j1 := 1$;

3)while $a[j1] = 0$ do $j1 := j1 + 1$;

4)for $jk := j1$ to 20 do write($a[jk]$):4);

显然, $j1$ 和其它变量没有什么联系。1)是初始化, 2)3)4)是输出数组 a 。

再看 g : 出现的位置是几层循环之内了, 应该很重要! 一会儿再分析!

再看 e : 作为循环变量, 没有什么意思。

通过变量分析, 我们知道了: x, y 是数据, y 每次提供最后一位给 $y1$, 循环 $y1$ 次。 $j1$ 和 g 的作用有待分析。

下面根据程序结构, 把程序分成几块, 逐个研究。最主要的程序段是两个 WHILE 循环中套一个 FOR 循环, 三重循环!!! 其实, 最外面一层很明确:判断什么时候结束($y=0$)。前后都很简单, 是一些变量和数组的初始化、输入、输出等, 下面重点剖析核心程序段。

1) $x := 3465$; $y := 264$; $jk := 20$;

for $j1 := 1$ to 20 do $a[j1] := 0$;

输入与变量、数组的初始化, 不要管它。

2)while $y \neq 0$ do

begin

$y1 := y \bmod 10$;

$y := y \div 10$;

while $y1 \neq 0$ do

begin

<< $g := x$;

for $e := jk$ downto 1 do

begin

$g := g + a[e]$;

$a[e] := g \bmod 10$;

$g := g \div 10$

end;

>>

$y1 := y1 - 1$

end;

$jk := jk - 1$

end;

<<>>中只有 6 行, 你可以模拟电脑“执行”几个语句在找规律。方法是:把循环“展开”, 再写一个变量值表即:

语句

执行后变量情况:

$g := x$;

$g := g + a[e]$;

$a[e] := g \bmod 10$;

$g := g \div 10$;

$g := g + a[e-1]$;

$a[e-1] := g \bmod 10$; $a[e-1] = a[e-1] + (x + a[e])$ 的进位

...

$g = x + a[e]$;

$a[e] = x + a[e]$ 的个位

$g = (x + a[e])$ 的前几位

$g = (x + a[e])$ 的前几位 + $a[e-1]$;

3)

```

j1:=1;
while a[j1]=0 do j1:=j1+1;
for Jk:=j1 to 20 do write(a[jk]:4); writeln

```

从前往后, 找到 $\neq 0$ 的位置开始, 输出数组元素的值(输出结果), 也不要管它。
块 2 最重要。

它的思想是: 每次取 Y 的最第位 y_1 , 执行 \ll 运算 $\gg y_1$ 次, 每次把 jk 减 1。

现在最重要的是 \ll 运算 \gg 中的在干什么?

注意到最后输出的 $a[]$, 要留意 $a[]$ 的变化!

$a[e]$ 总是取个位($g \bmod 10$), g 每次少一位, 和 $a[e-1]$ (别忘了 e 在循环!)相加...

难道是...高精度加法??? RIGHT!

它执行了 y_1 次, y_1 每次都是 y 的个位! 对了。程序就是在做 $x*y$

所以答案就是 $3465*264=914760$

再看它的输出格式, 输出的应该是: 9 1 4 7 6 0

其实有经验的话, 看到 g 这个变量名和 $g:=g+a[e]$; $a[e]:=g \bmod 10$; 这几个标志句子。就可以一下子知道程序的用意了。

总结一下本题: 重点考循环嵌套的执行过程以及对除法运算中的商、余数的基本方法; 主要程序段可以通过列表了解变量的变化情况; 要细心、耐心; 题目的本身没有多少值得研究的价值!!! 但有些题目**纯粹考算法思路**, 如下面的例子:



2. 算法题

```

program ex2;
var i,j,n:longint;
    b:array [0..31] of 0..1;
begin
    n=1999;
    i:=0;
    while n<>0 do
    begin
        b[i]:=n mod 2;
        i:=i+1;
        n:=n div 2
    end;
    for j:=i-1 downto 0 do write(b[j]);
end.

```

输出什么?

解答: 很明显, 是把十进制整数转换成二进制数, 所以输出 11111001111



3. 有些题目则是考数学, 或根据一些基本规则重复地做某个操作。如:

```

program exp1 (input,output);
var i,s,max: integer;
    a:array [1..10] of integer;
begin
    for i:=1 to 10 do read (a[i]);
    max:=a[1] ;s:=a[1];
    for i:=2 to 10 do
    begin

```

```

    if s<0 then s:=0; s:=
    s+a[i];
    if s>max then max:=s
end;
writeln('max=', max)
end.

```

输入: 8 9 -1 24 6 5 11 15 -28 9

输出: max=

解答: 本题主要做累加: $s := s + a[i]$; 再根据结果打擂台。

但最关键的语句是: `if s<0 then s:=0;` 它的作用是把 $s<0$ 的前若干个元素的和值屏蔽掉(置 0)。了解了这一点, 题目就很简单了。步骤如下:

	s<0?	s=8	s>max?	max=8;
l=2	N	8+9=17	Y	max=17;
l=3	N	17-1=16	N	max=17;
l=4	N	16+24=40	Y	max=40;
l=5	N	10+6=46	Y	max=46;
l=6	N	46+5=51	Y	max=51;
l=7	N	51+11=62	Y	max=62;
l=8	N	62+15=77	Y	max=77;
l=9	N	77-28=49	N	max=77;
l=10	N	49+9=58	N	max=77;

所以结果为: 77。

小结: 本质是求一个 n 长的整数数列的连续 x 长的子序列, 要求子序列的和最大!

注意: s 和 max !!! 另外本题给的输入数据比较简单, 所以有很多人没有完全懂也算对了结果, 把数据改成如下: -1 12 -103 65 34 -4 -27 8 -1234 9, 问结果是多少呢? 答: 9 !!!

4. 考子程序的调用, 尤其是递归或带参数(值参与变量型参数), 如:

```

PROGRAM EX3;
CONST N=10;
VAR S,I : INTEGER;
FUNCTION CO(I1:INTEGER) :
  INTEGER; VAR J1,S1 : INTEGER;
BEGIN
  S1:=N;
  FOR J1:= (N-1) DOWNT0 (N-I1+1) DO
    S1:= S1*J1 DIV (N-J1+1);
  CO:=S1
END;
BEGIN
  S:=N+1;
  FOR I:= 2 TO N DO S:=S + CO(I);
  Writeln('S=',S);
END.

```

解答过程:

- (1) 如果有子程序, 一般先读主程序, 了解主程序什么时候调用子程序? 干什么的? 调用了多少次? 本题调用了 $n-1$ 次, 并且累加函数的返回值!
- (2) 再单独阅读子程序, 分析变量、参数和返回值, 确定它的功能。本题函数猛一看好象比较复杂, 不过是通过一个循环结构完成累乘和累除, 下面再具体分析!

(3) 通过列表, 观察子程序中的变量变化情况, 以便找出规律, 确定子程序的作用。本题如下:

$$CO(2)=10*9/2$$

$$CO(3)=10*9*8/2/3$$

$$CO(4)=10*9*8*7/2/3/4$$

””

发现, 好像是组合数学的公式: $CO(i)=10!/(i!*(10-i)!)$

即: $C(m,n)=m!/(n!*(m-n)!)=m*(m-1)*...*(m-n+1)/n!$

(4) 所以结果很明确: $C(10,0)+C(10,1)+...+C(10,9)+C(10,10)=722$

总结: 灵感来源于丰富的数学基础和经验!

第四部分: 完善程序(28 分)

这部分题目得分率很低。没关系, 尽量做吧。实在不会把一些简单的填好就行了。有些题目即使不懂也能填出来:) 比如: $i:=i+1$; $i:=0$;

注意经常问自己程序中有没有做下面的事:

1) 初始化($i:=0$; $j:=0$; for $i:=1$ to n do $a[i]:=0$ 之类的)

2) 一些明显的动作:

a. 结果没有储存在需要的地方。

b. 累加器没有做加法

c. 输出

3) 关键动作。

例如交换排序程序的“交换”操作等很明显需要完成的操作。

分析方法和写运行结果类似, 注意分析变量和程序结构, 理解变量和模块的作用是解题的关键。

不熟练是不妨用自然语言描述一下。

一般的解题步骤:

1. 仔细阅读程序的目的是和算法、数据结构的描述!

千万不要一激动一紧张, 题目都没看透彻!!!

2. 把程序中的变量和题目中数据结构描述结合起来, 记住关键变量的作用!

有时就能填出一些简单的空!!! 不信?!!!

3. 结合问题的算法描述和要求及步骤, 把程序划分成几段, 每段完成指定的功能! 千万

不要忘记: 这是完善程序, 不是让你自己编!!! 一定要不断地结合题意和程序。

4. 逐步解决每段!

注意不要因为个别空而影响你对整个程序的把握, 不知道的, 先空在那儿, 把知道的填好, 最后再收拾那些难点!!!

注意一定要提醒自己: 程序中有没有做那些最基本的事。

5. 做完后, 不要忘了把程序从前往后读两遍, 看看是否完成了题目的任务; 还要检查一些细节性的问题, 如“>”还是“>=”, 是 $n-i$, 还是 $n-i+1$?

下面举例给予说明。



1. 基础题(算法、数据结构很清楚、很朴素, 送分!)

[程序的说明]

本程序对随机产生的 100 个 0 到 50 之间的随机数用一个数组存放后进行排序, 然后再将其中重复出现的数进行删除, 只保留一个, 使得剩下的数中任何两个都不相同且连续存储在原数组中。

```
const maxn=100;
```

```
type arraytype=array [1..maxn] of integer;
```

```
var i,j,temp,current,tail:integer;
```

```
    a:arraytype;
```

```
begin
```

```
    randomize;
```

```

for i:=1 to maxn do a[i]:=random(51);
for i:=1 to ____①____ do
  for j:= ____②____ to maxn do
    if a[i]<a[j] then
      begin temp:=a[i];a[i]:=a[j];a[j]:=temp end;
for i:=2 to maxn do
  if ____③____ then a[i]:=-a[i];
tail:=0;current:=1;
while ____④____ do
  begin
    while a[current]<0 do current:=current+1;
    tail:=tail+1;
    a[tail]:= ____⑤____ ;
    current:=current+1
  end;
if ____⑥____ then begin tail:=tail+1; a[tail]:=0 end;
for i:=1 to tail do write(a[i]:5); writeln

```

end.

[解答]

程序的思想已经不能再清楚了, 因为要排序(很明显是冒泡排序), 所以:

①maxn-1

②i+1

那么如何删除呢? 先分析一下变量的含义, **current** 和 **tail** 分别表示队列的头尾指针。再看删的过程: 好象是先做标记(置为负!), 然后从队首 **current** 开始找第 1 个没被做标记(>0)的数, 把它放到当前队尾 **tail** 的下一个位置。所以:

③a[i]=abs(a[i-1])

④(current<=maxn) and (a[current]<>0)

⑤a[current]

⑥(a[tail]<>0) and (a[current]=0)



2. 关键变量+特定的思想方法+灵感(1995 年初中组 2)

找出小于 33 的 6 个正整数, 用这些整数进行加法运算, 使得包括原来的整数在内能组成尽可能多的不同整数。

例如: 用 2, 3, 5 这三个数能可组成下面的数:

2, 3, 5, $2+3=5$ (但 5 已经存在)

$2+5=7$, $3+5=8$, $2+3+5=10$

所以用 2, 3, 5 能组成 6 个不同的数。程序要求: 输出所选的这 6 个数, 以及能组成不同整数的个数。

算法提要: 选择的这 6 个数, 用来组成数时应该尽可能不重复, 引入数组 A 保存找出的这 6 个整数。主要程序段:

```

A[1] := 1; t := 0;
For i := 2 to 6 do
  begin
    _____①_____;
    for j := 1 to i - 1 do s := _____②_____; a[i] :=
      _____③_____;
  end;
end;

```



```

For i:=1 to 6 do
  Begin
    t:= _____④_____ ;
    WRITE(a[i], ' ');
  End;
WriteLn('能组成不同整数的个数:', t)

```

解答:首先,根据蓝色的程序段,我们应该判断出③应该和 s 相关,而②是为了计算 s,所以本题的关键是变量 s 的含义。

不要着急,我们研究一下题目的例子和算法提要,发现:选择的 6 个数(a[i])应该尽可能不重复;且 $a[i] > a[i-1]$ 而 $a[i]$ 还要尽可能小;假设现在已求出了 $a[1], a[i-1]$, 那么为了满足“能组成尽可能多的不同整数”,则 $a[i]$ 应该取 $a[1]+a[2]+...+a[i-1]+1$,这样必然要设一个累加器,再看看程序:)还真是!!!所以得到:
①初始化 $s:=0$; ②累加 $s+a[i]$; ③赋值,注意多加 1: $s+1$;

那么④怎么填呢?它表示能组成的不同整数的个数,那为什么要扫描一遍数组呢?感觉也应该是累加!其实我们应该充分发挥上面已填好的程序段,发现:6 个数为:1 2 4 8 16 32(哦,难怪说小于 33!让我更加坚信上面做的是对的!),很明显是二进制数的问题吗?本质上就是一个求一个 6 位的二进制数最多能表示多少状态?答案为: $2^0+2^1+2^2+2^3+2^4+2^5=1+2+4+8+16+32$ 。不要激动,看题目④填什么?累加: $t+a[i]$ 。



3.复杂的问题描述+简单的程序+细心地处理细节问题(1995 年高中组 3)

设有一个实数,以字符串形式存放于数组 x 中,用 $x:\text{array}[1..N]\text{of char}$ 表示。其中 $x[1]$ 若为 '-', 表示负数;若为 '+', ' ' 或 ' ', 则表示正数。若为数字,也认为是正数。

例如 $x=(' ', '2', '0', ' ', ' ', '3', ' ', ' ', '5', '%')$ 则表示 203.5

$x=(' ', '1', ' ', ' ', ' ', '2', '0', '%')$ 则表示 -1.2

约定:在字符串 x 中,除 $x[1]$ 外,其后可以包含有若干个 '.' 与 ' ', 但仅以第一次出现的为准,空格不起任何作用,并以字符 '%' 作为结束标志。

程序要求:将输入的字符串还原成实数输出(小数点后无用的 0 应除去),还原的结果以下列形式存放(不需要输出)。

F:数符。正数放 0, 负数放 1。

A: $\text{array}[1..N]\text{of integer}$; 存放数字,不放小数点。

K:表示 A 中有效数字的个数。

J:表示小数点后的位数。

例如:数 203.24, 还原后结果的存放是:

F=0

A=(2, 0, 3, 2, 4)

K=5

J=2

又如:数 -33.0740, 还原后结果的存放是:

F=1

A=(3, 3, 0, 7, 4)

K=5

J=3

算法提要: $x:\text{array}[1..10]\text{of char}$;可放长度定为 10;首先读入字符串,然后处理数的符号,在还原的过程中,需要判定整数部分与小数部分,同时去除多余的空格和小数点,并约定输入是正确的,不用作出错检查。

只要程序段:

For I := 1 to 10 do a[I] := 0;

For I := 1 to 10 do read(x[I]);

J := 0; f := 0; k := 0; b := 0;

If x[1] = '-' then begin

_____①_____;

_____②_____;

End

Else if x[1] := ' ' then l := 2

Else l := 1;

While _____③_____ do l := l + 1;

While _____④_____ do

BEGIN

If (x[l]>= '0') and (x[l]<= '9') Then begin k:=k+1;

_____⑤_____;

If b=1 then _____⑥_____;
end

Else if (x[l]='.') and (b=0) then b := 1;

l:=l+1

END;

If j>0 then while a[k]=0 do begin

_____⑦_____

_____⑧_____

End;

解答:显然, 蓝色的程序段是用来处理实数的符号的, 所以根据约定①应该是设置负数标记, 即 f:=1;根据 else 后面的句子, 发现 i 为循环扫描的指针, 所以②应该是确定下一位置, 即 i:=2;

③很明显是过滤掉空格! 所以填: (x[i]=' ') and (i<10);

④也很明显, 一个大循环, 判断字符串是否扫描结束, 即: x[i]<>'%'

再看看 b 变量的含义: 根据 else 子句, 发现 b=1 表示整数部分结束! 所以⑤是把一个数字字符转换成数字填到数组 a 中 (a[k]:=ord(x[i])-48); ⑥填 j:=j+1; (j 表示小数点后的位数);

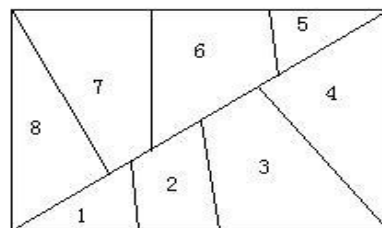
⑦⑧很明显, 是处理有小数、并且有多余的 0 的情况, 所以为: j:=j-1; k:=k-1;

小结: 注意程序前前后后看, 发现变量的作用和含义!!!

➤ 4. 典型算法+数据结构 (1996 年高中组 1/初中组 3)

四色问题: 设有下列形状的图形: (N=8),
其编号

为 1, 2, ..., N。



图形之间的相邻关系用下面的邻接矩阵表示:

其中: 1——相邻, 0——不相邻。

[程序要求] 将上面图形的每一个部分涂上红(1), 黄(2), 蓝(3), 绿(4)四种颜色之一, 要求相邻的部分有不同颜色。

输入方式: 邻接矩阵。

输出方式: 区域、颜色。

[算法描述] 用数组 R:ARRAY[1..N,1..N] OF 0..1 表示相邻关系,
S:ARRAY[1..N] OF INTEGER 表示颜色。

采用回溯的方法, 首先给第一个图形涂上红色(1), 然后在下面的图形中依次涂上其他颜色, 当有矛盾时回溯解决。

[程序]

	1	2	3	4	5	6	7	8
1	0	1	0	0	0	0	1	1
2	1	0	1	0	0	1	1	0
3	0	1	0	1	0	1	0	0
4	0	0	1	0	1	1	0	0
5	0	0	0	1	0	1	0	0
6	0	1	1	1	1	0	1	0
7			1	1	0	0	1	0
8	1	0	0	0	0	0	1	0

```

PROGRAM EXP2(INPUT,OUTPUT);
CONST N=8;
VAR I,J,K:INTEGER;
    R:ARRAY[1..N,1..N] OF 0..1;
    S:ARRAY[1..N] OF INTEGER;
BEGIN
    FOR I:=1 TO N DO
        BEGIN
            FOR J:=1 TO N DO READ(R[I,J]); READLN
        END;
        _____ ① _____ ; I:=2; J:=1;
    WHILE I<=N DO
        BEGIN
            WHILE (J<=4) AND (I<=N) DO
                BEGIN
                    K:=1;
                    WHILE _____ ② _____ DO K:=K+1;
                    IF K<I THEN _____ ③ _____
                        ELSE BEGIN
                            _____ ④ _____ ; I:=I+1; J:=1
                        END
                END;
            IF J>4 THEN BEGIN
                I:=I-1; _____ ⑤ _____
            END;
        END;
    FOR I:=1 TO N DO WRITELN(I,' → ',S[I])
END.

```

解答：邻接矩阵+回溯法，步骤略，答案如下：

- ① S[1]:=1;
- ② (K<I) AND (S[K]*R[I,J])<>J
- ③ J:=J+1;
- ④ S[I]:=J;
- ⑤ J:=S[I]+1;



5.子程序及参数(1999 年初中组)

[问题描述]

下面程序的功能是从键盘读取 A, B 数组的元素, A, B 数组均已从小到大排好序(无相同元素), 现将 A, B 合并为数组 C, 同样要求数组 C 也是从小到大排好序(有相同元素时只保留一个)。

程序中 N 表示数组 A, B 的长度, i, j, k 分别表示数组 A, B, C 的取数或存数的指针。

[程序清单]

```

program excp3;
const n=8;m=2*n;
type arr1=array[1..n]of integer;
      arr2=array[1..m]of integer;
var   a,b:arr1;c:arr2;i,j,k:integer;
procedure copy(x:arr1;var y:arr2;var i,j:integer);
begin

```

```

    i:=i+1;
    y[i]:=x[j];
    j:=j+1;
end;
begin
  for i:=1 to n do read(a[i]);readln; for i:=1
  to      n      do      read(b[i]);readln;
  i:=1;j:=1;_____①_____
  while_____②_____do
    if a[i]<b[j] then copy (a,c,k,i)
      else if b[j]<a[i] then copy (b,c,k,j) else
        begin
          copy(a,c,k,i);
          _____③_____
        end;
  while_____④_____do copy(a,c,k,i);
  while_____⑤_____do copy(b,c,k,j);
  for i:=1 to k do write (c[i]:4);
  writeln;
end.

```

解答:就本题而言,算法和题目本身对选手很清楚!线性表的归并操作在 **NOIP** 中考过多次,不管是用数组来操作还是用链表操作;甚至还有一些题目是它的变形(比如多项式的加法)。

本题中的 **copy** 过程是关键,好在题目并没有考到过程调用的语句(关键是参数的书写),所以下面只要理解了过程的作用和 4 个参数的含义,题目就会很容易了。

答案:① k:=0

② (i<=n)and (j<=n)

③ j:=j+1

④ i<=n

⑤ j<=n



6.特定的算法(1999 年高中组 2)

[问题描述] 用生成法求出 1, 2, ..., r 的全排列($r \leq 8$) [算法过程]

用数组: **a:array[1..r] of integer** ;表示排列;

初始化时, $a[i]:=1(i=1,2,\dots,r)$

设中间的某一个排列为 $a[1], a[2], \dots, a[r]$, 则求出下一个排列的算法为:

- (1) 从后面向前找,直到找到一个顺序为止(设下标为 j,则 $a[j-1] < a[j]$)
- (2) 从 $a[j] \sim a[r]$ 中,找出一个 $a[k]$ 比 $a[j-1]$ 大的最小元素
- (3) 将 $a[j-1]$ 与 $a[k]$ 交换
- (4) 将 $a[j], a[j+1], \dots, a[r]$ 由小到大排序。

[程序清单]

```

program exp4;
const r=7;
var   n,i,s,k,j,i1,t:integer;
      a:array[1..r] of integer;

procedure print1;
var   ik:integer;
begin
  for ik:=1 to r do write(a[ik]:8);writeln;

```

```

end
begin
for i:=1 to r do _____①_____ ;
print1;
s:=1;
for i:=2 to r do s:=s*i;
s:=s-1;
for i:=_____②_____ do
begin
j:=r;
while _____③_____ do j:=j-1;
k:=j;
for i1:=j+1 to r do
if _____④_____ then k:=i1;
t:=a[j-1];a[j-1]:=a[k];a[k]:=t;
for i1:=j to r-1 do
for k:=i1+1 to r do
if _____⑤_____ then
begin
t:=a[i1];a[i1]:=a[k];a[k]:=t;
end;
print1;
end;
end.

```

解题步骤:

- 1) 首先，本题给出了关键而详细的算法说明，但没有给一个样例，下面我们结合一个中间状态数据，看看如何生成下一个状态数据。

1 8 7 3 4 6 5 2

经过 4 步后得到: 1 8 7 3 5 2 4 6

这样，你对程序的逻辑过程就很清楚了！！！！

- 2) 把程序分段: 如上 4 中颜色。红色的过程表示输出，没问题！蓝色的显然是初始化，所以①填: $a[i] := i$;
绿色的表示统计总的方案数(并且已经输出了一个，所以-1);
紫色的程序段很明显是解决算法说明的 4 个步骤的，下面重点解决！

- 3) 看看 4 个步骤分别对应着哪些语句？注意对应和找关键动作！！！！

②应该填: 1 to s 或 s downto 1, 但不能写成 2 to s;

③填: $a[j-1] > a[j]$

④填: $(a[i1] > a[j-1])$ and $(a[i1] < a[k])$ 复合条件缺一不可，经常考！

⑤填: $a[i1] > a[k]$, 显然是从小到大冒泡排序用。 **小结:** 重视题目给出的每一个信息与程序中的哪些语句对应; 如果没有样例，自己找一个典型的，运行

运行找出规律; 程序的分块！！！！

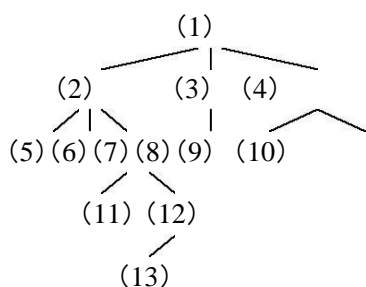


7. 数据结构题(1999 年高中组试题)

[问题描述] 求一棵树的深度与宽度。

[算法说明] 树可用数组 `tree:array[1..n,1..5] of integer;`

其中: $tree[l,1]$ 表示结点号; $tree[l,2]$ —— $tree[l,5]$ 所属结点



如上图可表示为:

1	2	3	4	0
2	5	6	7	0
3	8	0	0	0
4	9	10	0	0
5	0	0	0	0
6	0	0	0	0
7	11	12	0	0
8	0	0	0	0
9	0	0	0	0
10	0	0	0	0
11	0	0	0	0
12	13	0	0	0
13	0	0	0	0

在求解的过程中, 用到数组 $G: \text{ARRAY}[1..N, 1..7] \text{ OF INTEGER};$

其中: $G[l,1]$ 表示父结点, $G[l,2]$ 表示层次,

$G[l,3]$ 表示本结点号, $G[l,4]$ —— $G[l,7]$ 表示子女结点;

同时, 设 2 个指针 $SP1$ (取数指针), $SP2$ (存数指针)

[程序清单]:

```

program exGp3;
const n=13;
var i,j,k,sp1,sp2,n1,n2,jmax,p:integer;
    tree:array[1..n,1..5]of integer;
    g :array[1..n,1..7]of integer; begin

for i:=1 to n do
begin
    tree[i,1]:=i;
    for j:=2 to 5 do read (tree[i,j]);readln;
end;
sp1:=1;sp2:=1;g[1,1]:=0;g[1,2]:=1;g[1,3]:=1;
for i:=4 to 7 do g[1,i]:=tree[1,i-2];
while _____ ① _____ do
begin
    p:=g[sp1,2];n2:=g[sp1,3];_____ ② _____;j:=4;
    while _____ ③ _____ do
begin
        n1:=g[sp1,j];j:=j+1;_____ ④ _____;

```

```

        g[sp2,1]:=n2;g[sp2,2]:=p;g[sp2,3]:=n1; for i:=1
        to 4 do g[sp2,i+3]:=tree[n1,i+1];
    end;
    _____⑤_____
end;

writeln('maxd=',g[sp2,2]); j:=1;k:=g[1,2];jmax:=0; for i:=2
to sp2 do
if _____⑥_____ then j:=j+1
    else begin
        if j>jmax then jmax:=j;
        _____⑦_____;k:=g[i,2];
    end;

    if j>jmax then jmax:=j;
    writeln('max1=',jmax);
end.

```

解答：

- 1) 本题的数据结构含义，首先要搞清楚： $tree[i,j]$ 存储编号为 i 的结点的第 j 号孩子 ($2 \leq j \leq 5$, 即最多 4 个孩子), $tree[i,j]=0$ 表示不存在; $g[i,k]$ 是一个队列, $sp1$ 为读取队列用的指针, $sp2$ 为存储队列用的指针。 $g[i,1]$ 存储 i 的父结点, $g[i,2]$ 存储 i 所在的层次, $g[i,3]$ 存储本结点的编号 i , $g[i,4], g[i,5], g[i,6], g[i,7]$ 存储 i 的孩子结点编号。列出 g 的表格形式, 以便更加直观!!!
- 2) 程序关键在红色和蓝色的两段。先看红色段, ①显然表示队列非空时做,,, , 所以应该填: $sp1 \leq sp2$;
变量 p 是用来存放层次的, 所以②填: $p:=p+1$; 为该结点的孩子结点准备好层次; $n2$ 是表示当前处理的结点号, $n1$ 是表示 $n2$ 的孩子结点号(用 j 循环), ③这个地方的循环是为了遍历本结点的所有孩子, 所以③填: $g[sp1,j] < > 0$; 那么④⑤干什么呢? 不要忘记一个重要的事情, 队列的读取都需要移动指针($sp1, sp2$), 所以正好, ④为下面的存入操作作准备, 即 $sp2:=sp2+1$; ⑤为下一个结点的遍历操作作准备, 即读指针下移: $sp1:=sp1+1$;
- 3) 下面看看蓝色的程序段, 目的很明显是为了输出。再注意题目的目的: 输出树的宽度和深度! 深度简单, 其实就是 $g[sp2,2]$ 。题目也没有考你!!! 那么剩下的问题显然就是为了求宽度。方法也很简单, 就是求每一层的宽度(即 $g[i,4] \sim g[i,7]$ 中的非 0 个数, 或者按本题的方法是看 $g[i,2]$ 中最多有几个数相同), 然后打擂台找出最大值。因此, ⑥填: $g[i,2]=k$, 计算层次相同的元素个数放在 j 中, 用 j 与 $jmax$ 打擂台, 注意一层完了, j 值要还原, 宽度至少为 1, 所以⑦填: $j:=1$;

(2000 年高中组试题) 问题与上一样, 只是写程序的人不一样, 具体的风格不同而已。小结: 本题其实很重要的是队列的操作, 经常出现的还有: 堆栈操作、模拟法(解决递归等问题的现场保

护和恢复)。