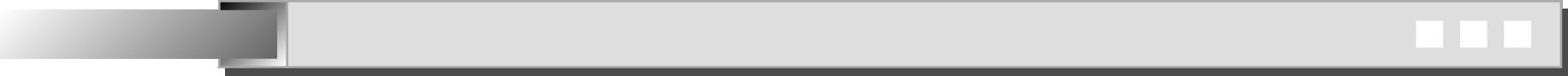
# 第1章

# 计算机系统基础知识

## 1.1 备考指南



### 1.1.1 考纲要求

根据考试大纲中相应的考核要求，在“计算机系统基础知识”模块上，要求考生掌握以下方面的内容。

(1) 计算机的类型和特点，包括微机(PC)、工作站、服务器、主机、大型计算机、巨型计算机、并行机。

(2) 中央处理器CPU，包括CPU的组成，常用的寄存器、指令系统、寻址方式、指令执行控制、中断控制、处理机性能。

(3) 主存和辅存，包括存储介质、高速缓存(Cache)、主存设备、辅存设备。

(4) I/O接口、I/O设备和通信设备，包括I/O接口、I/O设备(类型、特性)，通信设备(类型、特性)，I/O设备、通信设备的连接方法和连接介质类型。

(5) 数制及其转换，包括二进制、十进制和十六进制等常用数制及其相互转换。

(6) 数据的表示，包括数的表示(原码、反码、补码表示，整数和实数的机内表示方法，精度和溢出)、非数值数据的表示(字符和汉字的机内表示、声音和图像的机内表示)。

(7) 算术运算和逻辑运算，包括计算机中二进制数的运算方法、逻辑代数的基本运算。

### 1.1.2 考点统计



“计算机系统基础知识”模块，在历次程序员考试试卷中出现的考核知识点及分值分布情况如表1-1所示。

##### 表1-1 历年考点统计表

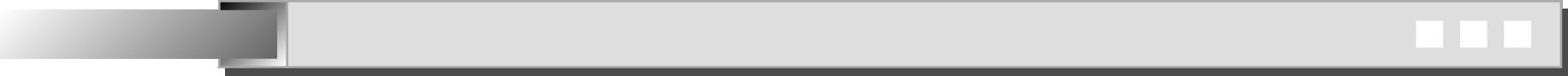
| 年 份 | 题 号 | 知 识 点 | 分 值 |
| --- | --- | --- | --- |
| 2009年  下半年 | 上午题：6～11、21~23 | 中央处理器CPU、主存和辅存、数制及其转换、算术运算和逻辑运算等 | 9分 |
| 下午题：无 |  | 0分 |
| 2009年  上半年 | 上午题：6～11、20～22、24 | 中央处理器CPU、主存和辅存、I/O接口、I/O设备和通信设备、数制及其转换、数据表示、算数运算和逻辑运算等 | 10分 |
| 下午题：无 |  | 0分 |
| 2010年  下半年 | 上午题：6～9、19～22 | 中央处理器、主存和辅存、数制及其转换、数据表示、算术运算和逻辑运算等 | 8分 |
| 下午题：无 |  | 0分 |
| 2010年  上半年 | 上午题：6～12、14、19～22 | 计算机的类型和特点、中央处理器CPU、主存和辅存、通信设备、数制及其转换、数据表示、算术运算和逻辑运算等 | 12分 |
| 下午题：无 |  | 0分 |

### 1.1.3 命题特点



纵观历年试卷，本章知识点是以选择题的形式出现在试卷中的。本章知识点在历次考试“综合知识”试卷中，所考查的题量大约为8～12道选择题，所占分值为8分～12分(约占试卷总分值75分中的10.67%～16.00%)。大多数试题偏重于理论知识，检验考生是否理解掌握相关的理论知识点，考试难度中等。

## 1.2 考点串讲



### 1.2.1 计算机系统的基本组成

#### 一、计算机系统的基本组成结构

计算机系统由硬件系统和软件系统两大部分组成，其基本组成结构如图1-1所示。

###### 1. 计算机硬件系统

计算机硬件是由运算器、控制器、存储器、输入设备和输出设备五大部分及总线组成，如图1-2所示。

运算器是计算机中进行数据加工的部件，主要完成算术运算和逻辑运算。

控制器是计算机中控制执行指令的部件，其主要功能如下。

* 正确执行每条指令。首先是获得一条指令，按硬件逻辑分析这条指令，再按指令格式和功能执行这条指令。
* 保证指令按规定序列自动连续地执行。
* 对各种异常情况和请求及时响应和处理。



##### 图1-1 计算机系统组成



##### 图1-2 计算机硬件结构

存储器是存放程序和工作数据的地方，分为内部存储器(或称主存储器)和外部存储器(或称辅助存储器)，分别简称为内存(或主存)和外存(或辅存)。内存速度快容量小，外存速度慢容量大。寄存器是CPU中的记忆设备，用来临时存放指令和数据，其速度比内存更快。

一般把CPU和主存储器的组合称为主机。输入/输出(I/O)设备位于主机之外，是计算机与外界交换信息的装置。

###### 2. 计算机软件系统

计算机软件是指为管理、运行、维护及应用计算机所开发的程序和相关文档的集合。其中，程序是让计算机硬件完成特定功能的指令序列，数据是程序处理的对象。计算机软件通常分为系统软件和应用软件。

1) 系统软件

系统软件是指那些为计算机所配置的、用于完成计算机硬件资源的控制与管理，以及为用户提供操作界面和为专业人员提供开发工具与环境的软件，如操作系统、程序设计语言及处理程序、数据库管理系统、实用程序与软件工具。

2) 应用软件

应用软件是指用于解决各种不同的具体应用问题的专门软件。应用软件可以分为通用软件和定制软件，如文字处理软件、电子表格软件、图形图像软件、网络通信软件、简报软件、统计软件等。

#### 二、计算机的类型和应用领域

按照计算机的工作能力，计算机可分为巨型机、大型机、小型机和微型机。微型机又可分为工作站、台式机、便携机、掌上型计算机、个人数字助理(PDA)等。按照功能是否专一，计算机可分为通用计算机和嵌入式计算机。

计算机的应用领域主要有数值计算、数据处理、实时控制(或过程控制)、人工智能、计算机辅助设计(CAD)、计算机辅助教学(CAI)、计算机辅助管理(CMA)等。

### 1.2.2 计算机中数据的表示及运算



#### 一、计算机中数据的表示

在计算机中，数字是以一串“0”或“1”的二进制代码来表示的，这是计算机唯一能识别的数据形式。数据必须转化成二进制代码来表示，也就是说，所有需要计算机加以处理的数字、字母、文字、图形、图像、声音等信息(人识数据)都必须采用二进制编码(机识数据)来表示和处理。

###### 1. 数制及其转换

按进位的方法进行计数，称为进位计数制。在采用进位计数制的数字系统中，如果只用*r*个基本符号来表示数值，则称其为*r*进制。每个数都可以用基数、系数和位数的形式来表示，即



* 基数(*K*)：是最大进位数(进制数)，数制的规则是逢*K*进1。例如，十进制基数为10，六十进制(时间)的基数为60等。
* 系数(*m*)：每个数位上的值，取值范围0～k-1。例如，234中百位系数为2，十位系数为3，个位系数为4。
* 位数(*n*)：各种进制数的个数。例如，十进制数234的位数为3，二进制数11010011的位数为8。

例如：(234)10=2×102+3×101+4×100(式中：*m*2=2，*m*1=3，*m*0=4；*K*=10；*n*=3)

显然，一个任意进制的数都可以按上述方法表示为其他进制的数。表1-2列出了计算机中常用的几种数制的对应关系。

##### 表1-2 计算机中常用数制的对应关系

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 十进制(D) | 二进制(B) | 八进制(O) | 十六进制(H) | 十进制(D) | 二进制(B) | 八进制(O) | 十六进制(H) |
| 0  1  2  3  4  5  6  7 | 0  1  10  11  100  101  110  111 | 0  1  2  3  4  5  6  7 | 0  1  2  3  4  5  6  7 | 8  9  10  11  12  13  14  15 | 1000  1001  1010  1011  1100  1101  1110  1111 | 10  11  12  13  14  15  16  17 | 8  9  A  B  C  D  E  F |

数制转换主要有如下几种。

1) *r*进制转换成十进制

方法：

*an*…*a*1*a*0.*a*-1…*a*-*m* (*r*) = *a*\**rn* +…+ *a*\**r*1 + *a*\**r*0 +*a*\**r*-1+…+*a*\**r*-*m*

例如：

10101(B)=1×24+1×22+1×20=21

101.11(B)= 1×22+1×20+1×2-1+1×2-2=5.75

101(O)=1×82+1×80=65

71(O)=7×81+1×80=57

101A(H)=1×163+1×161+10×160=4122

2) 十进制转换成*r*进制

方法：

* 整数部分：除以*r*取余数，直到商为0，余数从右到左排列。
* 小数部分：乘以*r*取整数，整数从左到右排列。

例如：

① 100.345(D)=1100100.01011(B) ② 100(D)=144(O)=64(H)

2 100 0 0.345 8 100 4

2 50 0 × 2 8 12 4

2 25 1 0.690 8 1 1

2 12 0 × 2 0

2 6 0 1.380

2 3 1 × 2 16 100 4

2 1 1 0.760 16 6 6

0 × 2 0

1.520

× 2

1.04

3) 八进制和十六进制转换成二进制

方法：

* 每一个八进制数对应二进制的三位。
* 每一个十六进制数对应二进制的四位。

例如：

7123(O)=111 001 010 011(B) 144(O)=001 100 100(B)

7 1 2 3 1 4 4

2C1D(H)=0010 1100 0001 1101(B) 64(H)=0110 0100(B)

2 C 1 D 6 4

4) 二进制转换成八进制和十六进制

方法：

* 整数部分：从右向左进行分组。
* 小数部分：从左向右进行分组。
* 转换成八进制每三位一组，不足补零。
* 转换成十六进制每四位一组，不足补零。

例如：

1 101 101 110.110 101(B)=1556.65(O)

1 5 5 6 6 5

11 0110 1110.1101 01(B)=36E.D4(H)

3 6 E D 4

###### 2. 二进制运算规则

* 加法：1+0=1；0+1=1；0+0=0；1+1=0(有进位)。
* 减法：1-0=1；1-1=0；0-0=0；0-1=1(有借位)。
* 乘法：0×0=0；1×0=0；0×1=0；1×1=1。
* 除法：是乘法的逆运算。

###### 3. 机器数和码制

各种数据在计算机中的表示形式称为机器数，其特点是采用二进制计数制，数的符号用0、1来表示，小数点则隐含表示而不占位置。真值是机器数所代表的实际数值。

机器数分无符号数和带符号数两种。无符号数表示正数，没有符号位。对于无符号数，若约定小数点的位置在机器数的最低位之后，则是纯整数；若约定小数点的位置在最高位之前，则是纯小数。对于带符号数，最高位是符号位，其余位表示数值，若约定小数点的位置在机器数的最低位之后，则是纯整数；若约定小数点的位置在最高数值位之前(符号位之后)，则是纯小数。

为方便运算，带符号的机器数可采用原码、反码和补码等不同的编码方法，这些编码方法称为码制。真值的符号数字化：我们用“+”和“-”来表示正负数，而计算机则在二进制数的最高位设置成符号位，通常用“0”表示正数，“1”表示负数。

1) 原码

规则：最高位为符号位，“0”表示正数，“1”表示负数。对数0则有“+0”和“-0”两种表示。

* 当0≤*X*时，[*X*]原=0*X* 例如，+7：00000111 +0：00000000
* 当0≥*X*时，[*X*]原=1|*X* | 例如，-7：10000111 -0：10000000

对*n*+1位字长用以表示整型数值的范围：-2*n*+1≤*X*≤2*n*-1。

2) 反码

规则：最高位为符号位，“0”表示正数，“1”表示负数。正数与原码相同，负数则要将除符号位的其他位按位取反。对数0则有“+0”和“-0”两种表示。

* 当0≤*X* 时，[*X*]反=0*X* 例如，+7：00000111 +0：00000000
* 当0≥*X* 时，[*X*]反=1|| 例如，-7：11111000 -0：11111111

对*n*+1位字长用以表示整型数值的范围：-2*n*+1≤*X*≤2*n*-1。

3) 补码

规则：最高位为符号位，“0”表示正数，“1”表示负数。正数与原码相同，负数则要将除符号位的其他位按位取反后加1。对数0只有“0”一种表示。

* 当0≤*X* 时，[*X*]补=0*X* 例如：+7：00000111 +0：00000000
* 当0＞*X* 时，[*X*]补=1||+1 例如：-7：11111001 -128：10000000

对*n*+1位字长用以表示整型数值的范围：-2*n*≤X≤2*n*-1。

补码运算的优点：将减法运算变成加法运算(因为运算器中只有加法器)。

例如：96-20=76

0110 0000 ←[+96]补

+ 1110 1100 ←[-20]补

1 0100 1100 →76 最高位的进位 1则自然丢失

4) 移码

规则：最高位为符号位，“1”表示正数，“0”表示负数。

当-2*n* ≤ *x* < 2*n* 时，[*x*]移 = 2*n*+*x*。

数值范围：-2*n* ≤ *x* ≤2*n* – 1。

特点：保持了数据原有的大小顺序，便于进行比较操作。

以上介绍的4种编码方法(设字长为4位，最高位为符号位)的对应关系如表1-3所示。

##### 表1-3 符号数的四种编码表示

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| *x*0*x*1*x*2*x*3 | 原 码 | 反 码 | 补 码 | 移 码 | *x*0*x*1*x*2*x*3 | 原 码 | 反 码 | 补 码 | 移 码 |
| 0000 | 0 | 0 | 0 | -8 | 1000 | -0 | -7 | -8 | 0 |
| 0001 | +1 | +1 | +1 | -7 | 1001 | -1 | -6 | -7 | 1 |
| 0010 | +2 | +2 | +2 | -6 | 1010 | -2 | -5 | -6 | 2 |
| 0011 | +3 | +3 | +3 | -5 | 1011 | -3 | -4 | -5 | 3 |
| 0100 | +4 | +4 | +4 | -4 | 1100 | -4 | -3 | -4 | 4 |
| 0101 | +5 | +5 | +5 | -3 | 1101 | -5 | -2 | -3 | 5 |
| 0110 | +6 | +6 | +6 | -2 | 1110 | -6 | -1 | -2 | 6 |
| 0111 | +7 | +7 | +7 | -1 | 1111 | -7 | -0 | -1 | 7 |

###### 4. 定点数和浮点数

1) 定点数

(1) 定点小数的表示。

小数点设在符号位(S)之后，其表示格式如下所示。

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| S |  |  |  |  |  |  |  |

设字长为*n*+1位，定点小数的数值表示范围如下。

* 原码表示：-(1-2-*n*)～+(1-2-*n*)。
* 反码表示：-(1-2-*n*)～+(1-2-*n*)。
* 补码表示：-1～+(1-2-*n*)。

例如，(-0.25)10→(-0.01)2，以原码定义表示为10100000。

(2) 定点整数的表示。

定点整数分为(有)符号数和无符号数两种表示格式。

① (有)符号数：小数点在符号位最末有效位之后。其表示格式如下。

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| S |  |  |  |  |  |  |  |

设字长为*n*+1位，符号数的数值表示范围如下。

* 原码表示：-(2-*n*-1)～+(2-*n*-1)。
* 反码表示：-(2-*n*-1)～+(2-*n*-1)。
* 补码表示：-2-*n*～+(2-*n*-1)。

例如，(-10)10→(-1010)2，以原码定义表示为10001010。

② 无符号数：不设符号位，小数点在符号位最末有效位之后。其表示格式如下。

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |

设字长为*n*+1位，无符号数的数值表示范围为0≤*N*≤2*n*+1-1。

例如，(255)10→(11111111)2，以原码定义表示为11111111。

2) 浮点数

构成：阶码*E*，尾数*M*，符号位*S*，基数*R*。

*N* = (-1)*S*×*M*×*RE*

|  |  |  |
| --- | --- | --- |
| *S* | *E* | *M* |

① 规格化：为了在尾数中表示最多的有效数据位，也为了数据表示的唯一性而定义的规则。如将尾数的绝对值限制在区间[0.5, 1]中，当尾数(*M*)用补码表示时，有以下两种情况。

* *M*≥0时，尾数规格化的形式：*M*=0.1*X*…*X*。
* *M*<0时，尾数规格化的形式：*M*=1.0*X*…*X*。

② 浮点数的表示范围：尾数的位数决定数的精度，阶码的位数决定数的范围。而表示范围与机器的具体的表示方法及字长有关，下面举例说明。

例：以*R*为基数，有*p*位阶码和*m*位二进制尾数代码的浮点数，阶码采用二进制正整数编码表示，求数值的表示范围。

解：最小规格化尾数：1/*R*

最大规格化尾数：1-2-*m*

最大阶码：2*p*-1

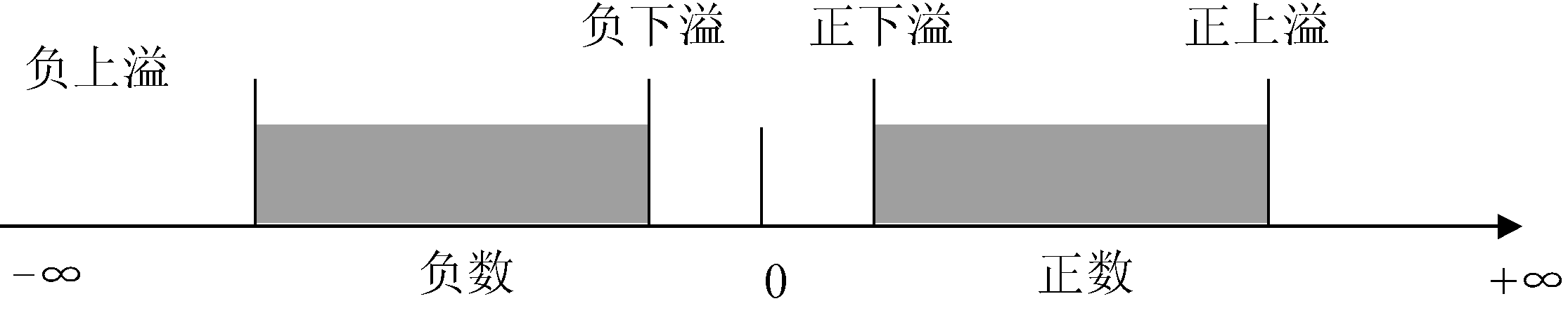
最小阶码：0

最小值：1/*R*

最大值：

注：本例中没有符号位，也没有考虑阶码为负的情况。如果考虑这些因素就要考虑阶码和尾数的编码方式。

③ 浮点数的溢出：当运算的结果超出该机器浮点数可表示的范围时，则产生浮点数溢出，浮点数可表示的范围如图1-3所示。比如上例中，当浮点数的运算结果小于1/*R*(或大于)时，则产生正下溢(或正上溢)。



##### 图1-3 浮点数的表示范围

④ 浮点数的实例。设浮点数格式如下：

|  |  |  |  |
| --- | --- | --- | --- |
| 阶符 | 阶码 | 数符 | 尾数 |

则数110.011(B)=+0.110011×2+11(规格化尾数)=0 110011×20 11(机器数格式)可表示为：

|  |  |  |  |
| --- | --- | --- | --- |
| 0 | 11 | 0 | 110011 |

3) 浮点数工业标准IEEE 754

规格化数格式如下：

(-1)*s*×1.*f*×2*E*

其中，1位数符(*s*)：正数为0，负数为1；除去了最高位的尾数( *f*)为原码表示；阶码(*E*)为特殊移码表示。

IEEE 754浮点数的范围如表1-4所示。

##### 表1-4 IEEE 754浮点数的表示范围

|  |  |  |
| --- | --- | --- |
| 格 式 | 最 小 值 | 最 大 值 |
| 单精度 | *E*=-126， *f* =0， 1.0×2-126 | *E*=127，*f*=11…1， 1.11…1×2127 = 2127×(2-2-23) |
| 双精度 | *E*=-1022， *f* =0， 1.0×2-1022 | *E*=1023，*f*=11…1， 1.11…1×21023 =21023×(2-2-52) |

例：将IEEE 754标准的精度浮点数0 10000110 01100000001000000000000转换为真值。

解：将特殊移码表示阶码转换为真值阶码，因为*E*=10000110-01111111=00000111，所以*E*=7；因为*f* =01100000001000000000000，所以1.*f*=1.01100000001；将1.*f*右移7位(因为*E*=7)=(10110000.0001)2=176.0625。

###### 5. 十进制数与字符的编码表示

数值、文字和英文字母等字符在进入计算机时，都必须转换成二进制表示形式，称为字符编码。

用4位二进制代码表示1位十进制数，称为十进制编码，简称BCD编码。常用的十进制数的编码有8421 BCD码、余3码、格雷码。

上述三种编码与十进制数的对应关系如表1-5所示。

##### 表1-5 常用编码与十进制数的对应关系

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 十进制数 | BCD码 | 格雷码 | 余3码 | 十进制数 | BCD码 | 格雷码 | 余3码 |
| 0 | 0000 | 0000 | 0011 | 5 | 0101 | 1110 | 1000 |
| 1 | 0001 | 0001 | 0100 | 6 | 0110 | 1010 | 1001 |
| 2 | 0010 | 0011 | 0101 | 7 | 0111 | 1000 | 1010 |
| 3 | 0011 | 0010 | 0110 | 8 | 1000 | 1100 | 1011 |
| 4 | 0100 | 0110 | 0111 | 9 | 1001 | 0100 | 1100 |

###### 6.  ASCII码

ASCII码(American Standard Code for Information Interchange)是美国标准信息交换码的简称，该编码已成为国际通用的信息交换标准代码。

ASCII码采用7个二进制位对字符进行编码，其格式为每1个字符有1个编码。每个字符占用1个字节，用低7位编码，最高位为0。其共有128个编码，编码从0～127，如表1-6所示，其中H表示高3位，L表示低4位。

##### 表1-6 ASCII码表

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| H  L | 000 | 001 | 010 | 011 | 100 | 101 | 110 | 111 |
| 0000 | NUL | DLE | SP | 0 | @ | P | ` | p |
| 0001 | SOH | DC1 | ! | 1 | A | Q | a | q |
| 0010 | STX | DC2 | " | 2 | B | R | b | r |
| 0011 | ETX | DC3 | # | 3 | C | S | c | s |
| 0100 | EOT | DC4 | $ | 4 | D | T | d | t |
| 0101 | ENG | NAK | % | 5 | E | U | e | u |
| 0110 | ACK | SYN | & | 6 | F | V | f | v |
| 0111 | BEL | ETB | ' | 7 | G | W | g | w |
| 1000 | BS | CAN | ( | 8 | H | X | h | x |
| 1001 | HT | EM | ) | 9 | I | Y | i | y |
| 1010 | LF | SUB | \* | : | J | Z | j | z |
| 1011 | VT | ESC | + | ; | K | [ | k | { |
| 1100 | FF | FS | ， | < | L | \ | l | | |
| 1101 | CR | GS | - | = | M | ] | m | } |
| 1110 | SO | RS | . | > | N | ↑ | n | ～ |
| 1111 | SI | US | / | ? | O | ← | o | DEL |

###### 7. 汉字编码

汉字处理包括汉字的编码输入、汉字的存储和汉字的输出等环节。在汉字处理的各阶段，分为输入码、(机)内码、交换码(国标码)和字形码，各种码对应的处理过程如下所示：

键盘管理程序 汉字处理程序 显示、打印

外部(输入)码 → 机内码 → 字形(输出)码

(键盘) (计算机存储、传输) (计算机输出汉字)

↑ ↓↑ ↓

汉字信息   交换码(国标码) 汉字信息

   ↓↑

   其他系统代码

1) 输入码

数字编码：用数字串代表一个汉字的输入。国标区位码等便是这种编码法。

拼音编码：是以汉语拼音为基础的输入方法。由于汉字同音字太多，因此输入重码率很高，因此，按拼音输入后还必须进行同音字选择，影响了输入速度。全拼、双拼、微软拼音等便是这种编码法。

字形编码：是以汉字的形状确定的编码。汉字总数虽多，但都是由一笔一画组成，全部汉字的部件和笔画是有限的。因此，把汉字的笔画部件用字母或数字进行编码，按笔画书写的顺序依次输入，就能表示一个汉字，五笔字型、表形码等便是这种编码法，这种方法的缺点是需要记忆很多的编码。

2) 内部码

汉字内部码(简称内码)是汉字在信息处理系统内部存储、处理、传输汉字用的代码。国家标准局GB 2312—1980规定的汉字国标码中，每个汉字内码占两个字节，每个字节最高位置“1”，作为汉字机内码的标识。以汉字“大”为例，国标码为3473H，两个字节的最高位为“1”，得到的机内码为B4F3H。又例如：

汉字 国标码 汉字机内码

沪 2706(00011011 00000110B) 10011011 10000110B

久 3035(00011110 00100011B) 10011110 10100011B

3) 字形码

汉字字形码是表示汉字字形的字模数据，通常用点阵、矢量函数等方式来表示。字形码也称为字模码，它是汉字的输出形式，随着汉字字形点阵和格式的不同，汉字字形码也不同。常用的字形点阵有16×16点阵、24×24点阵、48×48点阵等。

字模点阵的信息量是很大的，占用的存储空间也很大，以16×16点阵为例，每个汉字占用32(16×16/8=32)个字节，两级汉字大约占用256KB。因此，字模点阵只能用来构成“字库”，而不能用于机内存储。字库中存储了每个汉字的点阵代码，当显示输出时才检索字库，输出字模点阵得到字形。

汉字的矢量表示法是将汉字看作是由笔画组成的图形，提取每个笔画的坐标值，这些坐标值就可以确定每个笔画的位置，所有坐标值组合起来就是该汉字字形的矢量信息。每个汉字矢量信息所占的内存大小不一样。

#### 二、校验码

计算机系统运行时，在各个部件之间经常需要进行数据交换，为保证数据传送过程的正确无误，必须引入差错检查机制对数据进行校验，以检测是否有数据传送错误。其基本原理是：在编码中引入一定的冗余位，当被传送的编码中出现错误时就使之成为非法代码而被检测出。

###### 1. 奇偶校验码

奇偶校验码用于并行码的检错。其原理是：在*k*位数据码之外增加1位校验位，使*k*+1位码字中取值为1的位数总保持为偶数(偶校验)或奇数(奇校验)。

(1) 水平校验：设最高位为校验位。

原有数字位 生产新的码字： 偶校验 奇校验

0 0 0 1 1 0 0 0 1 0 0 0 0 1

0 1 0 1 0 0 1 0 1 1 0 1 0 1

(2) 垂直校验：设4个字节的数据为一组进行垂直奇校验。

第一字节数据： 10110011

第二字节数据： 11011001

第三字节数据： 01010101

第四字节数据： 10001100

垂直校验位： 01001100

(3) 垂直水平校验：设4个字节的数据为一组进行垂直水平奇校验。

水平校验位

第一字节数据： 10110011 0

第二字节数据： 11011001 0

第三字节数据： 01010101 1

第四字节数据： 10001100 0

垂直校验位： 01001100

###### 2. 海明码

海明码用于多位并行数据检错纠错处理。

实现：为*k*个数据位设*r*个校验位，使*k*+*r*位的码字(即海明码)能发现*k*位中任何一位出错且可以纠正。

其数据位*k*和校验位*r*必须满足如下关系式：

2*r*≥*k*+*r*+1

1) 海明码的编码规则

设*r*个校验位*PrPr*-1…*P*1，*k*个数据位*Dk*-1*Dk*-2…*D*0，产生的海明码为*Hr*+*kHr*+*k*-1…*H*1，则有如下规则。

* 规则1：*Pi*在海明码的2*i*-1位置，即*Hj*=*Pi*，*j*=2*i*-1；数据位则依序从低至高占据海明码中的其他位置。
* 规则2：海明码中的任意位都是由若干校验位来校验的。其对应关系是：被校验的海明位的下标等于所有参与校验该位的校验位的下标之和，而校验位则由其自身来校验的。

2) 海明码校验

下面以*k*=8对纠1位错的海明码的编码及校验原理给予说明。

(1) 确定*r*的位数。

当*r*=4时，有24≥8+4+1，可以满足2*r*≥*k*+*r*+1。

(2) 确定海明码的位置。

由规则1，*Pi*i对应*Hj*的位置j。

例如，确定*P*4的位置，因*j*=2*i*-1=24-1=8，则有*P*4在*H*8的位置。同理得到以下位置的对应关系：

*H*12 *H*11 *H*10 *H*9 *H*8 *H*7 *H*6 *H*5 *H*4 *H*3 *H*2 *H*1

*D*7 *D*6  *D*5 *D*4 *P*4 *D*3 *D*2 *D*1 *P*3 *D*0 *P*2 *P*1

(3) 确定编码方案。

由规则2，编码方案如下。

形成*Hi*与*P*1相关的数据位有：*S*1= *D*0⊕*D*1⊕*D*3⊕*D*4 ⊕*D*6

形成*Hi*与*P*2相关的数据位有：*S*2= *D*0⊕*D*2⊕*D*3⊕*D*5 ⊕*D*6

形成*Hi*与*P*3相关的数据位有：*S*3= *D*1⊕*D*2⊕*D*3⊕*D*7

形成*Hi*与*P*4相关的数据位有：*D*7⊕*D*6⊕*D*5⊕*D*4

得校验关系：*S*4= *D*4⊕*D*5⊕*D*6⊕*D*7

(4) 确定纠错译码方案。

设*G*1=*S*1⊕*P*1；*G*2=*S*2⊕*P*2；*G*3=*S*3⊕*P*3；*G*4=*S*4⊕*P*4。若采用偶校验则*G*4～*G*1的值全为0时数据正确，反之有错。用对出错位取反方法即可实现纠错。

例如，*G*4～*G*1=0111，其值为7，对应于*H*7(*D*3的位置)。将*D*3的值取反就实现了纠错。

###### 3. 循环冗余校验码(CRC)

利用生成多项式为*k*个数据位产生*r*个校验位来进行编码，其编码长度为*k*+*r*。CRC的代码格式为：

*n* *n*-1 *r*+1 *r* *r*-1   2 1

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  | … |  |  |  | … |  |  |

数据位 校验位

由此可知循环冗余校验码由两部分组成，左边为信息码，右边为校验码，若信息码占*k*位，则校验码就占*n*-*k*位，所以又称为(*n*, *k*)码。

#### 三、逻辑代数及逻辑运算

在逻辑代数中，逻辑值只有两个，即“T”与“F”(或“Y”与“N”)。我们知道，二进制数也只有两个值“1”与“0”，所以可用二进制数表示逻辑值，并充分利用逻辑运算的特点，进行信息的处理。

注意：运算按位进行，没有进位和借位。

###### 1. 基本的逻辑运算

1) “与”运算(逻辑乘运算)

与逻辑：决定事件发生的各条件中，所有条件都具备，事件才会发生(成立)。

两个逻辑变量的与逻辑关系表达式：*Y*= *A*·*B*(或*Y*=*A*∧*B*，也可简写为*Y*=*AB*)。

两个逻辑变量的与逻辑真值表：

*A* *B* *Y*

0 0   0

0 1   0

1 0   0

1 1   1

与运算规则：

0·0=0 0·1=0 1·0=0 1·1=1

*A*·0 =0 *A*·1 =*A* *A*·*A* =*A* ·*A*=0

例如： 1100 1000

· 1011 0101

1000 0000

与门逻辑符号如表1-7所示。

2) “或”运算(逻辑加运算)

或逻辑：决定事件发生的各条件中，有一个或一个以上的条件具备，事件就会发生(成立)。

两个逻辑变量的或逻辑关系表达式：*Y*=*A*+*B*(或*Y*=*A*∨*B*)。

两个逻辑变量的或逻辑的真值表：

*A* *B* *Y*

0 0 0

0 1 1

1 0 1

1 1 1

或运算规则：

0+0=0，0+1=1，1+0=1，1+1=1

*A*+0=*A*，*A*+1=1，*A*+*A*=*A*，*A*+=1

例如： 1100 1000

+ 1011 0101

1111 1101

或门逻辑符号如表1-7所示。

3) “非”运算(取反运算)

非逻辑：决定事件发生的条件只有一个，条件不具备时事件发生(成立)，条件具备时事件不发生。

非逻辑关系表达式：*Y*=。

非逻辑的真值表：

*A* *Y*

0 1

1 0

非运算规则：

=1 =0 =*A*

例如：=0011 0111

非门逻辑符号如表1-7所示。

##### 表1-7 各种门电路符号及逻辑表达式

|  |  |  |
| --- | --- | --- |
| 门 电 路 | 符 号 | 表 达 式 |
| 与门 |  |  |
| 或门 |  |  |
| 非门 |  |  |

###### 2. 常用的逻辑公式

* 交换律：*A*+*B* = *B*+*A*
* 结合律：*A*+*B*+*C*=(*A*+*B*)+*C*=*A*+(*B*+*C*) *ABC*=(*AB*)*C*=*A*(*BC*) *AB*=*BA*
* 分配律：*A*(*B*+*C*)=*AB*+*AC* *A*+*BC*=(*A*+*B*)(*A*+*C*)
* 原变量吸收律：*A*+*AB* =*A*
* 反变量吸收律：*A*+*B*=*A* +*B* +*AB*=+*B*
* 反演律(德摩根定理)：=+；=•
* 互补律：+*A*=1；·*A*=0

###### 3. 逻辑表达式及其化简

1) 逻辑表达式与真值表

逻辑表达式是用逻辑运算符把逻辑变量(或逻辑常量)连接在一起表示某种逻辑关系的表达式。把变量和表达式的各种取值都一一对应列举出来称为真值表。

例：证明+*B*·*C*+*A*·=+*C*。

解：对*A*、*B*、*C*的所有逻辑取值，如表1-8所示，两个逻辑表达式的函数值相等，证毕。

##### 表1-8 真值表求证逻辑表达式

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| *A* | *B* | *C* | •+*B*•*C*+*A*• | +*C* | *A* | *B* | *C* | •+*B*•*C*+*A*• | +*C* |
| 0 | 0 | 0 | •+0•0+0•=1 | +0=1 | 1 | 0 | 0 | •+0•0+1•=1 | +0=1 |
| 0 | 0 | 1 | •+0•1+0•1=1 | +1=1 | 1 | 0 | 1 | •+0•1+1•=1 | +1=1 |
| 0 | 1 | 0 | •+1•0+0•=0 | +0=0 | 1 | 1 | 0 | •+1•0+1•=0 | +0=0 |
| 0 | 1 | 1 | •+1•1+0•=1 | +1=1 | 1 | 1 | 1 | •+1•1+1•=1 | +1=1 |

2) 逻辑表达式的化简

利用逻辑运算规律可以对逻辑表达式进行化简。

例：化简+*BC*+*A*。

解：+*BC*+*A*

=((+*A*)+*BC*) (结合律、分配律)

=(+*BC*) (互补律)

=+*C* (吸收律)

#### 四、机器数的运算

###### 1. 机器数的加减运算

在计算机中，通常只设置加法器，减法运算要转换为加法运算来实现。机器数的加、减法运算一般用补码来实现，其运算方法如下：

*X*±*Y*→[*X*]补+[±*Y*]补

例如(采用8位定点整数)：

8-5→(+1000)2+(-101)2→(+00001000)+(-00000101)→(00001000)补+(11111011)补=100000011

运算结果中的后8位的真值为+3，是正确的。

当运算的结果超过了字长的表示范围时，则产生溢出。双符号位方法是常用的溢出判别方法。在CPU中的加法器前设1位寄存器*S*0，运算时接收来自最高位(符号位*S*)的进位。运算前*S*0、*S*被设为一操作数的符号，运算后对其进行判别，则有以下的逻辑关系：当运算后*S*0⊕*S*=1，则溢出；当运算后*S*0⊕*S*=0，则无溢出。

例如，8位定点整数的最大正数是127→(01111111)。若再加1则为10000000，按机器的表示格式，这个值被认为是-128，显然是不正确的，也就是说产生了溢出问题。下面用上述的双符号位方法完成此题的计算和判别。

127+1→(1111111)2+(1)2→转为机器数→(01111111)+(000000001)

运算过程：

0 1 1 1 1 1 1 1

+ 0 0 0 0 0 0 0 0 1

0 1 0 0 0 0 0 0 0

*S*0 *S*

因为*S*0⊕*S*=0⊕1=1，表示运算结果溢出。

###### 2. 机器数的乘除运算

在计算机中实现乘除运算，主要有以下3种方法。

* 纯软件方案，乘除运算通过程序来完成。该方法速度很慢。
* 通过增加少量用来实现左右移位的逻辑电路来实现。
* 通过专用的硬件阵列乘法器(或除法器)来实现。

###### 3. 浮点运算

1) 浮点加减运算

完成浮点数加减法有5个基本步骤：对阶、尾数加减、规格化、舍入和检查溢出。

例：两浮点数*x* = 201×0.1101，*y* = 211×(-0.1010)。假设尾数在计算机中以补码表示，可存储4位尾数，2位符号位，阶码以原码表示，求*x*+*y*。

解：将*x*、*y*转换成浮点数据格式

[*x*]浮 = 00 01，00.1101

[*y*]浮 = 00 11，11.0110

具体的步骤如下。

① 对阶，阶差为11-01=10，即2，因此将*x*的尾数右移两位，得：

[*x*]浮 = 00 11，00.001101

② 对尾数求和，得：

[*x*+*y*]浮 = 00 11，11.100101

③ 由于符号位和第一位数相等，不是规格化数，向左规格化，得：

[*x*+*y*]浮 = 00 10，11.001010

④ 舍入，得：

[*x*+*y*]浮 = 00 10，11.0010

⑤ 数据无溢出，因此结果为：

*x*+*y* = 210×(-0.1110)

2) 浮点乘除运算

浮点数相乘，其积的阶码等于两乘数的阶码之和，尾数等于两乘数的尾数之积，数符由两乘数的数符按逻辑异或求出。

浮点数相除，其商的阶码等于被除数的阶码减去除数的阶码，尾数等于被除数的尾数除以除数的尾数、数符由两除数的数符按逻辑异或求出。

### 1.2.3 计算机的基本组成及工作原理



#### 一、总线的基本概念

###### 1. 总线的定义与分类

总线是连接多个设备的信息传送通道，是一组信号线。一般可分为以下几类。

* 芯片内总线：集成电路芯片内部各部分的连接。
* 元件级总线：一块电路板内各元器件的连接。
* 内总线(又称系统总线)：计算机各组成部分(CPU、内存和外设接口)间的连接。
* 外总线(又称通信总线)：计算机对外的接口，可直接与相应的外设连接或与其他计算机相连接。

###### 2. 内总线

内总线有专用内总线和标准内总线之分。常见的内总线标准有以下几种。

* ISA总线(Industry Standard Architecture)(PC/AT)：工业标准体系结构，ISA总线是具有开放式结构的总线。ISA总线为62+36线，数据线16，地址线24。
* EISA总线(Enhanced Industry Standard Architecture)：EISA总线是ISA总线的扩展，现用在服务器上。EISA总线的数据线32位，与ISA总线兼容。
* PCI总线(Peripheral Computer Interconnect)：PCI总线是目前微型机上广泛采用的内总线。PCI总线有两种标准：适用于32位机的124个信号的标准和适用于64位机的188个信号的标准。PCI总线的传输率至少为133Mb/s，64位的传输率为266Mb/s。PCI总线的工作与处理机的工作是并行的。PCI总线上的设备是可即插即用的。

###### 3. 外总线

通用串行总线USB(Universal Serial Bus)：USB接口提供电源。最大数据传输率为12Mb/s。USB设备可以通过集线器Hub进行树状连接，最多可达五层，连接显示器和键盘等外设(最多127个)。USB 2.0的传送速率为480Mb/s，支持可即插即用功能。

SCSI总线(Small Computer System Interface)：SCSI总线是从通道发展而来的，其特点是设备独立性强；传输速度快，16位的Ultra2 SCSI数据传输率为80Mb/s。目前传输率高达320Mb/s；灵活性好(适用于各种外设)，最多可接63种外设，传送距离可达20m。

IEEE 1394(Firewire)：IEEE 1394总线可连接的设备数多，最多可接63部；传输速度快，可达400Mb/s；安装步骤简单，可以“点对点”或以“集线器”等方式串接；支持即插即用。

串行总线接口(RS-232)：RS-232是国际通用的一种串行通信接口标准。串行通信物理连接方式如下。

* 单工：单向传输。
* 全双工：可同时双向传输。
* 半双工：可双向传输，但同一时刻只能单向传输。

#### 二、中央处理单元(CPU)

###### 1. CPU的功能

CPU的基本功能如下。

* 程序控制：CPU通过执行指令来控制程序的执行顺序。
* 操作控制：一条指令功能的实现需要若干操作信号来完成，CPU产生每条指令的操作信号并将其送往不同的部件，控制相应部件的操作。
* 时序控制：CPU通过时序电路产生的时钟信号进行定时，可以控制各种操作按指定时序进行。
* 数据处理：完成对数据的加工处理。

###### 2. CPU的组成

CPU包括运算器、控制器、寄存器三大部分，一般被集成在一个大规模集成芯片上，是计算机的核心部件，具有计算、控制、数据传送、指令译码及执行等重要功能，它直接决定了计算机的主要性能。其主要功能部件包括以下各部分。

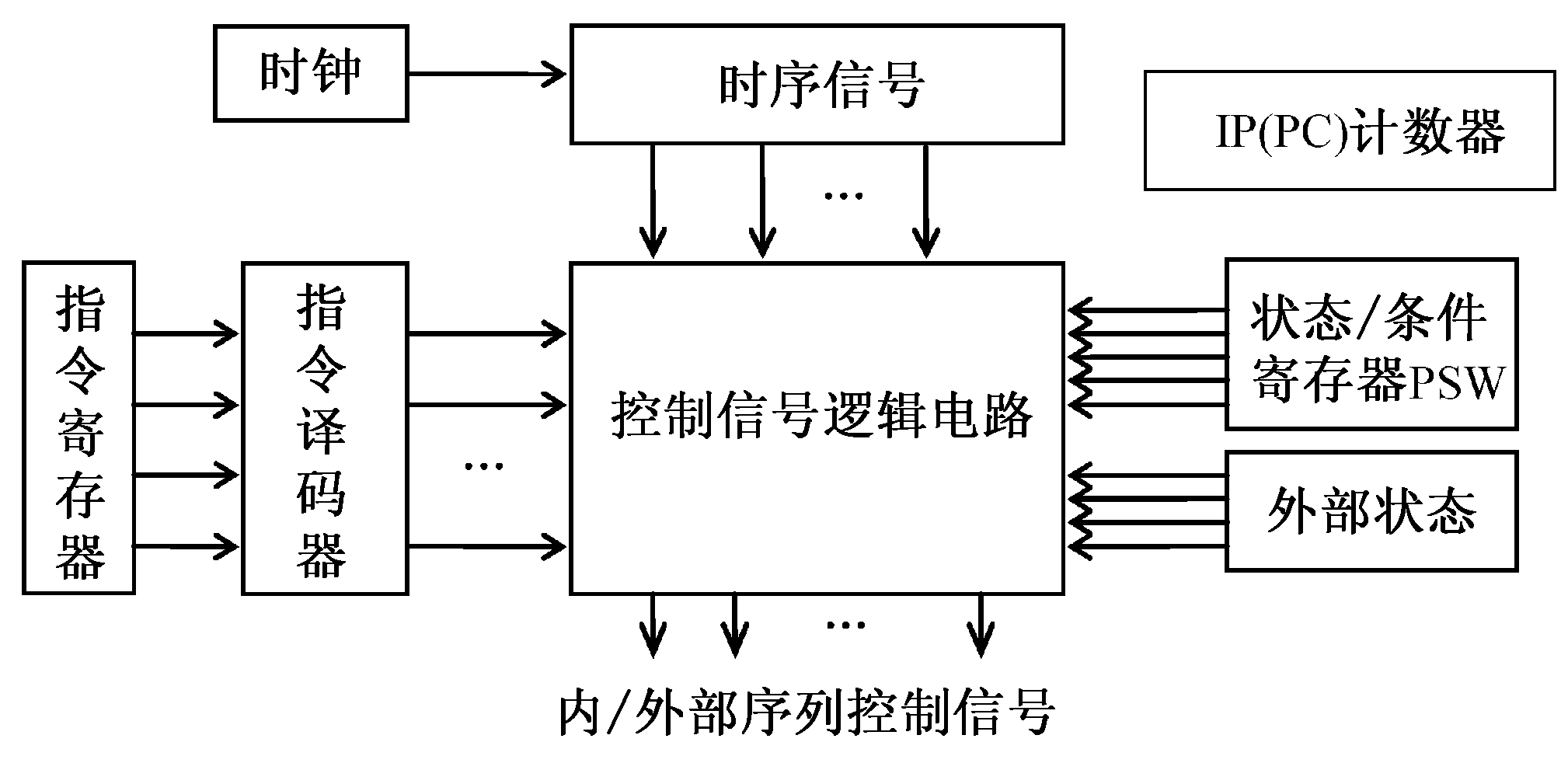
1) 运算器

运算器主要完成算术运算、逻辑运算和移位操作，主要部件有算术逻辑单元ALU、累加器ACC、标志寄存器、寄存器组、多路转换器和数据总线等。

2) 控制器

控制器实现指令的读入、寄存、译码和在执行过程有序地发出控制信号。其主要部件如图1-4所示。

* 程序计数器(PC)：当程序顺序执行时，每取出一条指令，PC内容自动增加一个值，指向下一条要取的指令。
* 指令寄存器：用于寄存当前正在执行的指令。
* 指令译码器：用于对当前指令进行译码。
* 状态/条件寄存器：用于保存指令执行完成后产生的条件码，另外还保存中断和系统工作状态等信息。
* 时序部件：用于产生节拍电位和时序脉冲。



##### 图1-4 控制器结构框图

3) 寄存器

寄存器用于暂存寻址和计算过程的信息。CPU中的寄存器通常分为存放数据的寄存器、存放地址的寄存器、存放控制信息的寄存器、存放状态信息的寄存器和其他寄存器等类型。

#### 三、存储系统

###### 1. 存储器的分类

存储器的分类主要有以下几种。

* 按存储器所处的位置分为内存和外存。
* 按构成存储器的材料分为磁存储器、半导体存储器和光存储器。
* 按工作方式分为只读存储器和读写存储器。
* 按访问方式分为按地址访问的存储器和按内容访问的存储器。
* 按寻址方式分为随机存储器、顺序存储器和直接存储器。

###### 2. 主存储器

主存储器简称为主存或内存。

1) 主存的种类

主存一般有RAM和ROM两种工作方式的存储器，其绝大部分存储空间由RAM构成。

2) 主存的组成

主存储器一般由地址寄存器、数据寄存器、存储体、控制线路和地址译码电路等部分组成，如图1-5所示。



##### 图1-5 主存储器结构框图

* 地址寄存器(MAR)：用来存放要访问的存储单元的地址码，其位数决定了其可寻址的存储单元的个数*M*，即*M*=2*N*。
* 数据寄存器(MDR)：用来存放要写入存储体中的数据或从存储体中读取的数据。
* 存储体：存放程序和数据的存储空间。
* 控制线路：根据读写命令控制主存储器各部分的相应操作。
* 译码电路：根据地址译码器中的地址码在存储体中找到相应的存储单元。

3) 主存储器的性能指标

* 容量：存储器芯片的容量是以存储1位二进制数(bit，比特)为单位的，因此存储器的容量即指每个存储器芯片所能存储的二进制数的位数。可由以下公式求得：  
   存储器芯片容量=单元数×数据线位数
* 存取周期：存储器芯片的存取周期是用存取时间来衡量的。它是指从CPU给出有效的存储器地址到存储器给出有效数据所需要的时间。存取时间越少，则速度越快。一般在20ns～300ns之间，记作*T*m。
* 存储器带宽：每秒钟能访问的比特数，记作*B*m。设每个存取周期存取数据位为*W*b，则*B*m=*W*b/*T*m。

###### 3. 高速缓存

高速缓冲存储器(Cache)所用芯片都是高速的，其存取速度可与微处理器相匹配，容量由几万字节到几十万字节，通常用来存储当前使用最多的程序或数据。Cache位于CPU与主存储器之间，是对程序员透明的一种高速小容量存储器。所谓透明是指程序员不必自己去加以操作和控制，而由硬件自动完成。每次访问存储器时，都先访问高速缓存，若访问的内容在高速缓存中，访问到此为止；否则，再访问主存储器，并把有关内容及相关数据块取入高速缓存。这样，如果大部分针对高速缓存的访问都能成功，那么在主存储器容量保持不变的情况下，访存速度可接近高速缓存的存取速度，这无疑可提高微机的运行速度。

Cache主要由两部分组成：控制部分和存储器部分。

###### 4. 外存储器

外存储器具有存储容量大、价格便宜、信息不易丢失、存取速度比内存慢、机械结构复杂、只能与主存储器交换信息而不能被CPU直接访问等特点，属于输入/输出设备。外存储器的这些特点正好与主存储器互为补充，共同支撑着整个计算机存储体系实现有效的功能。

1) 磁盘存储器

磁盘存储器是外存中最常用的存储介质，存取速度较快且具有较大的存储容量，分为软盘和硬盘存储器。

磁盘存储器的主要技术指标如下。

(1) 道密度：沿盘面半径方向单位长度内磁道的数目，单位是道/毫米。

(2) 位密度：磁道圆周上单位长度内存储的二进制位的个数，单位是位/毫米。

(3) 存储容量：存储容量=总的盘面数×每面的磁道数×每道的扇区数×每个扇区存储的字节数。

(4) 平均访问时间：平均访问时间=平均寻道(址)时间+平均等待时间。

* 平均寻道(址)时间表示从当前道移至目标道的平均时间，反映了磁头的移动定位速度。
* 平均等待时间：磁头到目标道后，等待到达目标扇区的平均时间。

(5) 数据传输率：单位时间内写入或读出的字节数。数据传输率=每道扇区数×每个扇区包括的字节数×磁盘的转数。

(6) 软盘：软盘存储器是由软盘片、软盘驱动器和软盘驱动器适配器3个部分组成的，其中，软盘片是存储介质，软盘驱动器是读写装置，适配器是通过总线与CPU相连的接口。

(7) 硬盘：微机的硬盘存储器是一个密封式的不可拆卸的、被固定安装在主机箱内的设备。大型计算机的硬磁盘(磁盘阵列)通常有单独的机柜。硬磁盘容量大(存储单位为GB)，信息传送速度(平均访问时间在几毫秒到十几毫秒)远远高于软盘，是计算机系统中最重要的一种外围存储设备。

2) 光盘存储器

光盘存储器的特点是存储量大、价位低、可靠性高、寿命长，特别适用于图像处理、大型数据库系统、多媒体教学等领域。光盘有音频光盘、视频光盘和计算机用数字光盘。光盘按其功能的不同，可分为CD-ROM(只读型光盘)、WORM(可写一次性光盘)和可重写型光盘。

3) USB移动硬盘和USB闪存盘

USB移动硬盘容量大，支持热插拔，即插即用。USB闪存盘又称为优盘，是使用闪存作为存储介质的一种半导体存储设备，采用USB接口标准。根据不同的使用要求，优盘还有基本型、加密型、启动型等类型。

#### 四、输入/输出技术

###### 1. 接口的功能及分类

1) 接口

接口又称为界面，是指两个相对独立的子系统之间的相连部分。用于连接主机和I/O设备的转换机构就是I/O接口电路。I/O接口的主要功能如下。

* 地址译码功能。
* 在主机和I/O设备间交换数据、控制命令及状态信息等。
* 支持主机采用程序查询、中断、DMA等访问方式。
* 提供主机和I/O设备所需的缓冲、暂存、驱动功能。
* 进行数据的类型、格式等方面的转换。

2) 接口的分类

接口主要有以下三种分类方式。

* 按数据的传送格式分为并行接口和串行接口。
* 按主机访问I/O设备的控制方式，可分为程序查询接口、中断接口、DMA接口以及通道控制器、I/O处理机等。
* 按时序控制方式可分为同步接口和异步接口。

###### 2. 主机和外设间的连接方式

主机和外设间的连接方式常见的有总线型、星型、通道方式和I/O处理机等，其中总线方式是基本方式。

总线是一组能为多个部件分时共享的信息传送线，用来连接多个部件并为之提供信息交换通路。共享是指连接到总线上的所有部件都可以通过它传递信息，分时性是指某一时刻只允许一个部件将数据发送到总线上。因此共享是利用分时实现的。

要实现分时共享，必须制定相应的规则，称为总线协议。总线协议一般包括：信号线定义、数据格式、时序关系、信号电平、控制逻辑等。

###### 3. I/O接口的编址方式

1) 与内存单元统一编址

将I/O接口中有关的寄存器或存储器部件看作存储器单元，与主存中的存储单元统一编址。这种编址方法的优点是原则上用于内存的指令全都可以用于接口。其缺点是地址空间被分成两部分，会导致内存地址不连续。

2) I/O接口单独编址

通过设置单独的I/O地址空间，为接口中的有关寄存器或存储部件分配地址码，需要设置专门的I/O指令进行访问。这种编址的优点是不占主存的地址空间，缺点是用于接口的指令太少。

###### 4. CPU与外设之间交换数据的方式

1) 直接程序控制

* 程序查询方式：在这种方式下，CPU通过执行程序查询外设的状态，判断外设是否准备好进行数据传送。程序查询方式的传输过程，如图1-6所示。



##### 图1-6 程序查询方式的传输过程

* 立即程序传送方式：在这种方式下，I/O接口总是准备好接收来自主机的数据，或随时准备向主机输入数据，CPU无须查看接口的状态，就执行输入/输出指令进行数据传送。这种方式又称为无条件传送或同步传送。

2) 中断方式

中断是在发生了一个外部事件时调用相应的处理程序(或称服务程序)的过程。中断服务程序与中断时CPU正在运行的程序是相互独立的，相互不传递数据。

(1) 中断处理中要解决的问题。

* 中断处理程序入口地址的形成(称为中断响应过程)：由硬件中断机构根据中断源引出中断向量表，其步骤是：关中断(屏蔽中断)→保存现场→识别中断→形成服务程序入口地址→执行服务程序→恢复现场→开中断。
* 中断屏蔽：由硬件中断屏蔽寄存器实现多重中断(中断嵌套)，即中断服务程序也可以被中断。多重中断的过程如图1-7所示。

(2) 实现中断屏蔽的两种方法。

* 在CPU内设置一个中断屏蔽寄存器，通过指令设置该寄存器关(或开)，用以屏蔽(或不屏蔽)对外部所有的中断请求，常在保存(或恢复)现场时使用。
* 采用中断屏蔽寄存器，每位对应一个中断源，用软件灵活地设置屏蔽寄存器的内容就可改变优先级，其原理如图1-8所示。

##### 图1-7 中断嵌套示意图 图1-8 中断屏蔽的原理

(3) 中断的三种类型。

* 内部中断(异常处理)：算术操作异常、非法指令、越权指令。
* 外部中断(可屏蔽，不可屏蔽)：键盘、鼠标、电源。
* 软件中断：系统功能调用。

3) DMA方式

目的：减少大批量数据传输时CPU的开销。

方法：采用专用部件生成访存地址并控制访存过程。

传输过程：预处理，数据传输，后处理。

DMA方式用于高速外围设备与内存之间批量数据的传输，其使用一个专门的DMA控制器来完成内存与设备之间的直接数据传送，而不用CPU干预。当本次DMA传送的数据全部完成时，才产生中断，请求CPU进行结束处理。DMA控制器与其他部件的关系如图1-9所示，其控制流程如图1-10所示。

##### 图1-9 DMA控制器与其他部件的关系 图1-10 DMA控制器的控制流程

因为CPU与DMA都要访问内存，所以会出现访存冲突，一般有以下解决方法。

* CPU等待DMA。
* DMA在存储器空闲时访问存储器(周期挪用)。
* CPU与DMA交替访问存储器(透明方式)。

4) 通道方式

通道是一个用来控制外围设备工作的专用处理机。它对外围设备实现统一管理，代替CPU对I/O操作进行控制，从而使I/O操作可以与CPU并行工作。通道是实现计算和传输并行的基础，通道的应用可以提高整个系统的效率。

通道的类型包括：选择通道、数组多路通道、字节多路通道。

通道的功能包括：接受CPU的指令；读取并执行通道程序；控制数据传送；读取外设的状态信息，提供给CPU；发出中断请求。

### 1.2.4 指令系统简介



指令是指让计算机完成某个操作而发出的命令。一条指令对应着一种基本操作，计算机所能执行的全部指令，就是计算机的指令系统。指令系统的设计原则为：完备性、正交性、规整性、可扩充性、有效性及兼容性。

#### 一、指令格式

###### 1. 指令的基本格式

指令的基本格式：操作码 | 操作数项

* 操作码：表示指令的操作性质，如加、减。
* 操作数项：是操作过程中涉及的数据来源，一般由目标数据项和源数据项两项组成，表示两个参与运算的数据。每项都可用各种寻址方式得到。根据指令中操作数项的数量，指令格式中的操作数项可分为下述几种：零操作数项、一操作数项、二操作数项、三操作数项。

###### 2. 指令的执行方式

顺序执行：PC寻址，即按指令的长度递增。

非顺序执行无条件转移指令：指令的操作数给出下条指令的地址。

非顺序执行条件转移指令：以条件状态(如8086中N, Z, V, C, P)和指令的操作数来确定下条指令的地址。

#### 二、寻址方式

###### 1. 寻址方式

隐含方式：隐含地指出目标操作数，如PUSH AX，栈指针寄存器(SP)指示的地址为目标操作数的地址。

立即数方式：源操作数在指令中给出，如图1-11所示。例如，ADD AX, 12。

寄存器方式：操作数项给出寄存器编号，如图1-12所示。例如，ADD AX, BX。

直接寻址方式：操作数项给出内存地址编号，如图1-13所示。例如，ADD AX, [200]。

间接寻址方式：寄存器间接寻址方式，寄存器中的内容为操作数的地址，例如，INC [BX]。存储器间接寻址方式，指令操作数项指示的地址中的内容是操作数的地址，如图1-14所示。

|  |  |
| --- | --- |
| 图1-11 立即数寻址方式 | 图1-12 寄存器寻址方式 |
| 图1-13 直接内存寻址方式 | 图1-14 存储器间接寻址方式 |

* 寄存器相对寻址方式：指令地址码部分给出的是一个偏移量，操作数地址等于本条指令的地址加上该偏移量。如图1-15所示。例如，MOV AX，8[R]。
* 变址和基址寻址方式：操作数地址等于变址寄存器的内容加偏移量。例如，MOV AX，[BX][DI]。
* 相对变址和基址寻址方式：操作数地址等于一个基址寄存器的值和一个变址寄存器的值以及本条指令中的8位/16位偏移量之和。例如：MOV AX，8[BX][DI]。



##### 图1-15 寄存器相对寻址方式

###### 2. 数据类型

计算机指令处理的基本数据类型有以下4种。

* 地址型：无符号的整型。
* 数值型：整型(字节、字、双字)、浮点数(单精度、双精度)。
* 字符型：以字节为单位的ASCII码字符串。
* 逻辑型：按二进制位进行逻辑运算的数据。

#### 三、指令的种类

###### 1. 数据传送类指令

数据传送类指令主要有数据传送指令和数据交换指令等。其功能是将原始数据、中间结果、最终结果，以及其他各种数据在CPU的寄存器和存储器之间传送。

###### 2. 算术运算类指令

CPU能够对字节、字或双字进行算术运算，包括加法、减法、乘法、除法、求补、加1、减1、比较等。

###### 3. 逻辑运算类指令

逻辑运算类指令对操作数中的各个位分别进行相应的逻辑运算，如与、或、异或、取反指令。

###### 4. 程序控制类指令

程序控制类指令用于改变指令执行的顺序。其主要有跳转指令、子程序调用和返回指令、陷阱指令等。

###### 5. 输入输出类指令

输入指令IN用于从外设端口接收数据，输出指令OUT用于向端口发送数据。

###### 6. 移位操作指令

移位操作可以对操作数向左或向右移动若干位。一般分为以下3种类型。

* 算术移位：该指令对带符号操作数进行移位，左移时从最低位依次向最高位移动，最低位补0，最高位进入“进位”中；右移时从最高位向最低位依次移动，最低位进入“进位”，而最高位(即符号位)保持不变。
* 逻辑移位：该指令对无符号操作数进行移位，左移时和算术左移相同，右移时和算术右移不同，是用0补充最高位。
* 循环移位：分为带进位和不带进位的循环移位两种，带进位的移位时要把进位带入到移位运算中，不带进位的则不用。

###### 7. 字符串操作类指令

字符串操作类指令的操作对象不只是单个的字节或字，而是内存中地址连续的字节串或字串。其主要包括：串传送指令、串比较指令、串搜索指令、串替换指令、串转换指令、串抽取指令等。

###### 8. 处理机控制类指令

处理机控制类指令用于对CPU实现控制，例如，对PSW中的标志实现置位或清零指令、停机指令、开中断指令、关中断指令、空操作指令等。

###### 9. 数据转换类指令

在功能较强的计算机中会有数据转换类指令。例如，将十进制数转换为二进制数、二进制数转换为十进制数、定点数和浮点数的相互转换等。

#### 四、指令的执行过程

通常一条指令的执行可分为以下步骤。

(1) 按指令指针计数器(IP)中的地址从内存读得一条指令存入CPU中的指令寄存器。

(2) 指令寄存器的指令经译码(与时序电路配合)有序地发出步骤(3)～步骤(6)的控制信号。

(3) 计算操作数的地址。

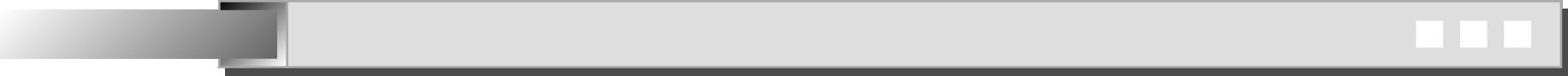
(4) 从该地址读出操作数。

(5) 执行本指令的运算操作。

(6) 保存结果。

(7) 计算下条指令的地址并存入指针计数器(IP)，转到步骤(1)进行下条指令的执行过程。

## 1.3 真题详解



### 综合知识试题

试题1(2010年下半年试题6)

中央处理单元(CPU)不包括 (1) 。

(1) A. 算术逻辑运算单元 B. 控制器 C. 通用寄存器组 D.  I/O总线

**参考答案：**(1)D。

**要点解析：**CPU包括运算器、控制器、寄存器三大部分。

试题2(2010年下半年试题7)

设内存按字节编址，若8K×8bit存储空间的起始地址为7000H，则该存储空间的最大地址编号为 (2) 。

(2) A. 7FFF　　 B. 8FFF　 C. 9FFF　　 D. AFFF

**参考答案：(2)B。**

**要点解析：**8bit是一个字节，1K=1024字节，8K×8bit=8×1024=8192(十进制)=2000H(十六进制)，存储空间的最大地址编号应为7000H+2000H-1=9000H-1=8FFFH。

试题3(2010年下半年试题8)

计算机中，执行一条指令所需要的时间称为指令周期，完成一项基本操作所需要的时间称为机器周期，时钟脉冲的重复周期称为时钟周期。因此 (3) 。

(3) A. 时钟周期大于机器周期　　 B. 时钟周期等于机器周期

C. 机器周期大于指令周期　　 D. 指令周期大于时钟周期

**参考答案：(3)D。**

**要点解析：**时钟周期是最基本的、最小的时间单位。指令周期是执行一条指令所需要的时间，一般由若干个机器周期组成。指令不同，所需要的机器周期数也不同。一条指令的执行过程划分为若干个阶段，如取指令、存储器读、存储器写等，每一项工作为一项基本操作，完成一项基本操作所需要的时间称为机器周期。可见，指令周期＞机器周期＞时钟周期。

试题4(2010年下半年试题9)

使用电容存储信息且需要周期性地进行刷新的存储器是 (4) 。

(4) A. DRAM　　 B. EPROM　　 C. SRAM　　 D. EEPROM

**参考答案：**(4)A。

**要点解析：**DRAM(Dynamic Random Access Memory)，即动态随机存储器最为常见的系统内存。DRAM 只能将数据保持很短的时间。为了保持数据，DRAM使用电容存储，所以必须隔一段时间刷新(refresh)一次，如果存储单元没有被刷新，存储的信息就会丢失(关机就会丢失数据)。

试题5(2010年下半年试题19)

用 (5) 时，校验位与信息位交叉设置。

(5) A. 循环冗余校验码　　B. 海明校验码　　C. 奇校验码　　D. 偶校验码

**参考答案：(5)B。**

**要点解析：海明码是利用奇偶性来检错和纠错的校验方法。海明码的构成方法是：在数据位之间插入*k*个校验位，通过扩大码距来实现检错和纠错。**

试题6和试题7(2010年下半年试题20和试题21)

若用8位机器码表示二进制数-111，则原码表示的十六进制形式为 (6) ；补码表示的十六进制形式为 (7) 。

(6) A. 81 　　 B. 87 　　 C. 0F 　　 D. FF

(7) A. F9 　　 B. F0 　　 C. 89 　　 D. 80

**参考答案：(6)B；(7)A。**

**要点解析：**[-111]原=10000111=87(十六进制)，[-111]反=11111000，[-111]补=11111001=F9  
(十六进制)。

试题8(2010年下半年试题22)

某逻辑电路有两个输入端和一个输出端，输入端用*X*和*Y*表示，输出端用*Z*表示。当且仅当*X*和*Y*同时为1时，*Z*才为0，则该电路的逻辑表达式为 (8) 。

(8) A. *X*·*Y* B.  C. *X*⊕*Y* D. *X*+*Y*

**参考答案：(8)B。**

**要点解析：用真值表验证知，题目所要求的表达式为**。

试题9(2010年上半年试题6)

将某ASCII字符采用偶校验编码(7 位字符编码+1 位校验码)发送给接收方，接收方收到的8位数据中，若 (9) ，则能确定传输过程中发生错误。

(9) A. “1”的个数为奇数     B. “1”的个数为偶数

C. 最低位为“1”             D. 最高位为“1”

**参考答案：**(9)A。

**要点解析：本题考查校验码的知识。采用偶校验编码时，数据位和校验位中“1”的个数应是偶数。当接收方收到的8位数据中“1”的个数为奇数时，可以确定传输过程中出错。**

试题10 (2010年上半年试题7)

若内存按字节编址，用存储容量为32K×8bit的存储器芯片构成地址编号A0000H 至DFFFFH 的内存空间，则至少需要 (10) 片。

(10) A.  4      B.  6       C.  8       D.  10

**参考答案：**(10)C。

**要点解析：地质编号A0000H至DFFFFH的内存空间中共有218(DFFFF-A0000=3FFFF)个存储单元，每个单元8bit，因此需要的存储器芯片数目为218/215=23个。**

试题11(2010年上半年试题8)

以下关于精简指令集计算机(RISC)指令系统特点的叙述中，错误的是 (11) 。

(11) A. 对存储器操作进行限制，使控制简单化

B. 指令种类多，指令功能强

C. 设置大量通用寄存器

D. 选取使用频率较高的一些指令，提高执行速度

**参考答案：**(11)B。

**要点解析：RISC指令系统的最大特点是：选取使用频率最高的一些简单指令，指令条数少；指令长度固定，指令格式种类少；只有取数/存数指令访问存储器，其余指令的操作都在寄存器之间进行。**

试题12(2010年上半年试题9)

32位微处理器的32是指 (12) 。

(12) A. 系统总线的宽度为32位    B. 处理的数据长度只能为32位

C.  CPU 字长为 32 位            D. 通用寄存器数目为32个

**参考答案：**(12)C。

**要点解析：在同一时间处理二进制数的位数称为字长。通常称处理字长为8位数据的CPU为8位CPU，32位CPU就是在同一时间内可处理字长为32位的二进制数据。**

试题13(2010年上半年试题10)

以下关于CPU与主存之间增加高速缓存(Cache)的叙述中，错误的是 (13) 。

(13) A. Cache扩充了主存储器的容量

B. Cache可以降低由于CPU 与主存之间的速度差异造成的系统性能影响

C. Cache的有效性是利用了对主存储器访问的局部性特征

D. Cache中通常保存着主存储器中部分内容的一份副本

**参考答案：**(13)A。

**要点解析：Cache(高速缓冲存储器)是一种特殊的存储器子系统，其中复制了频繁使用的数据以利于快速访问。Cache的出现是基于两种原因：首先是由于CPU的速度和性能提高很快而主存速度较低且价格高，其次就是程序执行的局部性特点。因此，将速度比较快而容量有限的SRAM构成Cache，目的在于尽可能发挥CPU的高速度。**

试题14和试题15(2010年上半年试题11和试题12)

衡量计算机的主要性能指标除了字长、存取周期、运算速度之外，通常还包括 (14) ，因为其反映了 (15) 。

(14) A. 外部设备的数量             B. 计算机的制造成本

C. 计算机的体积                 D. 主存储器容量大小

(15) A. 每秒钟所能执行的指令条数      B. 存储器读写速度

C. 计算机即时存储信息的能力      D. 该计算机保存大量信息的能力

**参考答案：**(14)D；(15)C。

**要点解析：试题(14)选D。因为计算机功能的强弱或性能的好坏，不是有某项指标来决定的，而是由它的系统结构、指令系统、硬件组成、软件配置等多方面的因素综合决定的。但主要性能指标是由字长、存取周期、运算速度以及主存储器容量的大小来决定的。**

**试题(15)选C。因为主存是CPU可以直接访问的存储器，需要执行的程序与需要处理的数据就是存放在主存中的。主存储器容量的大小反映了计算机即时存储信息的能力。随着操作系统的升级，应用软件的不断丰富及其功能的不断扩展，人们对计算机主存容量的需求也不断提高。**

试题16(2010年上半年试题14)

下列光盘格式中，不能多次擦除重写数据的是 (16) 。

(16) A. DVD-RAM  B. CD-R     C. DVD-RW    D. CD-RW

**参考答案：**(16)B。

**要点解析：题目给出的是常见的光盘格式，其实DVD-RAM和DVD-RW是DVD技术所支持的两种不同的可多次擦除重写的DVD光盘格式，CD-R指一次性可写(刻录)CD光盘，而CD-RW指可多次擦除、重写的CD光盘。**

试题17(2010年上半年试题19)

若不考虑I/O设备本身的性能，则影响计算机系统I/O数据传输速度的主要因素是  
 (17) 。

(17) A. 地址总线宽度           B. 数据总线宽度

C. 主存储器的容量        D.  CPU 的字长

**参考答案：**(17)B。

**要点解析：地址总线的宽度决定了CPU可以访问的物理地址空间，简单地说就是CPU到底能够使用多大容量的内存。CPU字长指CPU在单位时间内(同一时间)能一次处理的二进制数的位数。数据总线负责计算机中数据在各组成部分之间的传达，数据总线宽度是指在芯片内部数据传送的宽度，而数据总线宽度则决定了CPU与二级缓存、内存以及输入/输出设备之间一次数据传输的信息量。**

试题18(2010年上半年试题20)

十六进制数CC 所对应的八进制数为 (18) 。

(18) A. 314     B. 630      C. 1414      D. 3030

**参考答案：**(18)A。

**要点解析：十六进制数CC的二进制形式为11001100，从右向左每三位划分为一组，对应的八进制数为314。**

试题19(2010年上半年试题21)

**与*A*+****•*B*等价的逻辑表达式是 (19) 。(⊕、+、·分别表示逻辑异或、逻辑加、逻辑乘)**

**(19) A. *A*+****B. *A*+*B*     C. *A*⊕*B*      D. *A*·*B***

**参考答案：(19)B。**

**要点解析：由真值表验证知，与*A*+****·*B*等价的是*A*+*B*。**

试题20(2010年上半年试题22)

CPU 中的 (20) 的值可自动加1，以便实现程序指令的顺序执行。

(20) A. 指令寄存器(IR)       B. 程序计数器(PC)

C. 地址寄存器(AR)      D. 指令译码器(ID)

**参考答案：**(20)B。

**要点解析：指令寄存器用来保存当前正在执行的指令。当执行一条指令时，先把它从其内存取出放到数据寄存器中，然后再传送至IR。为了执行任何给定的指令，必须对操作码进行测试，以便识别所要求的操作。指令译码器就是做该工作的。指令寄存器中操作码字段的输出就是指令译码器的输入。操作码一经译码后，即可向操作控制器发出具体操作的特定信号。**

**地址寄存器用来保存当前CPU所访问的内存单元的地址。由于在内存和CPU之间存在着操作速度上的差别，所以必须使用地址寄存器来保持地址信息，直到内存的读/写操作完成为止。**

**为了保证程序指令能够连续地执行下去，CPU必须具有某些手段来确定下一条指令的地址。而程序计数器正是起到了这种作用，所以通常又称为指令计数器。在程序开始执行前，必须将它的起始地址，即程序的一条指令所在的内存单元地址送入PC，因此程序计数器的内容即是从内存提取的第一条指令的地址。当执行指令时，CPU将自动修改PC的内容，即每执行一条指令PC增加一个量，这个量等于指令所含的字节数，以便使其保持的总是将要执行的下一条指令的地址。由于大多数指令都是按顺序来执行的，所以修改的过程通常只是简单地对PC加1。**

试题21(2009年下半年试题6)

逻辑变量*X*、*Y*进行逻辑“异或”(用表示)运算的含义是：若*X*、*Y*取值相同(都为true或都为false)，则*X**Y*的值为false，否则*X**Y*的值为true。用逻辑“与”(∧)、“或”(∨)、“非”(-)表示*X**Y*的式子为 (21) 。

(21) A. (*X*∧*Y*)∧(*X*∧) B. (*X*∨*Y*)∧(∨*Y*)

C. (*X*∧)∨(∧*Y*) D. (*X*∨*Y*)∨(∨*Y*)

**参考答案：(21)C。**

**要点解析：**根据题意，异或代表当两者同为true或false时结果为false，其他情况均为true，即只要*X*、*Y*不同为true或同为false，结果就为true ，分析可知只有选项C符合要求。

试题22(2009年下半年试题7)

以下关于计算机中数据表示的叙述中，错误的是 (22) 。

(22) A. 计算机中的数值数据采用二进制表示，非数值性数据不使用二进制表示

B. 正整数的原码和补码表示形式相同，而负数的原码和补码表示形式不同

C. 数值中的小数点在硬件中不明确表示，而是采用约定位置的方式

D. 码长相同时，补码比原码可以多表示一个数

**参考答案：(22)**A。

**要点解析：**计算机中的所有数据都是采用1和0的二进制来表示的，选项A叙述错误。

试题23(2009年下半年试题8)

若阶码和尾数都采用补码表示，则格式浮点数的绝对值范围为 (23) 。

(23) A. 2-6～26 B. 2-63～264 C. 2-64～264 D. 2-64～263

**参考答案：(23)**D。

**要点解析：**本题考查的是浮点数的运算，根据浮点数的运算法则我们可以得知此浮点数的绝对值范围为2-64～263，所以本题正确答案为D。

试题24(2009年下半年试题9)

关于汉字编码的叙述，错误的是 (24) 。

(24) A. 采用矢量法表示汉字时，若两个汉字的笔画和字形不同，则它们的矢量编码一  
 定不同

B. 采用点阵法表示汉字时，若两个汉字的笔画和字形不同，则它们的点阵信息量  
 一定不同

C. 汉字的输入、存储和输出采用不同的编码，拼音码属于输入码

D. 汉字在计算机内存储时，其编码长度不能少于2字节

**参考答案：(24)**B。

**要点解析：**本题考查的是汉字编码的基础知识。采用点阵法表示汉字时，若两个汉字的笔画和字形不同，则它们的点阵信息量是可能相同的，所以本题应选择B。

试题25和试题26(2009年下半年试题10和试题11)

在微型计算机中，通常用主频来描述CPU的 (25) ；对计算机磁盘工作影响最小的因素是 (26) 。

(25) A. 运算速度 B. 可靠性 C. 可维护性 D. 可扩充性

(26) A. 温度 B. 湿度 C. 噪声 D. 磁场

**参考答案：**(25)A；(26)C。

**要点解析：**在微型计算机中，通常用主频来描述CPU的运算速度；对计算机磁盘工作影响最小的因素是噪声。

试题27(2009年下半年试题21)

某计算机的字长是32位，其内存容量是1GB，若内存空间按字编址，那么其地址范围是 (27) (十六进制表示)。

(27) A. 0000000～FFFFFFF B. 0000000～7FFFFFF

C. 30000000～3FFFFFFF D. 30000000～7FFFFFFF

**参考答案：(27)**A。

**要点解析：**本题考查的是内存编址的知识，按题中要求，1GB=230，字长32位=25，那么可以有215个这样的字，即地址有215个，十六进制表示的地址范围应该是0000000～FFFFFFF。

试题28(2009年下半年试题22)

二进制数11101.1001对应的八进制为 (28) 。

(28) A. 35.44 B. 35.11 C. 72.11 D. 73.10

**参考答案：(28)**A。

**要点解析：**本题考查的是计算机数的表示，根据二进制与八进制转换规则可知，对应的八进制数是35.44 。

试题29(2009年下半年试题23)

计算机各部件之间传输信息的公共通路称为总线，一次传输信息的位数通常称为总线的 (29) 。

(29) A. 宽度 B. 长度 C. 粒度 D. 深度

**参考答案：(29)** A。

**要点解析：**本题考查的是计算机总线的基本概念。计算机各部件之间传输信息的公共通路称为总线，一次传输信息的位数通常称为总线的宽度。

试题30(2009年上半年试题6)

若采用16-bit补码表示整数，则可表示的整数范围是 (30) 。

(30) A. [-215,215] B. (-215,215) C. (-215,215) D. [-215,215)

**参考答案：(30)** D。

**要点解析：**若采用16-bit补码表示整数，则可表示的整数范围应该是[-215, 215)。

试题31(2009年上半年试题7)

CPU执行算术运算或者逻辑运算时，算术运算部件(ALU)将计算结果保存在 (31) 中。

(31) A. 累加器AC B. 程序计数器PC

C. 指令寄存器IR D. 地址寄存器AR

**参考答案：(31)**A。

**要点解析：**CPU执行算术运算或者逻辑运算时，算术运算部件(ALU)将计算结果保存在累加器AC中。

试题32(2009年上半年试题8)

(32) 不是使用光(激光)技术来存取数据的介质。

(32) A. DVD B. EEPROM C. CD-ROM D. CD-RW

**参考答案：(32)**B。

**要点解析：**使用光(激光)技术来存取数据的介质包括DVD、CD-ROM和CD-RW等，不包括EEPROM。

试题33(2009年上半年试题9)

(33) 既有检错功能又有纠错功能。

(33) A. 水平奇偶校验 B. 垂直奇偶校验

C. 海明校验 D. 循环冗余校验

**参考答案：(33)**D。

**要点解析：**循环冗余校验既有检错功能又有纠错功能。

试题34和试题35(2009年上半年试题10和试题11)

计算机的用途不同，对其性能指标的要求也有所不同，以科学计算为主的计算机，对  
 (34) 要求较高，而且应该重点考虑 (35) 。

(34) A. 外存储器的读写速度 B. 主机的运算速度

C. I/O设备的速度 D. 显示分辨率

(35) A. CPU的主频和字长，以及内存容量

B. 硬盘读写速度和字长

C. CPU的主频和显示分辨率

D. 硬盘读写速度和显示分辨率

**参考答案：**(34) B；(35) A。

**要点解析：**计算机的用途不同，对其性能指标的要求也有所不同，以科学计算为主的计算机，对主机的运算速度要求较高，而且应该重点考虑CPU的主频和字长，以及内存的容量。

试题36(2009年上半年试题24)

若8位二进制数能被4整除，则其最低2位 (36) 。

(36) A. 不可能是01、00 B. 只能是10

C. 可能是01、00 D. 只能是00

**参考答案：(36)**D。

**要点解析：**若8位二进制数能被4整除，则其最低2位不可能是01和10，只能是00。因为前6位表示的数一定是4的整数倍，所以后两位只能是0。

试题37(2009年上半年试题20)

设两个8位补码表示的数*b*7*b*6*b*5*b*4*b*3*b*2*b*1和*a*7*a*6*a*5*a*4*a*3*a*2*a*1相加时溢出(*b*7、*a*7为符号标志)，则 (37) 。

(37) A. *b*7与*a*7的“逻辑或”结果一定为1

B. *b*7与*a*7的“逻辑与”结果一定为0

C. *b*7与*a*7的“逻辑异或”结果一定为1

D. *b*7与*a*7的“逻辑异或”结果一定为0

**参考答案：(37)**A。

**要点解析：**如果两个8位补码表示的数相加时发生溢出，则这两个数的符号位的逻辑或一定为1，所以本题正确答案为A。

试题38(2009年上半年试题21)

(38) 属于并行外部总线。

(38) A. PCI B. USB C. IEEE-488 D. IEEE-1394

**参考答案：(38)**C。

**要点解析：**PCI是系统总线，USB是串行总线，IEEE-1394也是高速串行接口标准，只有IEEE-488是并行外部总线。

试题39(2009年上半年试题22)

若指令系统中设置了专用I/O操作指令，则I/O接口 (39) 。

(39) A. 与内存单元必须统一编址

B. 可以独立编址

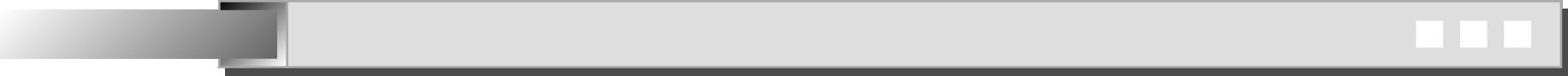
C. 必须采用DMA方式与内存交换数据

D. 必须采用中断方式与内存交换数据

**参考答案：(39)**B。

**要点解析：**若指令系统中设置了专用I/O操作指令，则I/O接口可以独立编址。其他三项的表述都是错误的。

## 1.4 强化训练



### 1.4.1 综合知识试题

试题1和试题2

如图1-16所示的微机主板的①处是 (1) 插槽，②处是 (2) 接口。

①

②



##### 图1-16 微机主板

(1) A. CPU B. PCI C. RAM D. IDE

(2) A. COM B. RJ-45 C. USB D. PS/2

试题3和试题4

主板(也称母板或系统板)是计算机硬件系统集中管理的核心载体，几乎集中了全部系统功能，是计算机中的重要部件之一。在如图1-17中所示的计算机主板上，①处是 (3) ，②处是 (4) 。

(3) A. CPU插槽 B. 内存插槽 C. PCI插槽 D. IDE插槽

(4) A. CPU插槽 B. 内存插槽 C. PCI插槽 D. IDE插槽



①

②

##### 图1-17 计算机主板

试题5

(5) 是指系统或其组成部分能在其他系统中重复使用的特性。

(5) A. 可扩充性 B. 可移植性 C. 可重用性 D. 可维护性

试题6

针对某计算机平台开发的软件系统，其　(6)　越高，越不利于该软件系统的移植。

(6) A. 效率　　 B. 成本　 C. 质量　　 D. 可靠性

试题7

在CRC(循环冗余校验)方法中，采用了 (7) 运算计算校验码。

(7) A. 逻辑与 B. 逻辑或 C. 循环移位 D. 模2除法(异或)

试题8

若内存按字节编址，用存储容量为8K×8bit的存储器芯片构成地址编号7000H至EFFFH的内存空间，则至少需要 (8) 片。

(8) A. 4 B. 6 C. 8 D. 10

试题9和试题10

已知*X*=–121，若采用8位机器码表示，则[*X*]原= (9) ，[*X*]补= (10) 。

(9) A. 11001001 B. 11111001 C. 01111001 D. 01011001

(10) A. 10110111 B. 10000111 C. 10100111 D. 01111001

试题11

将十六进制数9B转换为八进制数为 (11) 。

(11) A. 233 B. 433 C. 463 D. 531

试题12

已知某字符的编码为“0100101”，若最高位增加一个偶校验位，则其编码变为 (12) 。

(12) A. 10100101 B. 11001010 C. 01000110 D. 01010101

试题13和试题14

设机器码的长度为8，*X*为带符号纯小数，*Y*为带符号纯整数，[*X*]原 = 11111111，[*Y*]补 = 11111111，则*X*的十进制真值为 (13) ，*Y*的十进制真值为 (14) 。

(13) A. 1/128 B. –1/128 C. –127/128 D. 127/128

(14) A. –1 B. 127 C. –127 D. 1

试题15

欲知8位二进制数(*b*7*b*6*b*5*b*4*b*3*b*2*b*1*b*0)的*b*2是否为1，可将该数与二进制数00000100进行  
 (15) 运算，若运算结果不为0，则此数的*b*2必为1。

(15) A. 加 B. 减 C. 与 D. 或

试题16

汉字机内码与国标码的关系为：机内码 = 国标码 + 8080H。若已知某汉字的国标码为3456H，则其机内码为 (16) 。

(16) A. B4D6H B. B536H C. D4B6H D. C4B3H

试题17

用补码表示的8位二进制数11100000，其值为十进制数 (17) 。

(17) A. –31 B. –32 C. –64 D. –65

试题18

用ASCII码表示的大写英文字母B(42H)加偶校验后的二进制编码为 (18) 。

(18) A. 10001000 B. 10000010 C. 11000001 D. 01000010

试题19

两个带符号的数进行运算时，在 (19) 的情况下有可能产生溢出。

(19) A. 同符号数相加 B. 同符号数相减

C. 异符号数相加 D. 异符号数相“或”

试题20

无符号二进制数 100110.101 转换成的十进制数为　(20)　。

(20) A. 38.625　　　 B. 42.315　　　 C. 48.625 　　　 D. 68.815

试题21

用带符号位的定点补码表示纯小数，8 位编码 11111111 表示的十进制真值是 (21) 。

(21) A. 0 　　　 B. -1 　　 C. +2-7 D. -2-7

试题22

若信息为32 位的二进制编码，至少需要加　(22) 位的校验位才能构成海明码。

(22) A. 3 　　 B. 4 　　 C. 5 　　　 D. 6

试题23

某计算机中采用 48×48 数字化点阵字模表示一个汉字，字模中的每一个点在存储器中用一个二进制位存储。那么，存储 1024 个汉字要求的存储空间应为　(23) KB。

(23) A. 196 　　 B. 244 　 C. 288 　　　 D. 312

试题24

计算机系统中用来连接 CPU、内存储器和 I/O 接口的总线称为系统总线。 (24) 总线属于系统总线技术的一种。

(24) A. IEEE1394 B. PCI　 C. RS-232　 D. USB

试题25

微机系统中BIOS(基本输入输出系统)保存在 (25) 中。

(25) A. 主板上的ROM 　 B. DRAM

C. 主板上的RAM　 D. CD-ROM

试题26

(26) 不属于存储器的速度性能指标。

(26) A. 存储周期 　 B. 存取时间　 C. 主频　 D. 存储器带宽

试题27

下面关于Cache(高速缓冲存储器)的叙述， (27) 是错误的。

(27) A. 在体系结构上，Cache位于主存与CPU之间

B. Cache存储的内容是主存部分内容的复制副本

C. 使用Cache并不能扩大主存的容量

D. Cache的命中率只与其容量相关

试题28

计算机系统的可靠性通常用 (28) 来衡量。

(28) A. 平均响应时间 　 B. 平均故障间隔时间

C. 平均故障时间 　 D. 数据处理速率

试题29

计算机系统可维护性是指 (29) 。

(29) A. 对系统进行故障检测与修复的定期时间间隔

B. 系统失效后能被修复的概率

C. 在单位时间内完成修复的概率

D. 系统失效后在规定的时间内可修复到规定功能的能力

试题30和试题31

冯·诺依曼体系结构的计算机有两个主要组成部件：内存和处理器。其中，内存用于存储指令和数据。在大多数高级语言程序中， (30) 是内存单元的抽象。微处理器中的ALU可执行算术运算和 (31) 操作。

(30) A. 变量 B. 关键字 C. 语句 D. 数据类型

(31) A. 浮点 B. 定点 C. 逻辑 D. 控制

试题32

在软件开发中使用函数库可 (32) 。

(32) A. 提高软件的执行速度 B. 降低系统负载

C. 提高软件各部分之间的耦合度 D. 提高软件的可重用性

试题33

计算机中数据输入输出的控制方式有多种，“中断”方式的优点不包括 (33) 。

(33) A. I/O与CPU并行处理 B. 并行处理多种I/O

C. 实时响应突发事件 D. 批量传送数据

试题34

光盘驱动器与主机的接口总线常采用 (34) 总线。

(34) A. ISA B. CETRONIC C. EIDE(ATA) D. PCI

试题35

设某系统由P1和P2两个子系统组成，当且仅当两个子系统都能正常工作时，该系统才能正常工作。若P1和P2的可靠性均为0.9，则该系统的可靠性是 (35) 。

(35) A. 0.1 B. 0.45 C. 0.81 D. 0.9

试题36

评价计算机系统性能时，MIPS是衡量 (36) 的一种单位。

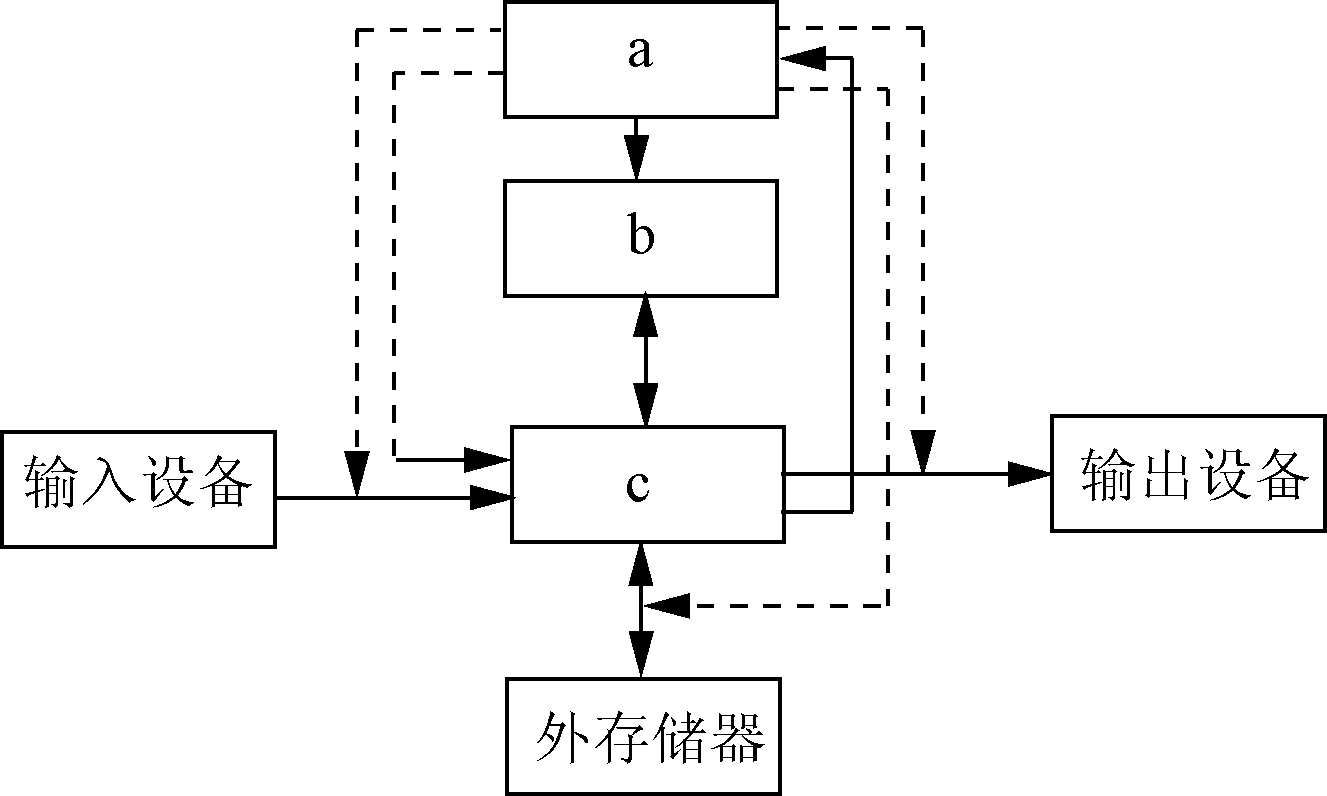
(36) A. 时钟频率 B. 运算速度 C. 系统可靠性 D. 系统失效率

试题37

计算机各功能部件之间的合作关系如图1-18所示。假设图中虚线表示控制流，实线表示数据流，那么 a、b 和 c 分别表示　(37)　。

(37) A. 控制器、内存储器和运算器 B. 控制器、运算器和内存储器

C. 内存储器、运算器和控制器 D. 内存储器、控制器和运算器



##### 图1-18 计算机各功能部件之间的合作关系图

试题38

使用 (38) 技术，计算机的微处理器可以在完成一条指令前就开始执行下一条指令。

(38) A. 迭代 B. 流水线 C. 面向对象 D. 中间件

试题39

CPU主要由运算单元(ALU)、控制单元(CU)、寄存器和时序电路组成，对指令进行译码的功能是由 (39) 实现的。

(39) A. 运算单元 B. 控制单元 C. 寄存器 D. 时序电路

试题40和试题41

(40) 用于存放执行指令的地址。若存储器按字节编址且指令长度为 16 位，则程序顺序执行时，每执行一条指令，程序计数器的值增加　(41)　。

(40) A. 算术逻辑单元 B. 指令寄存器 C. 程序计数器 D. 累加器

(41) A. 1 B. 2 C. 3 　　　　　 D. 4

### 1.4.2 综合知识试题参考答案



【试题1和试题2】答案：(1)A；(2)B。

解析：在微机主板中①处是CPU插槽，可以将CPU插入该插槽中。②处为RJ-45接口，可以用RJ-45连接线连接相关设备，如集线器、路由器等。

【试题3和试题4】答案：(3)B；(4)C。

解析：在计算机主板中①处是内存插槽，用来插内存条。②处为PCI接口卡插槽。

【试题5】答案：(5)C。

解析：可重用性是指系统或其组成部分能在其他系统中重复使用的特性。

【试题6】答案：(6)A。

解析：效率越高，软件系统与硬件/软件环境相关也就越大，所以越不利于该软件系统的移植。

【试题7】答案：(7)D。

解析：在CRC(循环冗余校验)方法中，采用了模2除法运算计算校验码。

【试题8】答案：(8)A。

解析：地址编号7000H～EFFFH的内存空间容量为215B，所以需要215×8/(8×210×8)=  
4片。

【试题9和试题10】答案：(9)B；(10)B。

解析：*X*=-1111001，[*X*]原=11111001，[*X*]补=10000111

【试题11】答案：(11)A。

解析：(9B)16 = (10 011 011)2 = (233)8

【试题12】答案：(12)A。

解析：偶校验是指数据编码(包括校验位)中“1”的个数应该是偶数。本题“0100101”中有3个“1”，校验位应设置为1，即“10100101”。

【试题13和试题14】答案：(13)C；(14)A。

解析：本题考查数值数据在计算机中的表示。

[*X*]原 =11111111时，*X*=-0.1111111=-127/128。

[*Y*]补= 11111111时，[*Y*]原 =10000001，则*Y*=-0000001=-1。

【试题15】答案：(15)C。

解析：如果要得到8位二进制数中某位的值，应屏蔽其他各位，而只保留该位进行与运算，然后根据得到的结果进行判别。如果是0说明该位为0；若是1，则说明该位为1。

【试题16】答案：(16)A。

解析：已知国标码为3456H，故机内码= 3456H + 8080H= B4D6H。

【试题17】答案：(17)B。

解析：将负数的补码表示转换为真值的方法是：除符号位外，数值位各位取反，然后再加1。所以，11100000的真值为-32。

【试题18】答案：(18)D。

解析：大写的英文字母B用7位二进制编码来表示为1000010。而加校验位后其编码为8位二进制数，且校验位应加在最高位上。加上偶校验后，包括校验位在内的1的个数应为偶数。所以，偶校验位应为0。加偶校验后的编码为01000010。

【试题19】答案：(19)A。

解析：溢出产生的原因是两数的运算结果超出了所规定的数值范围。而只有在两同符号数相加或异符号数相减时，才有可能出现溢出。

【试题20】答案：(20)A。

解析：将二进制数写成按权展开的多项式之和，可以计算得出十进制数为38.625。

【试题21】答案：(21)D。

解析：依题意，编码11111111所表示的十进制数是一个负数，其绝对值为01111111，因此该数应该为-2-7。

【试题22】答案：(22)D。

解析：构成海明码的条件是：2*k*-1≥*n*+*k*，其中，*n*为数据位数，*k*为校验位数。由此，可以得知，信息长度*n*=32时，*k*至少为6。

【试题23】答案：(23)C。

解析：由于每存储一个汉字需要48×48/8=288字节，所以1024个汉字要求的存储空间应该为288KB。

【试题24】答案：(24)B。

解析：系统总线又称为“内总线”，常见的系统总线技术有ISA总线、EISA总线和PCI总线等。

【试题25】答案：(25)A。

解析：BIOS是一组固定在计算机主板的ROM芯片上的程序。

【试题26】答案：(26)C。

解析：本题考查的是计算机系统的基础知识。存储器的速度性能指标包括存储周期、存取时间和带宽等，但不包括主频，主频是CPU的速度性能指标。

【试题27】答案：(27)D。

解析：Cache的命中率与下列因素有关：程序在执行过程中的地址流分布情况；当发生Cache块失效时，所采用的替换算法；Cache的容量；所采用的Cache预算法等。

【试题28】答案：(28)B。

解析：计算机系统的可靠性通常用平均故障间隔时间来衡量。平均故障间隔时间是指在规定时间内，设备无故障工作时间的平均值。

【试题29】答案：(29)D。

解析：计算机系统可维护性是指系统失效后在规定的时间内可修复到规定功能的能力。

【试题30和试题31】答案：(30)A；(31)C。

解析：在大多数高级语言程序中，将数据存储在变量中，程序运行时，要将全部数据放入内存中，因此变量是内存单元的抽象。微处理器主要由控制单元(CU)、算术逻辑运算单元(ALU)及寄存器组成，其中ALU可执行算术运算和逻辑运算。

【试题32】答案：(32)D。

解析：在软件开发中使用函数库主要是为了提高软件的可重用性，使这些函数库可重复使用。

【试题33】答案：(33)D。

解析：提高设备利用率的主要技术有中断技术、DMA技术、通道技术和缓冲技术。计算机中数据输入输出的控制方式有多种，其中“中断”方式的优点不包括批量传送数据。

【试题34】答案：(34)C。

解析：在当前PC中，光盘驱动器与主机的接口总线通常可采用SCSI和EIDE，但不能采用ISA、CETRONIC和PCI总线。

【试题35】答案：(35)C。

解析：依题意可知，两个子系统构成了串联系统。所以，整个系统的可靠性就是两个子系统可靠性的乘积，即0.9×0.9=0.81。

【试题36】答案：(36)B。

解析：评价计算机系统性能时，MIPS是衡量运算速度的一种单位。

【试题37】答案：(37)B。

解析：本题考查的是计算机硬件方面的基础知识。计算机有6种主要的硬件：控制器、运算器、内存储器、外存储器、输入和输出设备。根据计算机各硬件之间的合作关系可知，a应该为控制器，b为运算器，c为内存储器。

【试题38】答案：(38)B。

解析：本题考查计算机中流水线的概念。使用流水线技术，计算机的微处理器可以在完成一条指令前就开始执行下一条指令。

【试题39】答案：(39)B。

解析：CPU主要由运算单元、控制单元、寄存器和时序电路组成。对指令进行译码的功能是由控制单元实现的。

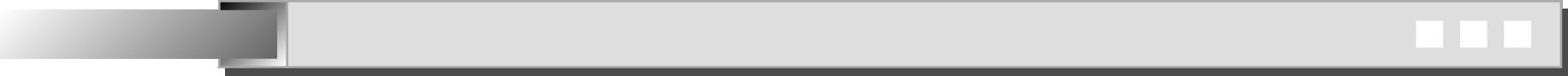
【试题40和试题41】答案：(40)C；(41)B。

解析：在CPU的部件中，程序计数器的功能是用于存放指令的地址。若存储器按字节编制，则16位长的指令占用2字节的空间，前后两条指令地址之间的偏移量等于2，所以每执行一条指令，程序计数器的值增加2。

# 第2章

# 操作系统基础知识

## 2.1 备考指南



### 2.1.1 考纲要求

根据考试大纲中相应的考核要求，在“操作系统基础知识”模块上，要求考生掌握以下方面的内容。

(1) 操作系统基础知识，包括操作系统的类型、功能。

(2) 处理机管理，包括进程的基本概念，进程的控制，进程间的通信，进程调度，信号量与P、V操作，高级通信原语，死锁和线程的基本概念等。

(3) 存储管理，包括主存保护、分区存储管理、分页存储管理、分段存储管理和虚存管理等。

(4) 设备管理，包括设备的类型、与设备分配有关的调度算法、通道、DMA与缓冲技术、假脱机和磁盘调度等。

(5) 文件管理，包括文件与文件系统的概念、文件的结构和组织等。

(6) 作业管理，包括作业管理的基本概念、作业调度及调度算法、评价作业调度算法应用的目的及对系统性能的影响。

(7) 图形用户界面和操作方法。

### 2.1.2 考点统计



“操作系统基础知识”模块，在历次程序员考试试卷中出现的考核知识点及分值分布情况如表2-1所示。

##### 表2-1 历年考点统计表

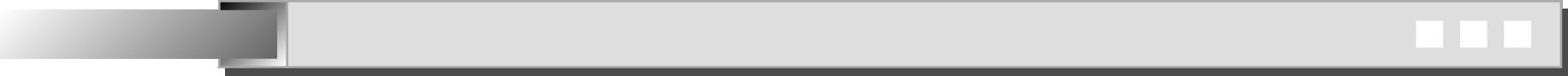
| 年 份 | 题 号 | 知 识 点 | 分 值 |
| --- | --- | --- | --- |
| 2009年  下半年 | 上半年： 26、27 | 处理机管理等 | 2分 |
| 下午题：无 |  | 0分 |
| 2009年  上半年 | 上午题：24、26、27 | 处理机管理、图形用户界面和操作方法等 | 3分 |
| 下午题：无 |  | 0分 |
| 2010年  下半年 | 上午题：25～27 | 文件管理、处理机管理等 | 3分 |
| 下午题：无 |  | 0分 |
| 2010年  上半年 | 上午题：23～27、30 | 存储管理、处理机管理等 | 6分 |
| 下午题：无 |  | 0分 |

### 2.1.3 命题特点



纵观历年试卷，本章知识点是以选择题的形式出现在试卷中的。本章知识点在历次考试的“综合知识”试卷中，所考查的题量大约为2～3道选择题，所占分值为2～3分(约占试卷总分值75分中的2.67%～4.00%)。

## 2.2 考点串讲



### 2.2.1 操作系统概述

#### 一、操作系统的定义

操作系统是计算机系统中最重要的系统软件，其他所有的软件都是建立在操作系统之上的，并在操作系统的统一管理和支持下运行。任何用户都是通过操作系统使用计算机的。

操作系统的定义为：操作系统(Operating System，OS)是计算机系统中的一个系统软件，它管理和控制计算机系统的硬件和软件资源，合理地组织计算机工作流程，以便有效地利用这些资源为用户提供一个功能强大、使用方便的工作环境，从而在计算机与用户之间起到接口的作用。

操作系统的主要任务是使硬件所提供的能力得到充分的利用，支持应用软件的运行并提供相应的服务。由于操作系统在计算机系统中占据着重要地位，所以它已经成为现代计算机系统中一个必不可少的关键组成部分。

#### 二、操作系统的作用

(1) 通过资源管理，提高工作效率。

操作系统的主要作用就是通过CPU管理、存储管理、设备管理和文件管理，对各种资源进行合理的分配，改善资源的共享和利用程度，最大限度地发挥计算机系统的工作效率，提高计算机系统的“吞吐量”(即系统在单位时间内处理工作的能力)。

(2) 改善人机界面，提供友好的工作环境。

操作系统既是计算机硬件和各种软件之间的接口，又是用户与计算机之间的接口。安装操作系统后，用户面对的不再是笨拙的裸机、由0和1组成的代码及一些难懂的机器指令，而是操作便利、服务周到的操作系统，操作系统明显地改善了用户界面，提高了用户的工作效率。

#### 三、操作系统的特征

操作系统主要有并发性(concurrency)、共享性(sharing)、虚拟性(virtual)和不确定性(non-determinacy)4个基本特征。

1) 并发性

并发性是指在计算机系统中存在着许多同时进行的活动。对计算机系统而言，并发是指宏观上看系统内有多道程序同时运行，微观上看实际上是串行运行。

2) 共享性

共享性是指系统中各个并发活动要共享计算机系统中的各种软、硬件资源，因此操作系统必须解决在多道程序间合理地分配和使用资源。

3) 虚拟性

虚拟性是操作系统中的重要特征，所谓虚拟是指把物理上的一台设备变成逻辑上的多台设备。例如，我们将在本章后面介绍的假脱机(spooling)技术，就是利用快速、大容量、可共享的磁盘作为中介，模拟多个非共享的低速的输入/输出设备，这样的设备称为虚拟设备。

4) 不确定性

通常一个程序的初始条件相同时，无论何时运行，结果都应该相同。但由于操作系统并发执行系统内的各种进程，与这些进程有关的事件如：从外部设备来的中断、输入输出请求、各种运行故障、发生的时间等都不可预测，如果处理不当，将导致系统出错，这种不确定性所带来的错误是很难查找的。

#### 四、操作系统的功能

1) 处理机管理

处理机是计算机系统的心脏，在单用户系统或单道系统中，处理机为一个用户或一个作业服务，其管理简单，但资源利用率低。为提高系统资源的利用率，引入了多道程序技术，即多个程序(作业)同时运行。在多道程序或多用户的情况下，要组织多个作业同时运行，对多个用户进行响应，就需要解决对处理机的分配、调度和资源回收等问题。处理机管理负责解决如何把CPU时间合理地、动态地分配给程序运行的基本单位——进程，使处理机得到充分的利用。许多操作系统是以作业和进程的方式进行管理的，实现作业和进程的调度，分配处理机，控制作业和进程的执行。现代的操作系统还引入了线程(thread)作为分配处理机的基本单位。

由于操作系统对处理机的管理策略不同，其提供的作业处理方式也就不同，如批处理方式、分时处理方式和实时处理方式，从而呈现在用户面前的就有不同的操作系统。在操作系统中，最重要的资源是处理机，最重要的管理是处理机管理。

2) 存储管理

计算机系统中，存储器(一般称为主存或内存)是运行程序和存放工作数据的部件，存储管理的工作主要是对内存储器进行分配、扩充和保护。

* 内存分配：在内存中除了操作系统和其他系统软件外，还要有一个或多个用户程序。如何分配内存，以保证系统及各用户程序的存储区互相不冲突，是内存分配所要解决的问题。
* 存储保护：系统中有多个程序在运行，如何保证一道程序在执行过程中不会有意或无意地破坏另一道程序？如何保证用户程序不会破坏系统程序？这些就是存储保护问题。
* 内存扩充：当用户作业所需要的内存量超过计算机系统所提供的内存容量时，如何把内部存储器和外部存储器结合起来管理，为用户提供一个容量比实际内存大得多的虚拟存储器，使这个虚拟存储器和内存一样方便使用，这就需要使用内存扩充。

存储器是计算机系统中最重要的资源之一，因为任何程序和数据，以及各种控制用的数据结构，都必须占有一定的存储空间，因此，存储管理的目的就是尽量提高内存的使用效率。存储管理的好坏直接影响着系统性能。

3) 设备管理

现代计算机系统常常配置很多种类的输入/输出设备，它们的输入/输出速度差别很大。计算机系统常常采用通道、控制器和设备3级控制方法管理这些设备。设备管理的任务就是监视这些资源的使用情况，根据一定的分配策略，把通道、控制器和设备分配给请求输入/输出操作的程序，并启动设备完成所需的操作。为了发挥设备和处理机的并行工作能力，常常采用缓冲技术和虚拟技术。

由于输入/输出设备种类很多，使用方法各不相同，因此，设备管理应为用户提供一个良好的界面，使具体的设备特性透明化，以便用户能方便、灵活地使用这些设备。

4) 文件管理(信息管理)

文件管理是对系统软件资源的管理。对用户来说，文件系统是操作系统中最直观的部分。我们把程序和数据统称为信息或文件，当一个文件暂时不用时，就把它放到外部存储器(如磁盘、磁带和光盘等)上保存起来。对这些文件如果不能很好地进行管理，就会引起混乱，甚至使其遭受破坏。这就是文件管理需要解决的问题。

文件管理的功能包括：建立、修改和删除文件；按文件名进行访问；决定文件信息的存放位置、存放形式及存取权限；管理文件间的联系及提供对文件的共享、保护和保密等，允许多个用户协同工作又不引起混乱。

5) 用户接口(作业管理)

上述4项功能是操作系统对软、硬件资源的管理。除此以外，操作系统也必须为用户提供一个友好的用户接口——命令接口和图形接口。一般来说，用户通过两种命令接口请求操作系统的服务。一种接口是作业一级的接口，即提供一组控制操作命令，如UNIX的Shell命令语言或作业控制语言(JCL)让用户组织和控制自己作业的运行。作业控制又分成两类：联机控制和脱机控制。另一种用户接口是程序一级的接口(编程接口)，即提供一组广义指令(或称系统调用、程序请求)供用户程序和其他系统程序调用。当这些程序要求进行数据传输、文件操作或有其他资源要求时，通过这些广义指令向操作系统提出申请，并由操作系统代为完成。

操作系统对计算机的资源进行全面管理，它的基本特征是多任务并行和多用户资源共享。多任务并行是指操作系统可以支持用户同时提交多项任务，同时工作；资源共享是指系统中的资源为多个用户共同使用。

#### 五、操作系统的类型

根据操作系统的使用环境和对作业的处理方式来划分，操作系统主要有以下几种基本类型。

1) 批处理操作系统

在批处理操作系统(Batch Processing Operating System)中，系统操作员将作业成批提交，由操作系统选择作业调入内存加以处理，最后由操作人员将运行结果交给用户。

批处理系统的特点：一是“多道”，指系统内可同时容纳多个作业；二是“成批”，指系统成批自动运行多个作业。批处理系统的目标是提高资源利用率和实现作业执行的自动化。

批处理操作系统分为单道批处理和多道批处理两种。

* 单道批处理操作系统：一次可提交多个作业，而不是单个作业。当一个作业运行结束后，随即自动调入同批的下一个作业运行，从而节省了作业之间的人工操作时间，提高了资源的利用率。早期单道批处理系统解决了作业自动转换问题，从而减少了作业建立和人工操作的时间。单道批处理存在的主要问题是：CPU和I/O设备使用忙闲不均(取决于当前作业的特性)，对以计算为主的作业，外设空闲；对以I/O为主的作业，CPU空闲。
* 多道批处理操作系统：正是为了解决单道批处理操作系统存在的问题而产生了多道批处理操作系统。它除了保持作业自动转换的功能外，还能支持同一批中的多道用户程序在一个CPU上同时运行。作业调度程序从后备作业中选取多个作业进入主存，在任意一个时刻，每当运行中的一个作业因输入/输出操作而需要调用外部设备时，就把CPU及时交给另一道等待运行的作业，从而将主机与外部设备的工作方式由串行改变为并行，进一步避免了因主机等待外设完成任务而白白浪费宝贵的CPU时间的情况。

2) 分时操作系统

分时操作系统(Time Share Operating System)是指一台计算机连接多个终端，系统把CPU时间分为若干个时间片，采用时间片轮转的方式处理用户的服务请求，对每个用户能保证及时响应，并提供交互会话能力。

分时操作系统具有下述特点。

* 多用户同时性：允许多个用户同时联机使用计算机。
* 交互性：每个用户可随时通过终端向系统提出服务请求，系统也可随时通过终端响应用户，从而加快了调试过程。
* 独立性：由于采用时间片轮转方式使一台计算机同时为多个用户服务，对于每个用户的操作命令又能快速响应，因此，用户彼此之间都感觉不到别人也在使用同一台计算机，如同自己独占计算机一样。
* 及时性：系统对用户的响应非常及时，不会让用户等待执行命令的处理时间过长。

分时操作系统的主要目标是保证用户响应的及时性。通常，计算机系统中往往同时采用批处理和分时处理方式来为用户服务，即时间要求不强的作业放入“后台”(批处理)处理，需频繁交互的作业放在“前台”(分时)处理。

3) 实时操作系统

实时操作系统(Real Time Operating System)是随着计算机应用于实时控制和实时信息处理而发展起来的。实时操作系统是指系统能够及时响应事件，并以足够快的速度完成对该事件的处理。实时操作系统包括实时控制系统和实时处理系统。实时控制是指生产过程控制(如炼钢、电力生产和数控机床)及武器控制等；实时处理是指实验数据采集和订票系统等。

实时操作系统的主要特点是及时性和高可靠性。

4) 网络操作系统

网络操作系统(Network Operating System)开发是在原来各自计算机操作系统的基础上，按照网络体系结构的协议、标准进行开发的，包括计算机网络管理、通信、资源共享、系统安全和多种网络应用服务等。其功能主要包括高效、可靠的网络通信；对网络中共享资源的有效管理；电子邮件、文件传输、共享硬盘、打印机等服务；网络安全管理；互操作能力。

5) 分布式操作系统

分布式操作系统(Distributed Operating System)与网络操作系统都是工作在一个由多台计算机组成的系统中，这些计算机之间可以通过一些传输设备进行通信和系统资源共享。分布式操作系统更倾向于任务的协同执行，并且各系统之间无主次之分，系统之间也无须采用标准的通信协议进行通信。分布式操作系统基本上废弃(或改造)了各单机的操作系统，整个网络设有单一的操作系统，由这个操作系统负责整个系统的资源分配和调度，为用户提供统一的界面。用户在使用分布式操作系统时不需要像使用网络操作系统那样，指明资源在哪台计算机上，因此分布式操作系统的透明性、稳固性、统一性及系统效率都比网络操作系统要强，但实现起来难度也大。分布式操作系统对于多机合作和系统重构、稳固性和容错能力有更高的要求，希望分布式操作系统有更短的响应时间、更大的吞吐量和更高的可靠性。

分布式操作系统与网络操作系统最大的差别是：网络操作系统的用户必须知道网址，而分布式操作系统的用户则不必知道计算机的确切地址；分布式操作系统负责全系统的资源分配，通常能很好地隐藏系统内部的实现细节，如对象的物理位置、并发控制、系统故障处理等对用户都是透明的。

6) 微机操作系统

微机操作系统(Microcomputer Operating System)是指配置在微型计算机上的操作系统。常用的微机操作系统有DOS、Windows、OS/2、UNIX和Linux等。其中，Microsoft公司开发的单用户单任务操作系统DOS是首先在IBM-PC机上使用的微机操作系统。MS-DOS操作系统是16位微机单用户单任务操作系统的标准。多任务操作系统Windows 98/NT/2000/XP是Microsoft公司开发的一系列图形用户界面的多任务、多线程的操作系统。

7) 嵌入式操作系统

嵌入式操作系统(Embedded Operating System)运行在嵌入式智能芯片环境中，对整个智能芯片及其控制的各种部件和装置等资源进行统一协调、处理、指挥和控制。

#### 六、研究操作系统的观点

研究和分析操作系统，可以从资源管理观点和虚拟机观点出发。

1) 资源管理观点

引入操作系统是为了合理地组织计算机的工作流程，管理和分配计算机系统硬件和软件资源，使资源能为多个用户共享。因此，操作系统是计算机资源的管理者。

这里的资源是指计算机系统进行数值计算和数据处理所需的物质基础，通常分为系统硬件资源和软件资源。硬件资源是组成计算机和计算机操作所需的物理实体，它们是看得见摸得着的设备，如处理机、存储器及输入/输出设备(键盘、显示器、打印机和磁盘等)。软件资源是依赖于一定的物理实体才能被人们所感知的一类资源，如程序和数据等，它们可经显示器或打印机等设备展现给用户。操作系统是控制和管理计算机系统资源的一组程序，其工作是当用户程序和其他程序争用这些资源时提供有序的和可控的分配。

我们通常将操作系统分为CPU管理、存储管理、设备管理、文件管理、用户与操作系统接口5个主要部分。主要研究资源的使用情况、资源的分配策略及分配和回收资源。

2) 虚拟机观点

从服务用户的机器扩充的观点来看，操作系统为用户使用计算机提供了许多服务功能和良好的工作环境。用户不再直接使用硬件机器(称为裸机)，而是通过操作系统来控制和使用计算机，从而把计算机扩充为功能更强、使用更方便的计算机系统(称为虚拟计算机)。操作系统的全部功能，如系统调用、命令、作业控制语言等，称为操作系统虚拟机。

虚拟机观点从功能分解的角度出发，考虑操作系统的结构，将操作系统分成若干个层次，每一层次完成特定的功能，从而构成一个虚拟机，并为上一层次提供支持，构成它的运行环境。这样，通过逐个层次的功能扩充最终完成操作系统虚拟机，从而向用户提供各种服务，完成用户的各项任务。

### 2.2.2 处理机管理



#### 一、基本概念

在计算机系统上运行的程序是指令的集合，每一个程序完成特定的任务。在只允许一个程序运行的系统(称为单道系统)中，这个程序独占系统资源，而系统按程序的指令顺序运行，程序的顺序执行有两个基本特征：程序的封闭性和程序的可再现性。

* 封闭性：指程序运行时独占系统资源，只有程序本身能改变系统的状态。
* 可再现性：指程序运行不受外部因素的影响，只要初始条件相同，运行结果就相同。

多道程序系统让多个程序在系统中轮流运行，当一个程序不用处理机时，另一个程序就使用。也就是说，处理机在程序间来回切换，从而获得宏观上的并行(微观上的串行)，以提高处理机的利用率。这种切换，通常是由中断引起的。由于中断以不可预测的次序发生，即程序的指令执行序列也以不可预测的次序前进，这样就会产生操作系统的另一个特性——不确定性。即在多道程序系统中，顺序程序的封闭性和可再现性消失了，需要采用一个新的概念——进程来描述程序的执行。进程是运行中的程序，是系统进行资源分配和调度的独立单位。

###### 1. 进程及其组成

进程是一个程序关于某个数据集的一次运行。进程是一个动态的概念，而程序是静态的概念，是指令的集合。因此，进程具有动态性和并发性。

进程通常由程序、数据和进程控制块(PCB)组成。程序是进程运行所对应的运行代码，一个进程对应于一个程序，一个程序可以同时对应于多个进程，代码在运行过程中不会被改变的程序，常称为纯码程序或可重入程序，这类程序是可共享的程序。

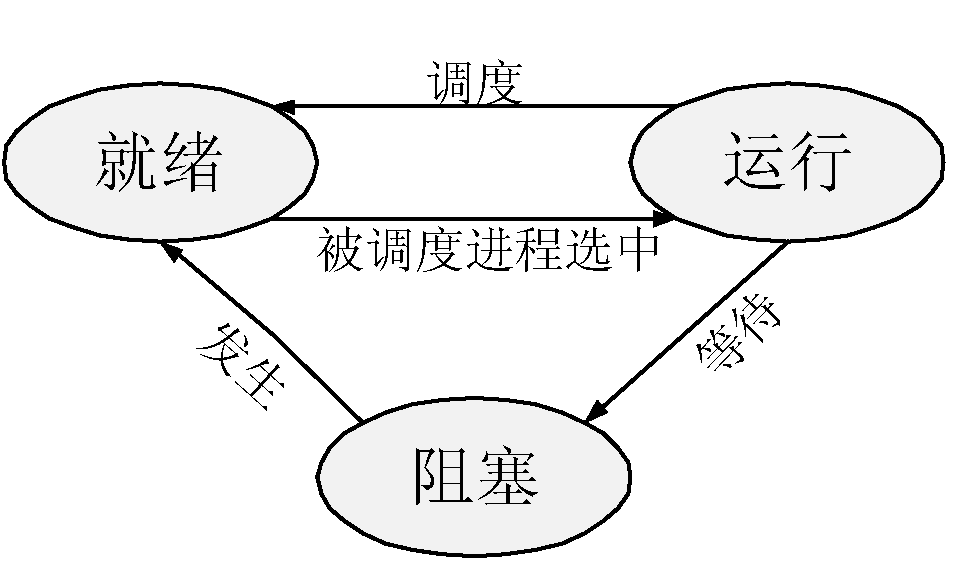
进程控制块是进程动态特性的集中反映，也是进程存在的唯一标志。在操作系统中，进程是进行系统资源分配、调度和管理的最小单位。现代操作系统中还引入了线程，线程是比进程更小的、能独立运行的基本单位，在引入线程的操作系统中，线程是进程中的一个实体，是CPU调度和分派的基本单位，是处理机分配的最小单位。

###### 2. 进程的状态及其转换

在多道系统中，进程的运行是走走停停的，在处理机上的交替运行，使它的运行状态不断变化。进程的状态主要有三态模型和五态模型。三态模型中最基本的状态有3种：运行、就绪和阻塞。

* 运行(running)：正占用处理机。
* 就绪(ready)：只要获得处理机即可运行。
* 阻塞(blocked)：也称等待或挂起状态，正等待某个事件(如I/O完成)的发生。

在进程运行的过程中，由于自身进展情况及外界环境的变化，这3种基本状态可以在一定的条件下相互转换，进程的状态及转换如图2-1所示。



##### 图2-1 进程状态及其转换图

五态模型比三态模型更加复杂，在三态模型的基础上增加了新建态和终止态。新建态对应于进程刚刚被创建还没有被提交时的状态，此时应在等待系统完成创建进程的所有必要信息。创建进程时分两个阶段：第一个阶段为一个新进程创建必要的管理信息；第二个阶段让该进程进入就绪状态。有了新建态，操作系统往往因系统的性能和内存容量的限制推迟新建态进程的提交。进程的终止态也可分为两个阶段：第一个阶段等待操作系统进行善后处理；第二个阶段释放内存。

#### 二、进程的控制

进程的控制就是对系统中所有进程从创建到消亡的全过程实施有效的控制。不仅要控制正在运行的进程，而且还要能创建新的进程，撤销已完成的进程。进程的控制机构是由操作系统内核实现的。通常将与硬件密切相关的模块放在紧挨硬件的软件层中，并使它们常驻内存，以便提高操作系统的运行效率，通常将这部分称为操作系统的内核，它为系统对进程进行控制和对存储器进行管理提供了有效的控制机制。

###### 1. 支撑功能

1) 中断处理

操作系统的各种重要活动最终都依赖于中断。例如，各种类型的系统调用、键盘命令的输入、设备驱动及文件系统等都依赖于中断。通常内核只对中断进行“有限次处理”，然后转入有关进程继续处理。这不仅可以减少中断处理的时间，还可以提高程序的并发性。

2) 时钟管理

操作系统的许多活动要用到时钟管理。如在分时系统时间片调度算法中，当时间片用完时，由时钟管理产生一个中断信号，通知调度程序重新调度。在实时系统中的截止时间控制、批处理系统中的最长运行时间的控制等都要用到时钟管理。

3) 原语操作

内核在执行某些基本操作时，往往是通过原语操作来实现的。原语是由若干条机器指令构成的，用于完成特定功能的一段程序。原语在执行的过程中是不可分割的。进程控制原语主要有：创建原语、撤销原语、挂起原语、激活原语、阻塞原语以及唤醒原语。

###### 2. 资源管理功能

资源管理功能包括：进程管理、存储器管理和设备管理。

#### 三、进程间的通信

###### 1. 同步与互斥

在操作系统中，多个进程并发执行，因此进程间必然存在资源共享和相互合作的问题。

1) 进程间的同步

一般情况下，一个进程相对于另一个进程的速度是不可预测的，也就是说，进程之间是异步运行的。为了成功地协同工作，有关进程在某些确定的点上应当保持同步：一个进程到达了这些点后，除非另一进程已经完成了某个活动，否则就停下来，等待该活动结束。

同步是指进程之间的一种协同工作关系，使这些进程相互合作，共同完成一项任务。进程间的直接相互作用构成进程的同步。同步机制应满足的基本要求是：有描述能力、可以实现、效率高、使用方便。

2) 进程间的互斥

在多道系统中，各进程可以共享各类资源，但有些资源却一次只能供一个进程使用。这种资源称为临界资源，如打印机、公共变量、表格等。互斥是要保证临界资源在某一时刻只被一个进程访问。

3) 临界区管理的原则

临界区是进程中对临界资源实施操作的那段程序。对互斥临界区管理的原则是：有空即进、无空则等、有限等待、让权等待。

###### 2. 信号量机制

信号量机制是一种有效的进程同步与互斥工具。目前主要有：整型信号量、记录型信号量、信号量集机制。

1) 整型信号量与P操作和V操作

信号量是一个整型变量，根据控制对象的不同被赋予不同的值。信号量分为两类：公用信号量，实现进程间的互斥，初值为1或资源的数目；私用信号量，实现进程间的同步，初值为0或某个正整数。

信号量S的物理意义：S≥0表示某资源的可用数，若S<0，则其绝对值表示阻塞队列中等待该资源的进程数。

除了设置初值外，对信号量只能进行特殊的操作：P操作和V操作。P操作和V操作都是不可分割的原子动作，也称为原语，其中P操作表示申请一个资源，V操作表示释放一个资源。

P操作和V操作都是原语。利用信号量S的取值表示共享资源的使用情况。在使用时，把信号量S放在进程运行的环境中，赋予其不同的初值，并在其上实施P操作和V操作，以实现进程间的同步与互斥。

P操作和V操作的定义如下。

P(S)：①S=S-1；②若S<0，则该进程进入S信号量的队列中等待。

V(S)：①S=S+1；②若S≤0，则释放S信号量队列上的一个等待进程，使之进入就绪队列。

当S>0时，表示还有资源可以分配；当S<0时，其绝对值表示信号量等待队列中进程的数目。每执行一次P操作，意味着要求分配一个资源；每执行一次V操作，就意味着释放一个资源。

2) 利用P操作和V操作实现进程的互斥

令信号量mutex的初值为l，进入临界区时执行P操作，退出临界区时执行V操作，于是临界区就改写成下列形式的代码段：

P(mutex);

临界区

V(mutex);

由于mutex初值为1，P、V是原子操作，可以实现互斥。

###### 3. 高级通信原语

P操作和V操作是用来协调进程间关系的，编程较困难、效率低，而且没有信息交换，故常称为低级通信原语。交换的信息量多时要引入高级通信原语，进程高级通信的类型主要有如下几种。

* 共享存储系统：相互通信的进程共享某些数据结构或存储区，以实现进程之间的通信。
* 消息传递系统：进程间的数据交换以消息为单位，程序员直接利用系统提供的一组通信命令(原语)来实现通信。如Send(A)、Receive(A)。
* 管道通信：所谓管道，是指用于连接一个读进程和一个写进程，以实现它们之间通信的共享文件(pipe 文件)。向管道(共享文件)提供输入的发送进程(即写进程)，以字符流的形式将大量的数据送入管道；而接受进程可从管道接受大量的数据。由于通信是采用管道的方式，所以称为管道通信。

#### 四、进程调度

进程调度即处理机调度，它的主要功能是确定在什么时候分派处理机，并确定分给哪一个进程。在一些操作系统中，一个作业从提交到完成需要经历高、中、低3级调度。

* 高级调度：又称“长调度”、“作业调度”或“接纳调度”，它决定处于输入池中的哪个后备作业可以调入主系统以便做好运行的准备，成为一个或一组就绪进程。系统中一个作业只需经过一次高级调度。
* 中级调度：又称“中程调度”或“对换调度”，它决定处于交换区中的哪个就绪进程可以调入内存，以便直接参与对CPU的竞争。在内存资源紧张时，为了将进程调入内存，必须将内存中处于阻塞状态的进程调至交换区，以便为调入进程腾出空间。
* 低级调度：又称“短程调度”或“进程调度”，它决定处于内存中的哪个就绪进程可以占用CPU，它是操作系统中最活跃、最重要的调度程序，对系统的影响很大。

###### 1. 调度方式

调度方式是指当有更高优先级的进程到来时该如何分配CPU。调度方式分为可剥夺式和不可剥夺式两种。可剥夺式是指当有更高优先级的进程到来时，强行将正在运行的进程所占用的CPU分配给高优先级的进程；不可剥夺式是指当有更高优先级的进程到来时，必须等待正在运行的进程自动释放占用的CPU，然后将CPU分配给高优先级的进程。

###### 2. 进程调度算法

常用的进程调度算法有：先来先服务、时间片轮转、优先级调度和多级反馈调度算法。

1) 先来先服务

先来先服务(FCFS)是按照作业提交或进程变为就绪状态的先后次序，分配CPU。即每当进入进程调度时，总是将就绪队列队首的进程投入运行。FCFS的特点比较有利于长作业，而不利于短作业；有利于CPU繁忙的作业，而不利于输入/输出繁忙的作业。

2) 时间片轮转

FCFS算法主要用于宏观调度，时间片轮转算法主要用于微观调度，通过时间片轮转，提高进程并发性和响应时间，从而提高资源利用率。

时间片轮转的实现过程是将系统中所有的就绪进程按照FCFS原则，排成一个队列。每次调度时将CPU分派给队首进程，让其执行一个时间片。时间片的长度从几毫秒到几百毫秒。在一个时间片结束时，发生时钟中断，调度程序据此暂停当前运行进程的执行，将其送到就绪队列的末尾，并通过上下文切换执行当前的队首进程。进程可以未使用完一个时间片，就出让CPU(如阻塞)。

时间片长度的确定主要考虑以下4个方面。

* 时间片长度变化的影响：时间片过长，退化为FCFS算法，进程在一个时间片内都执行完，造成响应时间长；时间片过短，用户的一次请求需要多个时间片才能处理完，上下文切换次数增加，系统效率降低，同样造成响应时间增长。
* 对响应时间的要求：T(响应时间)=N(进程数目)×q(时间片)。
* 就绪进程的数目：数目越多，时间片越小。
* 系统的处理能力：应当使用户输入在一个时间片内能处理完，否则会使响应时间、平均周转时间和平均带权周转时间延长。

3) 优先级调度

优先级调度分为静态优先级和动态优先级两种。

* 静态优先级：进程的优先级是在创建时就已确定好了，直到进程终止都不会改变。确定优先级的依据主要有进程类型(系统进程优先级较高)、对资源的需求(对CPU和内存需求较少的进程优先级较高)、用户要求(紧迫程度和付费多少)。
* 动态优先级：在创建进程时赋予一个优先级，在进程运行过程中还可以改变，以便获得更好的调度性能。进程每执行一个时间片，就降低其优先级，从而一个进程持续执行时，其优先级可能会降低到出让CPU为止。

4) 多级反馈调度

多级反馈调度算法是时间片轮转算法和优先级算法的综合与发展。其优点是：照顾了短进程、提高了系统吞吐量、缩短了平均周转时间；照顾输入/输出型进程，获得较好的输入/输出设备利用率和缩短响应时间；不必估计进程的执行时间，动态调节优先级。

#### 五、死锁

###### 1. 死锁的基本概念

当若干个进程竞争使用资源时，可能每个进程要求的资源都已被另一进程占用，于是也就没有一个进程能继续运行，这种情况称为死锁。例如，P1进程占有资源R1，P2进程占有资源R2，这时，P1又需要资源R2，P2也需要资源R1，它们在等待对方占有的资源时，又不会释放自己占有的资源，因而使双方都进入了无限等待状态。死锁是系统的一种出错状态，不仅浪费大量的系统资源，甚至还会导致整个系统的崩溃，所以死锁是应该尽量预防和避免的。

系统发生死锁时，死锁进程的个数至少为两个，所有死锁进程都有等待资源，其中至少有两个进程已占有资源。产生死锁的情况主要有：进程推进顺序不当；同类资源分配不当；PV操作使用不当。

###### 2. 产生死锁的4个必要条件

产生死锁的原因：一是系统提供的资源数量有限，不能满足每个进程的使用；二是多道程序运行时，进程推进顺序不合理。发生死锁必须同时具备下述4个条件。

* 互斥：进程互斥使用资源，任意时刻一个资源只为一个进程所独占，其他进程若请求一个已被占用的资源，只能等待占用者释放后才能使用。
* 不可剥夺(不可抢占)：进程所获得的资源在未使用完毕之前，不能被其他进程强行剥夺，而只能由获得该资源的进程自己释放。
* 请求保持：进程每次申请它所需要的一部分资源，在申请新的资源的同时，继续占用已分配到的资源。零星地请求资源，即已获得部分资源后再次请求资源时被阻塞。
* 循环等待：在进程资源有向图中存在一个进程环路，环路中每一个进程已获得的资源同时被下一个进程所请求。

进程资源有向图由方框、圆圈和有向边3个部分组成。其中，方框表示资源，圆圈表示进程。请求资源：○→□，箭头由进程指向资源；分配资源：○←□，箭头由资源指向进程。

###### 3. 解决死锁的方法

解决死锁的方法如下。

* 死锁的预防：根据产生死锁的4个必要条件，只要使其中之一不能成立，死锁就不会出现。
* 死锁的避免：最著名的死锁避免算法是Dijkstra提出的银行家算法。
* 死锁的检测：采用合理的死锁检测算法确定死锁的存在，并识别出与死锁有关的进程和资源，以供系统采用适当的解除死锁的措施。
* 死锁的解除：检测到死锁发生后，常采用资源剥夺法和撤销进程法解除死锁。

#### 六、线程

###### 1. 线程的基本概念

线程是比进程更小的能独立运行的基本单位。在引入线程的操作系统中，线程是进程中的一个实体，是CPU调度和分派的基本单位。线程自己基本上不占用系统资源，只占用一点儿在运行中必不可少的资源(如程序计数器、一组寄存器和栈)，但它可与同属一个进程的其他线程共享该进程所占用的全部资源。相应的，线程也同样有就绪、等待和运行3种基本状态。在有的系统中线程还有终止状态。

###### 2. 线程的属性

线程的属性如下。

* 每个线程都有一个唯一的标识符和一张线程描述表。
* 不同的线程可以执行相同的程序。
* 同一进程中的各个线程共享该进程的内存地址空间。
* 线程是处理机的独立调度单位，多个线程是可以并发执行的。
* 线程在生命周期内会经历等待状态、就绪状态和运行状态等各种状态变化。

###### 3. 引入线程的好处

传统的进程有两个基本属性：可拥有资源的独立单位、可独立调度和分配的基本单位。由于在进程的创建、撤销和切换中，系统必须为之付出较大的时空开销，因此在系统中所设置的进程数目不宜过多，进程切换的频率不宜太高，这就限制了并发程度的提高。引入线程后，将传统进程的两个基本属性分开，将线程作为调度和分配的基本单位，而将进程作为独立分配资源的单位。用户可以通过创建线程来完成任务，以减少程序并发执行时付出的时空开销。

引入线程的好处主要有如下几个。

* 创建一个新线程花费的时间少。
* 两个线程间切换花费的时间少。
* 由于同一进程内的线程共享内存和文件，线程之间相互通信无需调用内核，故不需要额外的通信机制，使通信更简便，信息传送速度也更快。
* 线程能独立执行，能充分利用和发挥处理机与外围设备并行工作的能力。

### 2.2.3 存储管理



#### 一、基本概念

现代计算机系统中的存储系统通常是多级存储体系，至少有主存(内存)和辅存(外存)两级，有的系统有更多级。系统中主存的使用一般分成两部分：一部分为系统空间，存放操作系统本身及相关的系统数据；另一部分为用户空间，存放用户的程序和数据。提高主存的利用率，对主存信息实现有效保护是存储器管理的主要任务。

###### 1. 存储器的结构

存储器的功能是保存数据，存储器的发展方向是高速度、大容量和小体积。一般存储器的结构有“寄存器－主存－外存”结构或“寄存器－缓存－主存－外存”结构。下面介绍几个与存储器相关的概念。

(1) 虚拟地址：数据的存放地址是由符号决定的，故又称为符号名地址，或者称为名地址，而把源程序的地址空间称为符号名地址空间或者名空间。它从0号单元开始编址，并顺序分配所有的符号名所对应的地址单元，所以它不是主存中的真实地址，故称为相对地址、程序地址、逻辑地址或虚拟地址。

(2) 地址空间：程序中由符号名组成的空间称为地址空间。源程序经过汇编或编译后再经过链接编辑程序加工形成程序的装配模块，即转换为相对地址编址的模块，它是以0为基址顺序进行编址的。相对地址也称为逻辑地址或虚拟地址，把程序中由相对地址组成的空间称为逻辑空间。相对地址空间通过地址重定位机构转换到绝对地址空间，绝对地址空间也称为物理地址空间。

(3) 存储空间：简单来说，逻辑地址空间(简称地址空间)是逻辑地址的集合，物理地址空间(简称存储空间)是物理地址的集合。

###### 2. 地址重定位

地址重定位是指程序的逻辑地址被转换成主存的物理地址的过程。在可执行文件装入时需要解决可执行文件中地址(指令和数据)和主存地址的对应关系。由操作系统中的装入程序Loader和地址重定位机构来完成。地址重定位分为静态地址重定位和动态地址重定位。

(1) 静态地址重定位，是指在程序装入主存时已经完成了逻辑地址到物理地址的转换，在程序的执行期间将不会再发生变化。其优点是：无需硬件地址转换机构的支持，只要求程序本身是可重定位的，它只对那些要修改的地址部分具有某种标识，由专门设计的程序来完成。

(2) 动态地址重定位，是指在程序运行期间完成逻辑地址到物理地址的转换。其实现机制要依赖硬件地址转换机构，如基地址寄存器BR。其优点是：程序在执行期间可以被换入和换出主存，以解决主存紧张的问题；可以在主存中移动，把主存中的碎片集中起来，以充分利用空间；不必给程序分配连续的主存空间，以便较好地利用较小的主存块，可以实现共享。

###### 3. 存储管理的功能

存储器管理的功能如下。

* 主存储器的分配和回收。
* 提高主存储器的利用率：减少碎片(也称零头)，允许多道程序动态共享主存。
* 存储保护：任务是确保每道程序都在自己的主存空间运行，互不干扰。
* 主存扩充：主存扩充的任务是从逻辑上扩充主存容量，使用户认为系统所拥有的主存空间远比其实际的主存空间(RAM)大得多。

#### 二、分区存储管理

存储管理主要包括分区存储管理、分页存储管理、分段存储管理、段页式存储管理和虚拟存储管理。其中分区存储管理是把主存的用户区划分成若干个区域，每个区域分配给一个用户作业使用，并限定它们只能在自己的区域中运行。按划分方式的不同，可分为固定分区、可变分区和可重定位分区。

* 固定分区。它是一种静态分区方式，在系统生成时已将主存划分为若干个分区，每个分区的大小可不等。
* 可变分区。它是一种动态分区方式，存储空间的划分是在作业装入时进行的，故分区的个数可变，分区的大小刚好等于作业的大小。
* 可重定位分区。它是解决碎片问题简单而又行之有效的方法。其基本思想是，移动所有已分配好的分区，使之成为连续区域。

分区划分完成后的问题就是如何进行分区的保护，通常采用上界和下界以及基址和限长寄存器保护法。

采用上界和下界寄存器保护法时，上界寄存器中存放作业的装入地址，下界寄存器装入作业的结束地址。形成的物理地址必须满足：

上界寄存器≤物理地址≤下界寄存器

采用基址和限长寄存器保护法时，基址寄存器中存放作业的装入地址，限长寄存器中装入作业的长度，形成的物理地址必须满足：

基址寄存器≤物理地址≤基址寄存器+限长寄存器

#### 三、分页存储管理

###### 1. 纯分页存储管理

分页原理：将一个进程的地址空间划分为若干个大小相等的区域，称为页。相应的，将内存空间划分成与页相同大小的若干个物理块，称为块或页框。

地址机构：分页系统的地址机构如图2-2所示，由两部分组成，页号P和偏移量W(即页内地址)。图中的地址长度为32位，其中0～11位为页内地址(每页大小为4KB)，12～31位为页号，所以允许的地址空间大小最多为1M个页。

31 12 11 0

|  |  |
| --- | --- |
| 页号(P) | 页内地址(W) |

##### 图2-2 分页系统的地址机构

系统将用户程序的逻辑空间按照同样大小也划分成若干页面，称为逻辑页面，有时也简称为页。程序的各个逻辑页面从0开始依次编号，称为逻辑页号或相对页号。每个逻辑页面内也从0开始编址，称为页内地址。用户程序的逻辑地址由逻辑页号和页内地址两部分组成。

页表：系统为每个进程建立一张页面映射表，简称页表。页表用于记录用户程序逻辑页面与内存物理页面之间的对应关系。页表的作用是实现从页号到物理块号的地址映射。

地址变换机构：其任务是利用页表把逻辑地址变换成内存中的物理地址。

###### 2. 快表

在地址映射过程中，共需两次访问内存。第一次是访问页表，得到数据的物理地址，第二次才是存取数据。显然，这样就增加了访问的时间。在地址映射机制中增加一个小容量的联想寄存器(相联存储器)，它由高速寄存器组成一张快表，用来存放当前访问最频繁的少数活动页的页号及相关信息。

快表只存放当前进程最活跃的少数几页，随着进程的推进，快表内容动态更新。当某一用户程序需要存取数据时，根据该数据所在的逻辑页号在快表中找出对应的物理页号，然后拼接页内地址，以形成物理地址；如果在快表中没有相应的逻辑页号，则地址映射仍然通过内存中的页表进行。

#### 四、分段存储管理

###### 1. 基本原理

在分段存储管理方式中，作业的地址空间被划分为若干个段，每个段是一组完整的逻辑信息，如有主程序段、子程序段、数据段及堆栈段等，每个段都有自己的名字，都是从0开始编址的一段连续的地址空间，各段长度是不等的。

分段系统的逻辑地址由段号(名)和段内地址两部分组成。在该地址结构中，允许一个作业最多有64K段，每个段的最大长度为64KB。

在分段式存储管理系统中，为每个段分配一个连续的分区，而进程中的各个段可以离散地分配到内存中不同的分区中。在系统中为每个进程建立一张段映射表，简称为“段表”。每个段在表中占用一个表项，在其中记录了该段在内存中的起始地址(又称为“基址”)和段的长度。进程在执行中，通过查段表来找到每个段所对应的内存区。所以说，段表实现了从逻辑段到物理内存区的映射。

###### 2. 段的动态链接和装配

所谓动态链接，是指在一个程序开始运行时，只将主程序装配好并调入内存，在运行过程中若访问一个新的模块时，再装配此模块，并与主程序链接起来。所以，动态链接是以段为基础的。

在可变分区分配方案中，主存中放置的程序常采用首次适应、最佳适应或最差适应算法实现，但运行的程序需连续存放在一个分区中，一个作业是由若干个具有逻辑意义的段(如主程序、子程序、数据段等)组成的。分段系统中，允许程序(作业)占据主存中若干分离的分区。每个分区存储一个程序分段。这样，每个作业需要几对界限地址，判定访问地址是否越界就困难了。在分段存储系统中常常利用存储保护键来实现存储保护。

#### 五、虚拟存储器管理

###### 1. 虚拟存储器的引入

1) 局部性原理

存储管理策略的基础是局部性原理——进程往往会不均匀地、高度局部化地访问主存。局部性表现为时间局部性和空间局部性两类。

* 时间局部性：是指最近被访问的存储位置，很可能不久的将来还要访问，如循环、栈等。
* 空间局部性：是指存储访问有成组的倾向，当访问了某个位置后，很可能还要访问其附近的位置，如访问数组、代码顺序执行等。

2) 虚拟存储器的定义

根据局部性原理，一个作业在运行之前，没有必要全部装入主存，而仅将当前要运行的那部分页面或段先装入主存启动运行，其余部分暂时留在磁盘上。

程序在运行时如果所要访问的页(段)已调入主存，便可继续执行下去；但如果所要访问的页(段)尚未调入主存(称为缺页或缺段)，程序应利用操作系统所提供的请求调页(段)功能，将它们调入主存，以使进程能继续执行下去。如果此时主存已满，无法再装入新的页(段)，则还要再利用页(段)的置换功能，将主存中暂时不用的页(段)调出至磁盘上，以便腾出足够的主存空间后，再将所要访问的页(段)调入主存，使程序继续执行下去。这样，便可使一个大的用户程序在较小的主存空间中运行，也可使主存中同时装入更多的进程并发执行。从用户角度看，该系统所具有的主存容量，将比实际主存容量大得多，人们把这样的存储器称为虚拟存储器。

虚拟存储器具有请求调入功能和置换功能，能仅把作业的一部分装入主存便可运行作业的存储器系统，能从逻辑上对主存容量进行扩充。

3) 虚拟存储器的实现

请求分页系统：在分页系统的基础上，增加了请求调页功能和页面置换功能所形成的页式虚拟存储系统。

请求分段系统：在分段系统的基础上，增加了请求调段功能和分段置换功能所形成的段式虚拟存储系统。

###### 2. 请求分页中的硬件支持

请求分页是目前常用的一种虚拟存储器方式。

1) 请求分页的页表机制

请求分页的页表机制是在纯分页的页表机制上形成的，由于只将应用程序的一部分调入主存，但还有一部分仍在磁盘上，故需在页表中再增加若干项，如状态位、访问字段、辅存地址等供程序(数据)在换进、换出时引用。

2) 缺页中断机构

在请求分页系统中，每当所要访问的页面不在主存时，便要产生一个缺页中断，请求操作系统将所缺页调入主存。与一般中断的主要区别在于：缺页中断在指令执行期间产生和处理中断信号，而一般中断在一条指令执行完后检查和处理中断信号。缺页中断返回到该指令的开始重新执行该指令，而一般中断则返回到该指令的下一条指令执行。

3) 地址转换机构

请求分页系统中的地址转换机构是在分页系统的地址转换机构的基础上，为实现虚拟存储器而增加了某些功能后形成的，如产生和处理缺页中断、从主存中换出一页等。

###### 3. 页面置换算法

1) 最佳置换算法

最佳(optimal)置换算法是一种理想化的算法，性能最好，但在实际上难于实现，所以该算法通常用来评价其他算法。

2) 先进先出置换算法

先进先出(FIFO)置换算法总是淘汰最先进入内存的页面。其算法实现简单，是一种最直观，也是性能最差的算法。

3) 最近最久未使用置换算法

最近最久未使用(Least recently used，LRU)置换算法是选择最近最久未使用的页面予以淘汰，系统在每个页面设置一个访问字段，用以记录这个页面自上次被访问以来所经历的时间T，当要淘汰一个页面时，选择T最大的页面。

4) 最近未用置换算法

最近未用(not used recently，NUR)置换算法将最近一段时间未引用过的页面换出，是一种LRU的近似算法。

### 2.2.4 设备管理



#### 一、设备管理概述

###### 1. 设备的分类

1) 按设备上数据组织方式分类

* 块设备：指以数据块为单位组织和传送数据的设备，如磁盘，磁带等。
* 字符设备：指以单个字符为单位存取信息的设备，如终端、打印机等。

2) 按资源分配的角度分类

* 独占设备：对这类设备来说，在一段时间内最多只能有一个进程占有并使用它。低速I/O设备一般是独占设备，如打印机、终端等。
* 共享设备：这类设备允许多个进程共享，即多个进程的I/O传输可以交叉。
* 虚拟设备：在一类设备上模拟另一类设备的技术称为虚设备技术。通常是用高速设备来模拟低速设备，以此把原来慢速的独占设备改造成能为若干进程共享的快速共享设备。就好像把一台设备变成了多台虚拟设备，从而提高了设备的利用率。我们称被模拟的设备为虚设备。Spooling技术就是一类典型的虚设备技术。

3) 按数据传输率分类

* 低速设备：指传输速率为每秒钟几个字节到数百个字节的设备。典型的设备有键盘、鼠标、语音的输入等。
* 中速设备：指传输速率在每秒钟数千字节至数万字节的设备。典型的设备有行式打印机、激光打印机等。
* 高速设备：指传输速率在数十万个字节至数兆字节的设备。典型的设备有磁带机、磁盘机、光盘机等。

4) 其他分类方法

按输入/输出对象的不同，设备可分为人机通信、机机通信设备。

按是否可交互，设备可分为非交互设备，如机机通信设备、外存、卡带机等；交互设备，如终端等。

###### 2. 设备管理的目标和任务

1) 操作系统设备管理的目标

* 向用户提供使用外围设备的方便、统一的接口。
* 充分利用中断技术、通道技术和缓冲技术，提高CPU与设备、设备与设备之间的并行工作能力，提高外围设备的使用效率。
* 保证在多道程序环境下，当多个进程竞争使用设备时，按照一定的策略分配和管理设备，以使系统能有条不紊地工作。

2) 设备管理的任务

设备管理的任务是保证在多道程序环境下，当多个进程竞争使用设备时，按一定策略分配和管理各种设备，控制设备的各种操作，完成输入/输出设备与内存之间的数据交换。

设备管理的主要功能如下所述。

* 动态地掌握并记录设备的状态。
* 设备分配和释放。
* 缓冲区管理。
* 实现物理输入/输出设备的操作。
* 提供设备使用的命令接口和编程接口。
* 设备的访问和控制，包括并发访问和差错处理。
* 输入/输出缓冲和调度，目的是提高输入/输出的访问效率。

#### 二、DMA与缓冲技术

###### 1. DMA技术

DMA技术基本思想是：在外围设备和主存之间开辟直接的数据交换通路。在内存与输入/输出设备间传送一个数据块的过程中，不需要CPU的任何干涉，只需要CPU在过程开始启动与过程结束时的处理，实际操作由DMA硬件直接执行完成。

###### 2. 缓冲技术

缓冲是计算机系统中常用的技术。一般来说，凡是数据到达速度和离去速度不匹配的地方都可以采用缓冲技术。缓冲可以采用硬件缓冲和软件缓冲两种技术。硬件缓冲是利用专门的硬件寄存器作为缓冲区；软件缓冲是利用操作系统的管理，用主存中的一个或多个区域作为缓冲区，进而可以形成缓冲池。

###### 3. Spooling技术

Spooling是外围设备联机操作(simultaneous peripheral operations on line)的缩写，常简称为Spooling系统或假脱机系统。假脱机技术实际上是用一类物理设备模拟另一类物理设备的技术，可以将低速的独占设备改造成一种可共享的设备，而且一台物理设备可以对应若干台虚拟的同类设备。Spooling系统由“预输入程序”、“缓输出程序”和“井管理程序”以及“输入和输出井”组成。

Spooling系统将一个作业从进入系统到完成后撤离系统的全过程，划分成输入、处理和输出3个并发执行的过程。当用户作业要进入系统时，由Spooling系统的预输入程序将作业信息从物理输入设备上送到磁盘上的指定区域(称为输入井)。输入井中的作业有4种状态。

* 输入状态。作业的信息正从输入设备上预输入。
* 收容状态。作业预输入结束但未被选中执行。
* 执行状态。作业已被选中并处于运行过程中，它可从输入井中读取数据信息，也可向输出井写信息。
* 完成状态。作业已经撤离，该作业的执行结果等待缓输出。

Spooling系统的引入缓和了CPU与设备速度的不均匀性，提高了CPU与设备的并行程度。

#### 三、磁盘调度

对磁盘的存取访问一般要有三部分时间。首先要将磁头移动到相应的磁道或柱面上，这个时间叫做寻道时间；一旦磁头到达指定磁道，必须等待所需要的扇区旋转到读/写头下，这个时间叫做旋转延迟时间；信息在磁盘和内存之间的实际传送时间叫做传送时间。磁盘调度的目的是使平均寻道时间最少。

###### 1. 磁盘驱动调度

一般可采用以下4种磁盘优化调度算法。

1) 先来先服务算法

先来先服务算法(FCFS)即按照访问请求的次序为各个进程服务，这是最公平且最简单的算法，但是效率不高。

2) 最短寻道时间优先算法

最短寻道时间优先算法(SSTF)以寻道优化为出发点，优先为距离磁头当前所在位置最近的磁道(柱面)的访问请求服务。这种算法使每次的寻道时间最短，但也存在缺点：不能保证平均寻道时间最短。

3) 扫描算法

扫描算法(SCAN)也是一种寻道优化的算法，它克服了SSTF算法的缺点，既考虑访问磁道与磁头当前位置的距离，又考虑磁臂的移动方向，且以方向优先。这种算法比较公平，而且效率较高。这种算法因其基本思想与电梯的工作原理相似，故又称为电梯算法。

4) 单向扫描调度算法

单向扫描调度算法(CSCAN)存在这样的问题：当磁头刚从里向外移动过某一磁道时，恰有一进程请求访问此磁道，这时该进程必须等待磁头从里向外，然后再从外向里扫描完所有要访问的磁道后，才处理该进程的请求，致使该进程的请求被严重地推迟。为了减少这种延迟，CSCAN算法规定磁头作单向移动。

###### 2. 旋转调度算法

系统应该选择延迟时间最短的进程对磁盘的扇区进行访问。当有若干等待进程请求访问磁盘上的信息时，旋转调度应考虑以下3种情况。

* 进程请求访问的是同一磁道上的不同编号的扇区。
* 进程请求访问的是不同磁道上的不同编号的扇区。
* 进程请求访问的是不同磁道上具有相同编号的扇区。

### 2.2.5 文件管理



#### 一、文件与文件系统

###### 1. 文件

文件是具有符号名的，在逻辑上具有完整意义的一组相关信息项的集合。文件可以是有格式的，也可以是无格式的。

信息项是构成文件的基本单位，可以是一个字符，也可以是一个记录。一个文件包括文件体和文件说明。

###### 2. 文件系统

操作系统的文件系统包括两个方面：一方面包括负责管理文件的一组系统软件；另一方面包括被管理的对象——文件。

文件系统的主要功能包括：按名存取、统一的用户接口、并发访问和控制、安全性控制、差错恢复等。

###### 3. 文件类型

根据文件的性质和用途不同，文件有多种分类方法。

* 按文件的用途，文件可以分为系统文件、库文件和用户文件等。
* 按信息保存期限，文件分为临时文件、档案文件和永久文件。
* UNIX系统将文件分为普通文件、目录文件和设备文件(特殊文件)等。
* 按文件的保护方式，文件可分为只读文件、读写文件、可执行文件和不保护文件等。

目前常用的文件系统类型有：FAT、VFAT、NTFS、Ext2、HPFS等。

文件分类的目的是对不同文件进行管理，提高系统效率和用户界面的友好性。

#### 二、文件的结构和组织

文件的结构是指文件的组织形式，从用户角度所看到的文件组织形式，称为文件的逻辑结构；从实现角度考察文件在辅助存储器上的存放方式，常称为文件的物理结构。

###### 1. 文件的逻辑结构

一般文件的逻辑结构可以分为两种：无结构的字符流文件和有结构的记录文件，后者也称为有格式文件。

记录文件可分为定长和不定长两类。

* 定长记录文件：指文件中所有记录的长度相同。
* 不定长记录文件：指文件中各记录的长度不相同。

在UNIX系统中，所有的文件都被看作是流式文件，系统不对文件格式进行处理。

###### 2. 文件的物理结构

文件的物理结构是指文件的内部组织形式，也就是文件在物理存储设备上的存放方法。常用的文件物理结构有以下3种。

1) 顺序结构

顺序结构又称连续结构。这是一种最简单的物理结构，它把逻辑上连续的文件信息依次存放在连续编号的物理块中。只要知道文件在存储设备上的起始地址(首块号)和文件长度(总块数)，就能很快地进行存取。这种结构的优点是访问速度快，缺点是很难增加文件的长度。

2) 链接结构

链接结构将逻辑上连续的文件分散存放在若干不连续的物理块中，每个物理块设有一个指针，指向其后续的物理块。只要指明文件的第一个块号，就可以利用链指针检索整个文件。这种结构的优点是文件长度容易动态变化，缺点是不适合随机存取访问。

3) 索引结构

采用索引结构时，系统为每个文件建立一张索引表，索引表中每一表项指出文件信息所在的逻辑块号和与之对应的物理块号。

对一些大的文件，索引表的大小超过一个物理块时，就会发生索引表的分配问题。一般采用多级(间接索引)技术，这时由索引表指出的物理块中存放的不是文件信息，而是存放文件信息的物理块地址。这样，如果一个物理块能存储*n*个物理块地址，则一次间接索引，可寻址的文件长度将变成*n*×*n*块。

UNIX文件系统采用三级索引结构，文件系统中的inode是基本的构件，它表示文件系统树形结构的节点。UNIX文件索引表项分为四种寻址方式：直接寻址、一级间接寻址、二级间接寻址和三级间接寻址。

#### 三、文件目录

系统为每个文件设置一个描述性数据结构——文件控制块FCB(File Control Block)，文件目录就是文件控制块的有序集合。

###### 1. 文件控制块

文件控制块是系统为管理文件而设置的一个数据结构。FCB是文件存在的标志，通常包含三类信息：基本信息类、存取控制信息类和使用信息类。

* 基本信息类：文件名、文件的物理位置、文件长度和文件块数等。
* 存取控制信息类：文件的存取权限。在UNIX系统中，将用户分成文件主、同组用户和一般用户3类，它们具有不同的操作权限。
* 使用信息类：建立文件的日期和时间、最后访问日期和时间、最后修改日期和当前使用的信息等。

为了实现文件目录的管理，通常将文件目录以文件的形式保存在外存空间，这个文件就被称为目录文件。目录文件是长度固定的记录式文件。

###### 2. 目录结构

文件目录的组织与管理是文件管理中的一个重要方面，常见的目录结构有3种：一级目录结构、二级目录结构和多级目录结构。

多级目录结构像一棵倒置的有根树，所以也称为树形目录结构。从树根向下，每一个节点是一个目录，叶节点是文件。DOS和UNIX等操作系统均采用多级目录结构。

在采用多级目录结构的文件系统中，用户要访问一个文件，必须指出文件所在的路径名。路径名包含从根目录开始到该文件的通路上所有各级目录名。各级目录名之间，目录名与文件名之间需要用分隔符隔开。例如，在DOS中分隔符为“\”，在UNIX中分隔符为“/”。绝对路径名是指从根目录开始的完整文件名，即由从根目录开始的所有目录名以及文件名构成的。

#### 四、存取方法和存取控制

###### 1. 文件的存取方法

文件的存取方法是指读写文件存储器上的一个物理块的方法。通常有顺序存取、随机存取和按键存取等方法。

1) 顺序存取

顺序存取就是按从前到后的次序依次访问文件的各个信息项。对于记录式文件，是按记录的排列顺序来存取的。对流式文件，顺序存取反映当前读写指针的变化，在存取完一段信息后，读写指针自动指出下次存取时的位置。

2) 随机存取

随机存取又称直接存取，即允许用户根据记录键存取文件的任意记录，或者是根据存取命令把读写指针移到指定处读写。

3) 按键存取

按键存取法是直接存取法的一种，它不是根据记录的编号或地址来存取文件中的记录，而是根据文件中各记录的某个数据项内容来存取记录的，这种数据项称之为“键”。因此，将这种存取法称为按键存取法。

###### 2. 文件存储空间的管理

外存空间管理的数据结构通常称为磁盘分配表(disk allocation table)。常用的空间管理方法有位示图、空闲区表、空闲块链和成组链接法。

1) 位示图

在外存上建立一张位示图(bitmap)，记录文件存储器的使用情况。每一位对应文件存储器上的一个物理块，取值0和1分别表示空闲和占用。文件存储器上的物理块依次编号为0，1，2，…，假如系统中字长为32位，那么在位示图中的第一个字对应文件存储器上的0，1，2，…，31号物理块；第二个字对应文件存储器上的32，33，34，…，63号物理块；以此类推。这种方法的主要特点是位示图的大小由磁盘空间的大小(物理块总数)决定，位示图的描述能力强，适合各种物理结构。

2) 空闲区表

将外存空间上一个连续未分配区域称为“空闲区”。操作系统为磁盘外存上所有空闲区建立一张空闲表，每个表项对应一个空闲区，空闲表中包含序号、空闲区的第一块号、空闲块的块数等信息。它适用于连续文件结构。

3) 空闲块链

每个空闲物理块中都有指向下一个空闲物理块的指针，所有空闲物理块构成一个链表，链表的头指针放在文件存储器的特定位置上(如管理块中)。

4) 成组链接法

在UNIX系统中，将空闲块分成若干组，每100个空闲块为一组，每组的第一个空闲块登记了下一组空闲块的物理盘块号和空闲块总数，假如一个组的第一个空闲块号等于0的话，意味着该组是最后一组，即无下一组空闲块。

#### 五、文件的使用

操作系统在操作级(命令级)和编程级(系统调用和函数)向用户提供文件的服务。操作系统在操作级向用户提供的命令有目录管理类命令、文件操作类命令(如复制、删除和修改)、文件管理类命令(如设置文件权限)等。操作系统在编程级向用户提供的系统调用主要有以下6种。

* 创建文件：如create(文件名，参数表)。
* 删除文件：如delete(文件名)。
* 打开文件：如open(文件名，参数表)。
* 关闭文件：如close(文件名)。
* 读文件：如read(文件名，参数表)。
* 写文件：如write(文件名，参数表)。

#### 六、文件的共享和保护

###### 1. 文件的共享

文件共享是指不同用户进程使用同一文件。常见的文件链接有硬链接和符号链接两种。

1) 硬链接

文件的硬链接是指两个文件目录表目指向同一个索引节点的链接，该链接也称基于索引节点的链接。文件硬链接不利于文件主删除它所拥有的文件，因为文件主要删除它所拥有的共享文件，必须首先删除(关闭)所有的硬链接，否则就会造成共享该文件的用户目录表目指针悬空。

2) 符号链接

符号链接是指建立的新的文件或目录与原来文件或目录的路径名映射。当访问一个符号链接时，系统通过该映射找到原文件的路径，并对其进行访问。符号链接的缺点：其他用户读取符号链接的共享文件和读取硬链接的共享文件相比，需要增加读盘操作的次数。

###### 2. 文件的保护

文件系统对文件的保护常采用存取控制方式进行，所谓存取控制就是不同的用户对文件的访问有不同的权限，以防止文件被未经文件主同意的用户访问。

1) 存取控制矩阵

理论上存取控制方法可采用存取控制矩阵实现，它是一个二维矩阵，一维列出计算机的全部用户，另一维列出系统中的全部文件，矩阵中每个元素A*ij*是表示第*i*个用户对第*j*个文件的存取权限。通常存取权限有可读、可写、可执行以及它们的组合。

2) 存取控制表

存取控制表是按用户对文件的访问权限的差别对用户进行分类，由于某一文件往往只与少数几个用户有关，所以这种分类方法可使存取控制表大为简化。UNIX系统使用的就是这种存取控制表方法。它把用户分成3类，包括文件主、同组用户和其他用户，每类用户的存取权限为可读、可写、可执行的组合。

3) 用户权限表

用户权限表是以用户或用户组为单位将用户可存取的文件集中起来存入表中，表中每个表目表示该用户对相应文件的存取权限，这相当于存取控制矩阵一行的简化。

4) 密码

在创建文件时，由用户提供一个密码，在文件存入磁盘时用该密码对文件内容进行加密。进行读取操作时，要对文件进行解密，只有知道密码的用户才能读取文件。

#### 七、系统的安全与可靠性

###### 1. 系统的安全

系统的安全涉及两类问题：一类涉及技术、管理、法律、道德和政治等问题；另一类涉及操作系统的安全机制。

一般从4个级别上对文件进行安全性管理：系统级、用户级、目录级和文件级。

文件级安全管理是通过系统管理员或文件主对文件属性的设置来控制用户对文件的访问。通常可设置的属性有：只执行、隐含、只读、读写、共享、系统。用户对文件的访问由用户访问权、目录访问权和文件属性三者的权限决定。

###### 2. 文件系统的可靠性

文件系统的可靠性是指系统抵抗和预防各种物理性破坏和人为性破坏的能力。

1) 转储和恢复

常用的转储方法有：静态转储和动态转储、海量转储和增量转储。

2) 日志文件

在计算机系统工作的过程中，操作系统把用户对文件的插入、删除和修改的操作写入日志文件。一旦发生故障，可利用日志文件来进行系统故障的恢复。

3) 文件系统的一致性

影响文件系统可靠性的因素之一是文件系统的一致性问题。通常解决方案是采用文件系统的一致性检查，包括块的一致性检查和文件的一致性检查。

### 2.2.6 作业管理



#### 一、作业管理概述

作业是系统为完成一个用户的计算任务(或一次事务处理)所做的工作总和。

###### 1. 作业控制

用户可以采用脱机和联机两种控制方式来控制作业的运行。在脱机控制方式中，作业运行的过程是无须人工干预的，因此用户必须编写成作业说明书连同作业一起提交给计算机系统。在联机控制方式中，操作系统向用户提供了一组联机命令，用户可以通过终端输入命令，将自己的意图告诉计算机，以控制作业的运行过程。

作业由程序、数据和作业说明书3部分组成。作业说明书包括作业基本情况、作业控制、作业资源要求的描述。其中：作业基本情况包括用户名、作业名、编程语言、最大处理时间等；作业控制描述包括作业控制方式、作业步的操作顺序、作业执行的出错处理；作业资源要求描述包括处理时间、优先级、主存空间、外设类型和数量、实用程序要求等。

###### 2. 作业状态

作业的状态分为4种：提交、后备、执行和完成。

* 提交：作业提交给计算机中心，通过输入设备送入计算机系统的过程所处的状态称之为提交状态。
* 后备：作业通过spooling系统输入到计算机系统的后备存储器(磁盘)中，随时等待作业调度程序调度时的状态。
* 执行：一旦作业被作业调度程序选中，为其分配了必要的资源，并为其建立相应的进程后，该作业便进入了执行状态。
* 完成：当作业正常结束或异常终止时，便进入完成状态。此时由作业调度程序对该作业进行善后处理。如撤销作业的作业控制块，收回作业所占的系统资源，将作业的执行结果形成输出文件放到输出井中，由spooling系统控制输出。

###### 3. 作业控制块(JCB)和作业后备队列

所谓作业控制块是记录与该作业有关的各种信息的登记表。作业控制块是作业存在的唯一标志。通常将作业控制块排成一个或多个队列，而这些队列通常称为作业后备队列，即作业后备队列是由若干个作业控制块组成的。

#### 二、作业调度

作业调度主要是从后备状态的作业中挑选一个(或一些)作业投入运行。根据不同的调度目标，有不同的调度算法。

###### 1. 调度算法的选择

调度算法的选择主要有以下5点原则：响应时间快；周转时间或加权周转时间短；均衡的资源利用率；吞吐量大；系统反应时间短。

###### 2. 调度算法

作业调度算法有许多种，常见的有如下几种。

* 先来先服务(FCFS)：按作业到达的先后次序调度，它不利于短作业。
* 短作业优先(SJF)：按作业的估计运行时间调度，估计运行时间短的作业优先调度，它不利于长作业，可能会使一个估计运行时间长的作业迟迟得不到服务。
* 响应比高者优先(HRN)：综合上述两者，既考虑作业估计运行时间，又考虑作业等待时间，响应比是：

HRN=(估计运行时间+等待时间)/估计运行时间

* 优先级调度算法：根据作业的优先级别，优先级高者先调度。
* 均衡调度算法：根据系统运行情况和作业本身进行分类，调度程序从这些分类中轮流挑选作业执行。

###### 3. 作业调度算法性能的衡量指标

在一个以批量处理为主的系统中，通常用平均周转时间或平均周转系数来衡量调度性能的优劣。假设作业*Ji*(*i*＝1，2，…，*n*)的提交时间为*tsi*，执行时间为*tri*，作业完成时间为*toi*，则作业*Ji*的周转时间*Ti*和周转系数*Wi*分别定义为：

*Ti*＝*toi*－*tsi* (*i*＝1，2，…，*n*)

*Wi*＝*Ti/tri* (*i*＝1，2，…*，n*)

*n*个作业的平均周转时间*r*和平均周转系数*W*分别定义为：

*T*＝， *W*＝

从用户的角度来说，总是希望自己的作业在提交后能立即执行，这就意味着当等待时间为0时作业的周转时间最短，即*Ti＝tri*。但是作业的执行时间*tri*并不能直观地衡量出系统的性能，而周转系数*Wi*却能直观地反应系统的调度性能。从整个系统的角度来说，不可能满足每个用户的这种要求，而只能是系统的平均周转时间或平均周转系数最小。

#### 三、人机界面

人机界面是计算机中实现用户与计算机通信的软件和硬件部分的总称。人机界面也称为用户接口或用户界面。从计算机用户界面的发展过程来看，人机界面的发展可分为以下4个阶段。

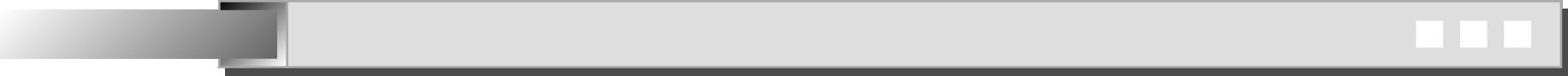
(1) 控制面板式用户界面。

(2) 字符用户界面。

(3) 图形用户界面。

(4) 新一代用户界面。

## 2.3 真题详解



### 综合知识试题

试题1(2010年下半年试题25)

在操作系统文件管理中，通常采用 (1) 来组织和管理外存中的信息。

(1) A. 字处理程序　　B. 设备驱动程序　　C. 文件目录　　D. 语言翻译程序

**参考答案：**(1)C。

**要点解析：操作系统中的文件系统专门负责管理外存储器上的信息，使用户可以“按名”高效、快速和方便地存储信息。为了实现“按名存取”，系统必须为每个文件设置用于描述和控制文件的数据结构，它至少要包括文件名和存放文件的物理地址，这个数据结构称为文件控制块，文件控制块的有序集合称为文件目录。换句话说，文件目录是由文件控制块组成的，专门用于文件检索。文件控制块也称为文件的说明或文件目录项(简称目录项)。**

试题2和试题3(2010年下半年试题26和试题27)

假设系统中进程的三态模型如图2-3所示，图中的a, b和c的状态分别为 (2) ；当现运行进程执行P操作后，该进程 (3) 。



##### 图2-3 进程的三态模型

(2) A. 就绪、运行、阻塞　　B. 运行、阻塞、就绪

C. 就绪、阻塞、运行　　D. 阻塞、就绪、运行

(3) A. 处于运行状态　　　　B. 处于阻塞状态

C. 处于就绪状态　　　　D. 处于运行状态或者进入阻塞状态

**参考答案：(2)A；(3)D。**

**要点解析：试题(2)选A。**在多道程序系统中，进程的运行是走走停停，在处理器上交替运行，状态也不断地发生变化，因此进程一般有三种基本状态：运行、就绪和阻塞，也称为三态模型，如图2-4所示。



##### 图2-4 进程的三种基本状态

①运行：当一个进程在处理机上运行时，称该进程处于运行状态。显然，对于单处理机系统，处于运行状态的进程只有一个。②就绪：一个进程获得了除处理机外的一切所需资源，一旦得到处理机即可运行，则称此进程处于就绪状态。③阻塞：也称等待或睡眠状态，一个进程正在等待某一事件发生(例如，请求I/O而等待I/O完成等)而暂时停止运行，这时即使把处理机分配给该进程，它也无法运行，故该进程处于阻塞状态。

试题(3)选D。PV操作是实现进程同步与互斥的常用方法。P操作和V操作是低级通信原语，在执行期间不可分割。其中，P操作表示申请一个资源，V操作表示释放一个资源。P操作的定义：S：S-1，若S≥0，则执行P操作的进程继续执行；若S＜0，则置该进程为阻塞状态(因为无可用资源)，并将其插入阻塞队列。题中，将现在运行进程执行P操作，即将b进程执行P操作，此时，如若系统中还存在剩余资源空间，则b进程继续处于运行状态，若系统中没有剩余资源空间，则b进程进入阻塞状态。

试题4和试题5(2010年上半年试题23和试题24)

Windows系统中的磁盘碎片整理程序 (4) ，这样使系统 (5) 。

(4) A. 仅将卷上的可用空间合并，使其成为连续的区域

B. 只能使每个文件占用卷上连续的磁盘空间，合并卷上的可用空间

C. 只能使每个文件夹占用卷上连续的磁盘空间，合并卷上的可用空间

D. 使每个文件和文件夹占用卷上连续的磁盘空间，合并卷上的可用空间

(5) A. 对文件能更有效地访问，而对文件夹的访问效率保持不变

B.  对文件夹能更有效地访问，而对文件的访问效率保持不变

C.  对文件和文件夹能更有效地访问

D. 将磁盘空闲区的管理方法改变为空白文件管理方案

**参考答案：**(4) D；(5) C。

**要点解析：试题(4)选D，试题(5)选C。因为，在Windows系统中的磁盘碎片整理程序可以分析本地卷，使每个文件或文件夹占用卷上连续的读磁盘空间，合并卷上的可用空间使其成为连续的空闲区域，这样系统就可以更有效地访问文件或文件夹，以及更有效地保存新的文件和文件夹。通过合并文件和文件夹，磁盘碎片整理程序还将合并卷上的可用空间，以减少新文件出现碎片的可能性。合并文件和文件夹碎片的过程称为碎片整理。**

试题6(2010年上半年试题25)

在磁盘移臂调度算法中， (6) 算法可能会随时改变移动臂的运动方向。

(6) A. 电梯调度和先来先服务           B. 先来先服务和单向扫描

C. 电梯调度和最短寻道时间优先   D. 先来先服务和最短寻道时间优先

**参考答案：(6)**D。

**要点解析：在操作系统中常用的磁盘调度算法有：先来先服务、最短寻道时间优先、扫描算法、循环扫描算法等。其中，先来先服务是最简单的磁盘调度算法，它根据进程请求访问磁盘的先后次序进行调度，所以该算法可能会随时改变移动臂的运动方向。最短寻道时间优先算法根据进程请求，访问磁盘的寻道距离短的优先调度，因此该算法可能会随时改变移动臂的运动方向。电梯调度法的工作原理是先响应同方向(向内道或向外道方向)的请求访问，然后再响应反方向的请求访问，如同电梯的工作原理一样，因此该算法可能会随时改变移动臂的运动方向。单项扫描算法是电梯调度法的改进，该算法在返程时不响应请求访问，目的是为了解决电梯调度法带来的饥饿问题。**

试题7和试题8(2010年上半年试题26和试题27)

若一个单处理器的计算机系统中同时存在3个并发进程，则同一时刻允许占用处理器的进程数 (7) ；如果这3个进程都要求使用2个互斥资源R，那么系统不产生死锁的最少的R 资源数为 (8) 个。

(7) A. 至少为 1 个    B. 至少为3个    C. 最多为1个    D. 最多为3个

(8) A. 3              B. 4            C. 5            D. 6

**参考答案：**(7) C；(8) B。

**要点解析：试题(7)选C，因为一个单处理器的计算机系统中尽管同时存在3个并发进程，但是同一时刻允许占用处理器的进程数只能是1个。引入多道程序设计的目的是为了提高资源的利用率，例如，进程P1要输入/输出时，可以将CPU分配给进程P2，使得进程P1的输入/输出和进程P2的程序执行并发运行。**

**试题(8)选B，对于选项A，在操作系统为每个进程分配1个资源R后，若这3个进程再分别请求1个资源R时，系统已无可供分配的资源R，则这3个进程由于请求的资源R得不到满足而死锁。如果选择选项B，那么操作系统为每个进程分配1个资源R后，系统还有1个可供分配的资源R，能满足其中的1个进程的资源要求，它运行完毕释放占有的资源R后，可以使其他进程也能得到所需的资源R并运行完毕。**

试题9(2010年上半年试题30)

某有限状态自动机的状态图如图2-5所示(状态0是初态，状态2是终态)，则该自动机不能识别 (9) 。



##### 图2-5 某有限状态自动机的状态图

(9) A. abab       B. aabb          C. bbaa         D. bbab

**参考答案：(9)**A。

**要点解析：对于选项A，从状态图的状态0出发，识别a后到达状态0，识别b后到达状态1，由于不存在从状态1出发识别a的状态转移，因此，abab不能被该自动机识别。对于选项B，识别aabb的状态转移路是状态0→状态0→状态1→状态2。对于选项C，识别bbaa的状态转移路是状态0→状态1→状态2→状态2→状态2。对于选项D，识别bbab的状态转移路是状态0→状态1→状态2→状态2→状态2。**

试题10和试题11(2009年下半年试题26和试题27)

在操作系统的进程管理中，若系统中有 10 个进程使用互斥资源 R ，每次只允许 3 个进程进入互斥段(临界区)，则信号量 S 的变化范围是 (10) ；若信号量 S 的当前值为-2，则表示系统口有 (11) 一个正在等待该资源的进程。

(10) A. -7～1 B. -7～3 C. -3～0 D. -3～10

(11) A. 0 B. 1 C. 2 D. 3

**参考答案：**(10)B；(11)C。

**要点解析：**本题考查的是进程和信号量的知识。信号量S的物理意义：S≥0表示某资源的可用数；若S<0，则其绝对值表示阻塞队列中等待该资源的进程数。在操作系统的进程管理中，若系统中有10 个进程使用互斥资源R，每次只允许3个进程进入互斥段(临界区)，则信号量 S 的变化范围是-7～3；若信号量S的当前值为-2，则表示系统中有2个正在等待该资源的进程。

试题12(2009年上半年试题24)

打印质量好、速度快，且工作原理与复印机相似的打印机是 (12) 。

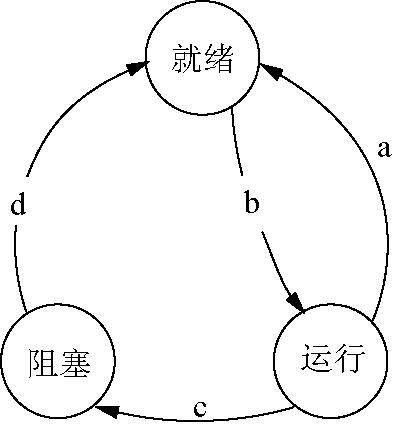
(12) A. 行式打印机 B. 激光打印机 C. 喷墨打印机 D. 点阵打印机

**参考答案：(12)B。**

**要点解析：**打印质量好、速度快，且工作原理与复印机相似的打印机是激光打印机。

试题13和试题14(2009年上半年试题26和试题27)

假设系统中进程的三态模型如图2-6所示，图中a和b处应分别填写 (13) ，图中的c和d处应分别填写 (14) 。



##### 图2-6 进程三态模型图

(13) A. 时间片到、作业调度程序调度 B. 执行了P操作、时间片到

C. 时间片到、进程调度程序调度 D. 时间片到，执行了P操作

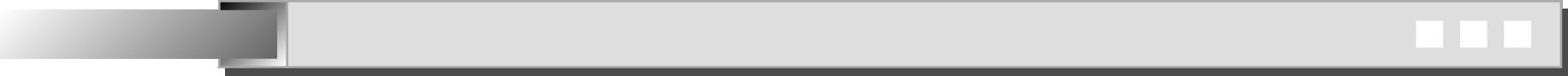
(14) A. I/O完成、执行了P操作 B. 执行了P操作、I/O完成

C. 时间片到、执行了V操作 D. 执行了V操作、I/O完成

**参考答案：(13)C；(14)B。**

**要点解析：**由系统进程的三态模型可知，a处应为时间片到，b处应为进程调度程序调度；c处应为执行了P操作，d处应为I/O完成。

## 2.4 强化训练



### 2.4.1 综合知识试题

试题1

计算机加电自检以后，引导程序首先装入 (1) ，否则，计算机不能做任何事情。

(1) A. 操作系统 B. 编译程序 C. Office系列软件 D. 应用软件

试题2

操作系统的任务是 (2) 。

(2) A. 把源程序转换为目标代码

B. 管理计算机系统中的软、硬件资源

C. 负责存取数据库中的各种数据

D. 负责文字格式编排和数据计算

试题3和试题4

已知有 6 个进程共享一个互斥段，如果最多允许 3 个进程同时进入互斥段，则信号量S 的变化范围是 (3) ；若信号量S 的当前值为-2，则表示系统中有 (4) 个正在等待该资源的进程。

(3) A. -5～1 B. -3～3 C. -2～4 D. -2～5

(4) A. 0 B. 1 C. 2 D. 3

试题5

若进程P1正在运行，操作系统强行撤下P1所占用的CPU，让具有更高优先级的进程P2运行，此时P1进程处于 (5) 状态。

(5) A. 就绪 B. 等待 C. 结束 D. 善后处理

试题6和试题7

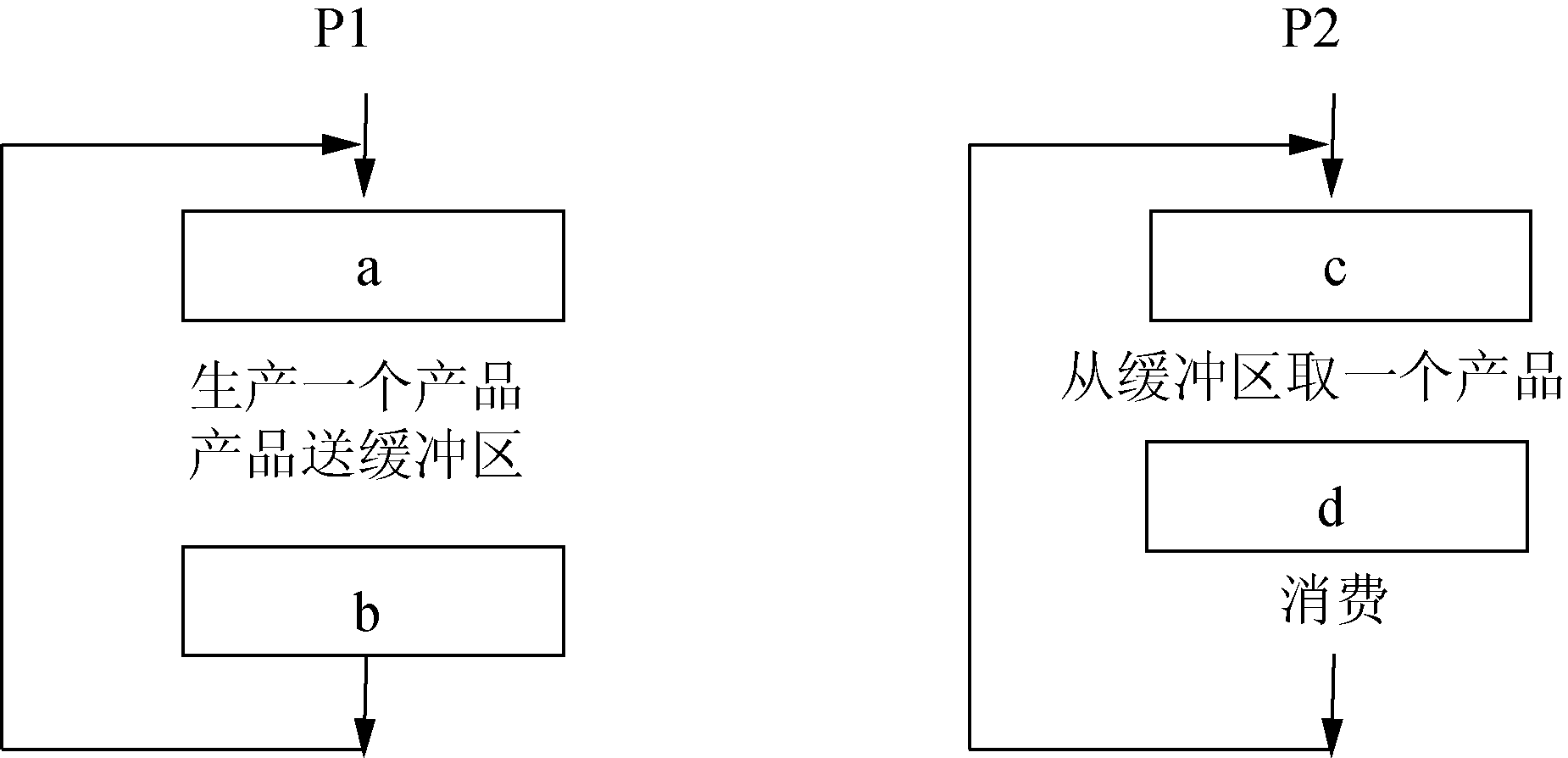
已知有10个进程共享一个互斥段，如果最多允许6个进程同时进入互斥段，则信号量S的初值为 (6) ，信号量S的变化范围是 (7) 。

(6) A. 0 B. 1 C. 5 D. 6

(7) A. 0～10 B. -1～9 C. -4～6 D. –5～5

试题8和试题9

某系统中有一个缓冲区，进程 P1 不断地生产产品送入缓冲区，进程 P2 不断地从缓冲区中取出产品消费，用 P、V 操作实现进程间的同步模型如图2-7所示。假设信号量 S1 的初值为 1，信号量 S2 的初值为 0，那么 a、b 处应分别填 (8) ，c、d 处分别填 (9) 。



##### 图2-7 进程间的同步模型图

(8) A. V(S2)、P(S1) B. P(S1)、V(S2) C. P(S2)、V(S1) D. V(S1)、P(S2)

(9) A. V(S2)、P(S1) B. P(S1)、V(S2) C. P(S2)、V(S1) D. V(S1)、P(S2)

试题10

产生中断时，由硬件保护并更新程序计数器PC的内容，其主要目的是 (10) 。

(10) A. 节省内存空间并提高内存空间的利用率

B. 提高中断处理程序的运行速度

C. 简化中断处理程序的编写过程

D. 快速进入终端处理程序并正确返回被中断的程序

试题11和试题12

若分页系统地址的结构如图2-8所示，该系统页的大小为 (11) 字节，页号的取值范围为 (12) 。

(11) A. 1024 B. 2048 C. 4096 D. 8192

(12) A. 0～255 B. 1～256 C. 0～511 D. 1～512

    19 12 11 0

|  |  |
| --- | --- |
| 页号 | 页内地址 |

##### 图2-8 分页系统地址结构

试题13

某页式存储管理系统中的地址结构如图2-9所示，则 (13) 。

(13) A. 页的大小为1KB，最多有8M 页

B. 页的大小为2KB，最多有4M 页

C. 页的大小为4KB，最多有1M 页

D. 页的大小为8KB，最多有2M 页



##### 图2-9 页式存储管理系统的地址结构

试题14

在计算机系统中常用的输入/输出方式有无条件传送、中断、程序查询和DMA等，采用　(14)　方式，数据在传输过程中不需要 CPU 参与工作。

(14) A. 中断　　　 B. DMA C. 程序查询 D. 无条件传送

试题15

对于一个基于网络的应用系统，在客户端持续地向服务端提交作业请求的过程中，若作业响应时间越短，则服务端 (15) 。

(15) A. 占用内存越大 B. 越可靠

C. 吞吐量越大 D. 抗病毒能力越强

### 2.4.2 综合知识试题参考答案



【试题1】答案：(1)A。

解析：本题考查的是操作系统的基本知识。操作系统位于硬件之上，所有其他软件之下，是其他软件的共同环境与平台。计算机加电以后，首先引导操作系统。不引导操作系统，计算机不能做任何事。

【试题2】答案：(2)B。

解析：操作系统的任务是管理计算机系统中的软、硬件资源；编译或汇编程序把源程序转换为目标代码；数据库管理系统负责存取数据库中的各种数据；文字处理软件和计算软件负责文字的格式编排和数据计算。

【试题3和试题4】答案：(3)B；(4)C。

解析：最多允许 3 个进程同时进入互斥段，所以信号量S初值应为3，当6个进程依次进入互斥段时，S逐渐递减，最小为3-6=-3，所以变化范围为-3～3。如果S为-2，表示系统中有2个正在等待该资源的进程。

【试题5】答案：(5)A。

解析：进程一般有三种基本状态：运行、就绪和阻塞。其中，就绪状态表示一个进程获得了除处理机外的一切所需资源，一旦得到处理机即可运行，这里称此进程处于就绪状态，若进程P1正在运行，操作系统强行撤下P1进程所占用的CPU，让具有更高优先级的进程P2运行，此时P1进程处于就绪状态。

【试题6和试题7】答案：(6)D；(7)C。

解析：本题中已知有10个进程共享一个互斥段，如果最多允许6个进程同时进入互斥段，说明系统有6个单位的资源，信号量S的初值应设为6。当第一个申请该资源的进程对信号量S执行P操作，信号量S减1后等于5，进程可继续执行；当第二个申请该资源的进程对信号量S执行P操作，信号量S再减1后等于4，进程可继续执行；……当第10个申请该资源的进程对信号量S执行P操作，信号量S减1等于-4。所以信号量S的变化范围是-4～6。

【试题8和试题9】答案：(8)B；(9)C。

解析：依题意可知，如果P1、P2能正常运行，a、b、c、d四处应该分别为P(S1)、V(S2)、P(S2)、V(S1)。

【试题10】答案：(10)D。

解析：本题考查的是操作系统基础，如果产生中断，由硬件保护并更新程序计数器PC的内容，那么其主要目的是快速进入终端处理程序并正确返回被中断的程序。

【试题11和试题12】答案：(11)C；(12)A。

解析：图中页内地址的长度为12位，每页大小应为212=4096字节；页号的长度为8位，取值范围为0～255。

【试题13】答案：(13)C。

解析：页内地址的长度是12位，=4096，即4K；页号部分的地址长度是20位，=1048576，即1M。

【试题14】答案：(14)B。

解析：由于DMA方式是在DMA控制器硬件的控制下实现数据的传输，在传输过程中不需要 CPU 参与工作，其速度也最快。

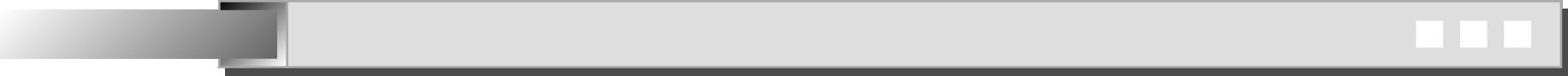
【试题15】答案：(15)C。

解析：衡量系统效率的常用指标包括响应时间、吞吐量和周转时间等，其中作业的响应时间会直接影响系统吞吐量，作业响应时间越短，系统的吞吐量越大。

# 第3章

# 数据库基础知识

## 3.1 备考指南



### 3.1.1 考纲要求

根据考试大纲中相应的考核要求，在“数据库基础知识”模块上，要求考生掌握以下方面的内容。

(1) 数据库管理系统的主要功能和特征。

(2) 数据库模型(概念模式、外模式、内模式)。

(3) 数据模型、E-R图。

(4) 数据操作(关系运算)。

(5) 数据库语言(SQL)。

(6) 数据库的主要控制功能(并发控制、安全控制)。

### 3.1.2 考点统计



“数据库基础知识”模块，在历次程序员考试试卷中出现的考核知识点及分值分布情况如表3-1所示。

##### 表3-1 历年考点统计表

| 年 份 | 题 号 | 知 识 点 | 分 值 |
| --- | --- | --- | --- |
| 2009年  下半年 | 上午题：58～62 | 数据库管理系统的主要功能和特征、数据操作(关系运算) | 5分 |
| 下午题：无 |  | 0分 |

续表

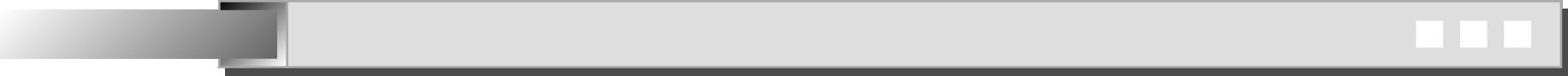
| 年 份 | 题 号 | 知 识 点 | 分 值 |
| --- | --- | --- | --- |
| 2009年  上半年 | 上午题：57～62 | E-R图、关系运算、数据库语言(SQL) | 6分 |
| 下午题：无 |  | 0分 |
| 2010年  下半年 | 上午题：57～62 | E-R图、数据库语言SQL、数据库管理系统的主要功能及特征 | 6分 |
| 下午题：无 |  | 0分 |
| 2010 年  上半年 | 上午题：56～62、64 | 数据处理、数据库管理、数据库模型等 | 8分 |
| 下午题：无 |  | 0分 |

### 3.1.3 命题特点



纵观历年试卷，本章知识点是以选择题的形式出现在试卷中的。本章知识点在历次考试“综合知识”试卷中，所考查的题量大约为5～8道选择题，所占分值为5～8分(约占试卷总分值75分中的6.67%～10.67%)。大多数试题偏重于实践应用，检验考生是否理解相关的理论知识点和实践经验，考试难度中等。

## 3.2 考点串讲



### 3.2.1 基本概念

#### 一、数据库与数据库系统

**数据库系统(DBS)是由数据库、硬件、软件和人员组成的。**

**(1) 数据库(Database，缩写为DB)是指长期保存在计算机上，并按照某种模型组织起来的，可共享的数据集合。**

(2) 硬件：构成计算机系统的各种物理设备。

(3) 软件：包括操作系统、数据库管理系统及应用程序。

(4) 人员：主要有4类。第一类是系统分析员和数据库设计人员；第二类为应用程序员；第三类为最终用户；第四类是数据库管理员。

#### 二、数据库管理技术的发展

数据库管理技术是计算机软件领域的一个重要分支，产生于20世纪60年代末，其发展大致经过以下3个阶段。

###### 1. 人工管理阶段(20世纪50年代中期以前)

人工管理阶段的计算机主要用于科技计算。外存只有磁带、卡片和纸带等，软件只有汇编语言，尚无数据管理方面的软件。数据处理方式是批处理。

###### 2. 文件系统阶段(20世纪50年代后期至60年代中期)

文件系统阶段的计算机不仅用于科技计算，还用于信息管理。外存已有了磁盘、磁鼓等直接存取的存储设备。软件中已有专门管理数据存储的软件——文件系统，它是操作系统的一部分，有时也称为“信息处理模块”。数据处理方式有批处理，也有联机实时处理。

###### 3. 数据库系统阶段(20世纪60年代末开始)

数据库系统阶段数据管理的特点如下。

(1) 采用复杂的数据模型表示数据结构。数据不再面向某个应用，而是面向整个应用系统。数据冗余明显减少，实现了数据共享。

(2) 有较高的数据独立性。数据库系统和文件系统的区别是：①数据库对数据的存储是按照同一结构进行的，不同程序都可以直接操作这些数据(即对应用程序的高度独立性)；②数据库系统对数据的完整性、唯一性和安全性都有一套有效的管理手段(即数据的充分共享性)；③数据库系统还提供管理和控制数据的各种简单操作命令，以方便用户操作(即操作方便性)。

### 3.2.2 数据模型



#### 一、数据模型的基本概念

###### 1. 数据的描述

在数据处理中，涉及不同的数据描述领域。从事物的特性到计算机里的具体表示，经历了3个数据领域——现实世界、信息世界和机器世界。

1) 现实世界

现实世界(real world)的数据就是客观存在的各种报表、图表和查询格式等原始数据。

2) 信息世界

信息世界(information world)是现实世界在人们头脑中的反映，人们把它用文字和符号记载下来。在信息世界中，数据库技术用到下列术语。

* 实体(entity)：客观存在并且可以相互区别的东西称为实体。如一个女学生、一辆汽车等。也可以是抽象的事件，如一次篮球比赛、一次上网等。
* 实体集(entity set)：性质相同的同类实体集合，称为实体集。如所有的男学生、全国篮球锦标赛的所有比赛等。
* 属性(attribute)：实体有若干特性，每一个特性称为一个属性。每个属性有一个值域，其类型可以是整型、实型或字符型。如学生有姓名、年龄、性别等属性，相应值域的类型分别是字符串、整数和字符型。
* 键(key)：实体中凡能唯一标识实体集中每个实体的属性或属性集就称为实体的键，有时也称为实体标识符。例如，学生的姓名(不允许重名)可以作为学生实体的键。

3) 机器世界

信息世界的信息在机器世界中以数据形式进行存储。机器世界中数据描述的术语有以下4个。

* 字段(field)：标记实体属性的符号集称为字段或数据项。它是可以命名的最小数据单位。字段的命名往往与属性名相同。
* 记录(record)：字段的有序集合称为记录。一般来说，用一个记录描述一个实体。
* 文件(file)：同一类记录的汇集称为文件。文件是描述实体集的，所以它又可以定义为描述一个实体集的所有符号集。
* 键(key)：能唯一标识文件中每个记录的字段或字段集，称为文件的键或记录的键。

###### 2. 数据模型的定义

模型是对现实世界的抽象。数据库技术中用模型的概念描述数据库的结构与语义，对现实世界进行抽象。数据库中的数据是有结构的，这种结构反映出事物和事物之间的联系，是按照某种数据模型来组织数据的。

#### 二、数据模型的三要素

数据库的数据模型的三要素是：数据结构、数据操作和数据的约束条件。

* 数据结构：是所研究的对象类型的集合。
* 数据操作：是指对数据库中的各种对象的实例(值)允许执行的操作的集合。其主要用于描述系统的动态特性。
* 数据的约束条件：是一组完整性规则的集合。它给出数据及其联系所具有的制约和依赖规则。这些规则用于限定数据库的状态及状态的变化，以保证数据库中数据的正确性、有效性和相容性。

目前常用的数据模型有两种类型：概念数据模型和基本数据模型。

#### 三、E-R模型

概念数据模型中最著名的模型是“实体联系模型”(Entity-Relationship Model，E-R模型)。这个模型直接从现实世界中抽象出实体类型及实体间的联系，然后用实体联系图(E-R图)表示数据模型。在E-R图中，用矩形框表示实体类型，用菱形框表示实体间的联系类型，用椭圆表示实体或联系的属性，实体间联系用箭头标出并注上联系的种类。

E-R图的3个要素是：实体、属性和实体之间的联系。数据库设计的第一步就是要使用E-R图描述数据组织模式，然后进一步转换成任意一种DBMS支持的数据类型。

###### 1. 实体

实体是现实世界中可以区别于其他对象的“事件”或“物体”。

###### 2. 属性

E-R模型中的属性主要有：简单属性和复合属性；单值属性和多值属性；NULL属性；派生属性。

###### 3. 联系

实体的联系有两类：一类是实体内部的联系，反映在数据上是同一记录内部各字段间的联系；另一类是实体与实体之间的联系，反映在数据上就是记录之间的联系。

两个不同实体集的实体间的联系主要有以下3种情况：一对一联系(1∶1)、一对多联系(1∶M)、多对多联系(M∶N)。

上面3种联系是实体之间最基本的联系，类似地，也可以定义为多个实体集(3个或3个以上)之间的各种联系，或定义同一个实体集的实体间联系。

E-R模型建立的一般步骤如下。

(1) 确定实体类型。

(2) 确定实体间联系的类型。

(3) 根据实体类型和联系类型画出E-R图。

(4) 确定实体类型和联系类型的属性。

#### 四、基本数据模型

基本数据模型主要有层次模型、网状模型、关系模型三种。

###### 1. 层次模型

用树型结构表示实体类型及实体之间联系的数据模型称为层次模型(hierarchical model)。层次结构是一棵树，树的节点是记录类型，非根节点有且只有一个父节点。上一层记录类型和下一层记录类型的联系是1∶M联系(包括1∶1联系)。

层次模型的特点是：记录之间的联系通过指针实现，实现容易，且查询效率较高。

层次模型的缺点是：只能表示1∶M联系，虽有多种辅助手段实现M∶N联系，但较复杂，不易掌握；由于层次顺序的严格和复杂，引起数据的查询、插入、删除也较复杂，因此应用程序的编写比较复杂。

###### 2. 网状模型

用网络结构表示实体类型及实体间联系的数据模型称为网状模型(network model)。网状模型的数据结构是有向图结构。有向图中的节点是记录类型，箭头表示记录间的1∶M联系。

网状模型的特点是：记录之间的联系通过指针实现，M∶N联系容易实现，且查询效率较高。

网状模型的缺点是：结构复杂，程序员必须熟悉数据库的逻辑结构。

###### 3. 关系模型

关系模型(relational model)是目前最常用的数据模型之一。其主要特征是：用表格结构表达实体集，用键表示实体间的联系。不仅可用关系描述实体本身，还可用关系描述实体之间的联系。可直接表示多对多的联系，每个属性不可再分，建立在数学概念基础上，有较强的理论依据。与前两种模型相比，关系模型比较简单，容易为初学者接受。关系模型是由若干个关系模式组成的集合，关系模式相当于前面提到的记录类型，它的实例称为关系。每个关系实际上是一张表格。

关系模型和网状模型、层次模型的最大差别是：用关键码(主码)而不是用指针导航数据，表格简单、易懂。典型的关系数据库管理系统产品有DB2、Oracle、Sybase、SQL Server、Informix，以及微机型产品的dBASE、FoxBASE、FoxPro等。

### 3.2.3 数据库管理系统的功能和特征



#### 一、数据库管理系统的功能

数据库管理系统(DBMS)主要实现对共享数据有效的组织、管理和存取。

###### 1. 数据定义

DBMS提供数据定义语言(Data Definition Language，DDL)，用户可以对数据库的结构进行描述，包括外模式、模式和内模式的定义；数据库的完整性定义；安全保密定义，如口令、级别、存取权限等。这些定义存储在数据字典中，是DBMS运行的基本依据。

###### 2. 数据库操作

DBMS向用户提供数据操纵语言(Data Manipulation Language，DML)，实现对数据库中数据的基本操作，如检索、插入、修改和删除等。DML分为宿主型和自含型。

* 宿主型是指将DML语句嵌入某种主语言(如C、COBOL等)中使用。
* 自含型是指可以单独使用的DML语句，供用户交互使用。

###### 3. 数据库运行管理

DBMS的数据库运行管理功能包括在数据库运行期间对多用户环境下的并发控制、安全性检查和存取控制、完整性检查和执行、运行日志的组织管理、事务管理和自动恢复等。这些功能可以保证数据库系统的正常运行。

###### 4. 数据组织、存储和管理

DBMS分类组织、存储和管理各种数据，包括数据字典、用户数据、存取路径等。DBMS要确定以何种文件结构和存取方式在存储级别上组织这些数据，以提高存取效率。实现数据间的联系、数据组织和存储的基本目标是提高存储空间的利用率。

###### 5. 数据库的建立和维护

数据库的建立和维护包括数据库的初始建立、数据的转换、数据库的转储和恢复、数据库的重组和重构、性能监测和分析等。

###### 6. 其他功能

DBMS的其他功能包括DBMS与网络中其他软件系统的通信功能，一个DBMS与另一个DBMS或文件系统的数据转换功能等。

上面所有的功能是一般的DBMS所具备的功能，通常在大、中型机上实现的DBMS功能较强、较全，在微机上实现的DBMS功能较弱。

注意：应用程序并不属于DBMS的范围。因为应用程序是用主语言和DML编写的。程序中的DML语句由DBMS执行，而其余部分仍由主语言的编译程序完成。

#### 二、DBMS的特征

###### 1. DBMS特征

1) 数据结构化且统一管理

数据库中的数据由DBMS统一管理。数据模型不仅描述数据本身，还描述数据之间的联系。数据易维护、易扩展，冗余明显减少，实现了数据共享。

2) 有较高的数据独立性

数据的独立性是指数据与程序独立，将数据的定义从程序中分离出去。数据的独立性包括数据的物理独立性和数据的逻辑独立性。

3) 数据控制功能

数据控制功能包括对数据库中数据的安全性、完整性、并发和恢复的控制。

* 数据库的安全性：是指防止不合法的使用所造成的数据泄露、更改或破坏。
* 数据库的完整性：是指数据库的正确性和相容性，是防止合法用户使用数据库时向数据库加入不符合语义的数据。
* 并发控制：并发操作带来的问题是数据的不一致性，主要有丢失更新、不可重复读和读脏数据3类。DBMS负责协调并发事务的执行，保证数据库的完整性不受破坏，避免用户得到不正确的数据。
* 故障恢复：数据库中的4类故障分别是事务内部故障、系统故障、介质故障及计算机病毒。故障恢复主要是指恢复数据库本身，恢复的原理非常简单，就是要建立冗余数据。

###### 2. RDBS、OODBS和ORDBS

1) 关系数据库系统(relation database system，RDBS)

RDBS是支持关系模型的数据库系统。在关系模型中，实体以及实体间的联系都用关系来表示。关系数据库的型也称为关系数据库模式，是对关系数据库的描述，是关系模式的集合。关系数据库的值也称为关系数据库，是关系的集合。

2) 面向对象的数据库系统(object-oriented database system，OODBS)

OODBS是支持以对象形式进行数据建模的数据库管理系统。它支持对象的类，类属性的继承，子类。

3) 对象关系数据库系统(Object-oriented Relation Database System，ORDBS)

ORDBS在传统的关系数据模型基础上，提供元组、数组、集合之类更为丰富的数据类型以及处理新的数据类型操作的能力，这样形成的数据模型称为“对象关系数据模型”，基于对象关系数据模型的DBS称为对象关系数据库系统。

### 3.2.4 数据库模式



#### 一、数据库的三级模式结构

1975年2月，美国国家标准化委员会(ANSI)提出：数据库的数据体系结构分成内部级、概念级和外部级三个级别。这三个结构之间往往差别很大。为实现这三个抽象级别的转换，DBMS在这三级之间提供了两层映射：外模式/概念模式映射和概念模式/内模式映射。

###### 1. 概念模式

概念模式(也称为模式)是数据库中全部数据的逻辑结构和特征的描述，由若干个概念记录类型组成。概念模式的一个具体值称为模式的一个实例。概念模式不但要描述概念记录类型，而且要描述记录之间的联系、所允许的操作、数据的一致性、安全性和其他数据控制方面的要求。

在概念模式中必须不涉及存储结构、访问技术等细节。只有这样，概念模式才算做到了数据独立性，而在概念模式基础上定义的外模式才能做到数据独立。

描述概念模式的数据定义语言称为“模式DDL”。

###### 2. 外模式

外模式是用户与数据库系统的接口。外模式是用户用到的那部分数据的描述，它由若干个外部记录类型组成。用户使用数据操纵语言(DML)对数据库进行操作，实际上是对外模式的外部记录进行操作。例如，读一个记录值，实际上是读一个外部记录值(即逻辑值)，而不是读数据库的内部记录值。

描述外模式的数据定义语言称为“外模式DDL”。有了外模式后，程序员不必关心概念模式，而只与外模式发生联系，按照外模式的结构存储和操纵数据。

###### 3. 内模式

内模式(也称为存储模式)是数据库在物理存储方面的描述，包括定义所有的内部记录类型、索引、文件的组织方式，以及数据控制方面的细节。

内部记录并不涉及物理记录，也不涉及设备的约束，比内模式更接近于物理存储和访问的那些软件机制，是操作系统的一部分(即文件系统)，例如，从磁盘读数据或写数据到磁盘上的操作等。

###### 4. 概念模式/内模式映射

概念模式/内模式映射存在于概念级和内部级之间，用于定义概念模式和内模式间的对应性，有时也称为“模式/内模式映射”。由于这两级的数据结构可能不一致，即记录类型、字段类型的组成可能不一样，因此需要这个映射来说明概念记录和内部记录间的对应性。

如果数据库的内模式要做修改，即数据库的存储设备和存储方法有所变化，那么概念模式/内模式映射也要做出相应的修改，但概念模式很可能仍然保持不变，也就是对内模式的修改尽量不影响概念模式，当然对于外模式和应用程序的影响更小，这样我们就称数据库达到了物理数据独立性。

概念模式/内模式映射一般放在内模式中描述。

###### 5. 外模式/概念模式映射

外模式/概念模式映射存在于外部级和概念级之间，用于定义外模式和概念模式间的对应性，即外部记录类型和概念记录类型间的对应性。

如果数据库的整体逻辑结构(即概念模式)要做修改，那么外模式/概念模式映射也要做相应的修改，但外模式很可能仍然保持不变，也就是对概念模式的修改尽量不影响外模式。当然对于应用程序的影响就更小，这样我们就称数据库达到了逻辑数据独立性。

外模式/概念模式映射一般放在外模式中描述。

#### 二、集中式数据库系统

所谓集中式数据库系统，就是集中在一个中心场地的电子计算机上，以统一处理方式支持的数据库系统。这类数据库无论是逻辑上还是物理上，都是集中存储在一个容量足够大的外存储器上，其基本特点如下。

* 集中控制处理效率高、可靠性好。
* 数据冗余少、数据独立性高。
* 易于支持复杂的物理结构获得对数据的有效访问。

#### 三、C/S数据库体系结构

简而言之，C/S(客户/服务器)数据库体系是在网络基础上，以数据库管理为后援，以微机为工作站的一种系统结构。

客户/服务器处理模型是共享设备处理的一种自然扩充。在这种模型中应用被划分成两部分，分别在客户端和服务器端完成。

采用客户/服务器体系结构，客户机主要负责数据表示服务，而服务器主要负责数据库服务。

ODBC(开放的数据库连接)和JDBC(Java程序数据库连接)标准定义了应用程序和数据库服务器通信的方法，也就是定义了应用程序接口。应用程序可以用它们来打开与数据库的连接，发送查询和更新以及获取返回结果。

#### 四、并行数据库系统

并行数据库系统由多个物理上连在一起的CPU构成。并行体系结构的数据库类型分为共享内存式多处理器和无共享式并行体系结构。

###### 1. 共享内存式多处理器

共享内存式多处理器是指一台计算机上同时有多个活动的CPU，它们共享单个内存和一个公共磁盘接口。

###### 2. 无共享式并行体系结构

无共享式并行体系结构是指一台计算机上同时有多个活动的CPU，但它们都有自己的内存和磁盘。

#### 五、分布式数据库系统

分布式数据库系统由多个地理上分开的CPU构成，每个场地都有独立处理能力并能完成局部应用；每个场地也都参加全局应用程序的执行，通过网络通信访问系统中多个场地的数据。数据库中的数据不是存储在同一场地中的，这就是分布式数据库的“分布性”特点，也是与集中式数据库的最大区别。

#### 六、Web数据库

Web数据库就是用户利用浏览器作为输入接口，输入所需要的数据，浏览器将这些数据传送给网站，而网站再对这些数据进行处理，或者对后台数据库进行查询操作等，最后网站将操作结果传回给浏览器，通过浏览器将结果告知用户。网站上的后台数据库就是Web数据库。

通常，Web数据库的环境由硬件元素和软件元素组成。硬件元素包括Web服务器、客户机、数据库服务器和网络。软件元素包括客户端必须有能够解释执行HTML代码的浏览器，如IE、Netscape等；在Web服务器中，必须具有能自动生成HTML代码的程序，如ASP、JSP、CGI等；具有能自动完成数据操作指令的数据库系统，如Access、SQL Server等。Web数据库的结构一般采用3层结构方式，它由4类3层组成。

###### 1. 第一层

第一层是浏览器(browser)层，该层又称为客户(client)层，它由客户机上的浏览器组成，它是一种用户逻辑层，负责与客户互联。

###### 2. 第二层

第二层是应用服务器层，该层由Web服务器与应用服务器两部分组成。这两类服务器一般在同一层中，该层有时也称为应用服务器或Web服务器。

###### 3. 第三层

第三层是数据库服务器(Database Server)层。该层也可称为分布式数据库服务器层，其主要功能是提供数据服务，负责接受从应用服务器发送的SQL请求，通过相应处理后将结果返回给应用服务器。

Web数据库通过Web方式将网上数据库做分布式处理，构成了一种简单、有效的分布式数据库结构，也为分布式数据库提供了一种新的实现方式。

### 3.2.5 关系数据库与关系运算



#### 一、关系数据库的基本概念

###### 1. 属性和域

在关系数据模型中，用二维表格结构表示实体类型，用关键码(关键字)表示实体类型和实体间的联系。字段称为属性，字段值称为属性值，记录类型称为关系模型，记录称为元组，元组的集合称为关系或实例。

一个事物需要用若干特征来表示，这些特征称为属性。每个属性的取值范围称为该属性的域(domain)。在关系模型中，所有的域都必须是原子数据，这种限制称为第一范式条件。

###### 2. 笛卡儿积与关系

设关系*R*和*S*的元数分别为*r*和*s*。定义*R*和*S*的笛卡儿积是一个(*r*+*s*)元的元组集合，每个元组的前*r*个分量来自*R*的一个元组，后*s*个分量来自*S*的一个元组，记为*R*×*S*。若*R*有*m*个元组，*S*有*n*个元组，则*R*×*S*有(*m*×*n*)个元组。

###### 3. 关系的相关名词

* 目或度：这里*R*表示关系的名字，*n*表示关系的目或度。
* 候选码：若关系中的某一属性或属性组的值能唯一标识一个元组，则称该属性或属性组为候选码。
* 主码：用户选做元组标识的一个候选码称为主码。
* 外码：某个关系的主码相应的属性在另一关系中出现，此时该主码就是另一关系的外码。
* 全码：关系模型的所有属性组是这个关系模式的候选码，称为全码。

###### 4. 关系的性质

* 分量必须取原子值，每一个分量必须是不可再分的数据项。
* 列是同质的，每一列中的分量必须是同一类型的数据，来自同一个域。
* 属性不能重名，每列为一个属性，不同的列可来自同一个域。
* 行列的顺序无关。因为关系是一个集合，所以不考虑元组间的顺序。属性在理论上讲也是无序的，但使用时，往往会考虑顺序。
* 任何两个元组不能完全相同，这是由主码约束来保证的。但有些数据库若用户没有定义完整性约束条件，则允许有多个相同的元组。

###### 5. 关系的3种类型

* 基本关系(通常又称为基本表或基表)：是实际存在的表，它是实际存储数据的逻辑表示。
* 查询表：查询结果对应的表。
* 视图表：是由基本表或其他视图表导出的表。由于它本身不独立存储在数据库中，数据库中只存放它的定义，所以常称为虚表。

#### 二、关系数据库模式

在数据库中要区分型和值，关系数据库中的型也称为关系数据库模式，是关系数据库结构的描述。实际上关系的概念对应于程序设计语言中变量的概念，而关系模式对应于程序设计语言中类型定义的概念。关系数据库的值是这些关系模式在某一时刻对应的关系的集合，通常称为关系数据库。

关系的描述称为关系模式，可以表示为：*R*(*U*, *D*, dom, *F*)。其中*R*表示关系名；*U*是组成该关系的属性名的集合；*D*是属性的域；dom是属性向域的映射集合；*F*是属性间数据的依赖关系组合。

通常将关系模式简记为*R*(*U*)或*R*(*A*1, *A*2, …, *An*)。其中*R*表示关系名；*A*1, *A*2, …, *An*为属性名或域名。通常在关系模式主属性下加下划线来表示该属性为主码属性。

#### 三、完整性约束

为了维护数据库中数据与现实世界的一致性，对关系数据库的插入、删除和修改操作必须有一定的约束条件，这就是关系模型的三类完整性：实体完整性，参照完整性和用户定义完整性。

###### 1. 实体完整性

实体完整性(Entity Integrity)是指主属性的值不能为空或部分为空。关系模型中的一个元组对应一个实体，一个关系则对应一个实体集。例如，一条学生记录对应着一个学生，学生关系对应着学生的集合。关系模型中以主属性来唯一标识元组。例如，学生关系中的属性“学号”可以唯一标识一个元组，也可以唯一标识学生实体。

###### 2. 参照完整性

参照完整性(Referential Integrity)是指如果关系*R*2的外码*X*与关系*R*1的主码相符，则*X*的每个值或者等于*R*1中主码的某一个值，或者取空值。

实体完整性和参照完整性是关系模型必须满足的完整性约束条件，被称做关系的两个不变性。

###### 3. 用户定义完整性

用户定义完整性(User-defined Integrity)是针对某一具体关系数据库的约束条件。它反映某一具体应用所涉及的数据必须满足的语义要求。

#### 四、关系代数运算

###### 1. 关系代数的分类及其运算符

关系代数是对关系进行集合代数的运算，是基于关系代数的操作语言，称为关系代数语言，简称关系代数。关系代数的运算对象是关系，运算结果也是关系，关系代数用到的运算符主要包括以下4类。

* 集合运算符：∪(并)、-(差)、∩(交)、×(广义笛卡儿积)。
* 专门的关系运算符：(选择)、∏(投影)、▷◁ (连接)、\*(自然连接)、÷(除)。
* 算术比较运算符：>(大于)、≥(大于等于)、<(小于)、≤(小于等于)、=(等于)、≠(不等于)。
* 逻辑运算符：∧(与)、∨(或)、┐(非)。

关系代数的运算按运算符的不同主要分为以下两类。

* 传统的集合运算：把关系看成元组的集合，以元组作为集合中的元素来进行运算，其运算是从关系的“水平”方向即行的角度进行的，包括并、差、交和笛卡儿积等运算。
* 专门的关系运算：不仅涉及行运算，也涉及列运算，这种运算是为数据库的应用而引进的特殊运算，包括选取、投影、连接和除法等运算。

###### 2. 关系代数操作

并、差、笛卡儿积、投影和选择是5种基本运算，其他运算可由基本运算导出。

* 并：设有两个关系*R*和*S*，*R*和*S*的并是由属于*R*或属于*S*的元组组成的集合，记为*R*∪*S*。
* 差：*R*-*S*定义为属于*R*但不属于*S*的所有元组的集合。
* 笛卡儿积：设关系*R*和*S*的元数分别为*a*和*b*，*R*和*S*的笛卡儿积是一个(*a*+*b*)元的元组集合，每个元组的前*a*个分量来自*R*的一个元组，后*b*个分量来自*S*的一个元组。记为*R*×*S*。
* 投影：对关系进行垂直分割，消去关系中某些列，重新安排列次序，再删去重复的元组。由于某些列删除后，某些元组可能会变得完全相同，那些相同的元组经投影操作后只保留一个。所以，在关系代数中，对一个关系进行投影操作以后，新关系的元组个数小于或等于原来关系的元组个数。
* 交：*R*∩*S*定义为属于关系*R*又属于关系*S*的元组的集合。
* 选择：根据某些条件对关系作水平分割，选择符合条件的元组。
* 连接：定义为从*R*和*S*的笛卡儿积中，选择属性间满足一定条件的元组的集合。
* 自然连接：在*R*×*S*中，选择*R*和*S*公共属性值均相等的元组，并去掉*R*×*S*中重复的公共属性列。如果两个关系没有公共属性，则自然连接就转化为笛卡儿积。

如果关系*R*和关系*S*作自然连接时，将关系*R*中原舍弃的元组放到新关系中，这种操作称为“左外连接”，用符号“”表示，如*R**S*。如果关系*R*和关系*S*作自然连接时，将关系*S*中原舍弃的元组放到新关系中，那么这种操作称为“右外连接”，用符号“”表示，如*R**S*。自然连接的操作符为▷◁ ，如*R*▷◁ *S*。

### 3.2.6 关系数据库SQL语言简介



结构化查询语言(Structured Query Language，SQL)的主要功能包括：数据查询、数据控制、数据操纵和数据定义。

#### 一、SQL数据库体系结构

SQL语言支持的数据库的体系结构也是三级结构，但术语与传统关系模型术语不同，在SQL中，关系模式称为“基本表”，存储模式称为“存储文件”，基本表(Base Table)是独立存在的表，不是由其他的表导出的表。一个关系对应一个基本表，一个或多个基本表对应一个存储文件。子模式称为“视图”，视图(view)是一个虚拟的表，是从一个或几个基本表导出的表，它本身不独立存在于数据库中，数据库中可能只存放视图的定义而不存放视图对应的数据，这些数据仍存放在导出视图的基本表中。当基本表中的数据发生变化时，从视图中查询出来的数据也随之改变。元组称为“行”，属性称为“列”。

SQL语言支持数据库的三级模式结构，如图3-1所示。其中，外模式对应于视图和部分基本表，模式对应于基本表，内模式对应于存储文件。



##### 图3-1 SQL语言支持的关系数据库的三级逻辑结构

#### 二、SQL的基本组成

SQL包括了所有对数据库的操作，主要由以下几个部分组成。

* 数据定义语言：这一部分又称为“SQL DDL”，定义数据库的逻辑结构包括定义基本表、视图和索引3个部分。
* 交互式数据操纵语言：这一部分又称为“SQL DML”，包括数据查询和数据更新两大类操作，其中数据更新又包括插入、删除和修改3种操作。
* 事务控制：SQL提供定义事务开始和结束的命令。
* 完整性：SQL DDL包括定义数据库中的数据必须满足的完整性约束条件的命令，对于破坏完整性约束条件的更新将被禁止。
* 权限管理：SQL DDL中包括对关系和视图的访问权限的说明。
* 嵌入式SQL语言的使用规定：规定SQL语句在宿主语言的程序中使用的规则。

#### 三、SQL数据定义

SQL语言使用数据定义语言(DDL)实现其数据定义功能，可对数据库用户、基本表、视图、索引进行定义和撤销。SQL的DDL主要是定义基本表、视图和索引3个部分。

###### 1. 基本表的定义、修改和撤销

1) 基本表的定义

基本表定义的语法如下：

CREATE TABLE SQL 模式名.基本表名(列名,类型,… 完整性约束…)

完整性约束包括主键子句(Primary Key)、检查子句(Check)和外键子句(Foreign Key)。

2) 基本表的修改

基本表修改的语法如下：

ALTER TABLE 基本表名 ADD/DROP(增加/删除) 列名 类型名(增加时写出)

删除时有子句：

[CASCADE|RESTRICT]

前者为连锁删除，后者为约束删除，即没有对本列的任何引用时才能删除。

3) 基本表的撤销

基本表撤销的语法如下：

DROP TABLE 基本表名 [CASCADE|RESTRICT]

###### 2. 定义和删除视图

1) 视图的定义

视图定义的语法如下：

CREATE VIEW 视图名(列名表)

AS SELECT 查询语句

[WITH CHECK OPTION];

2) 视图的删除

视图删除的语法如下：

DROP VIEW 视图名

###### 3. 定义和删除索引

索引分为聚集索引和非聚集索引。聚集索引是指索引表中索引项的顺序与表中记录的物理顺序一致的索引。

1) 索引的定义

索引定义的语法如下：

CREATE [UNIQUE] INDEX 索引名 ON 基本表名(列名表 [ASC|DESC])

2) 索引的删除

索引删除的语法如下：

DROP INDEX 索引名

SQL系统中的索引一经建立，就由系统使用和维护它，用户不再干预，直到撤销为止。这种索引称为隐式索引。而FoxPro系统中的索引属于显式索引，用户经常要使用索引名打开索引文件。

综上所述，凡创建都用CREATE，删除都用DROP，改变都用ALTER，再跟类型和名字，附加子句就很容易了。

#### 四、SQL数据查询

###### 1. SELECT语句的基本结构

SELECT语句的基本结构如下：

SELECT列名表(逗号隔开) FROM 基本表或视图序列 WHERE 条件表达式

在这里，重点要掌握条件表达式中各种运算符的应用，如算术比较运算符=、>、<、<>，逻辑运算符AND、OR、NOT，集合成员资格运算符IN、NOT IN，此外，嵌套的SELECT语句的用法要特别注意理解。

在查询时，SELECT语句可以有多种写法，如连接查询、嵌套查询和使用存在量词的嵌套查询等。这些查询语句读者最好都能掌握，如果不能全部掌握，起码应能写出一种正确的查询语句。

###### 2. SELECT 语句完整的句法

SELECT语句可以由6个子句组成，但只有前两个子句是必不可少的，其他子句可以省略。完整的句法如下：

SELECT 列名表(逗号隔开)

FROM 基本表或视图序列

[WHERE 条件表达式] (此为和条件子句)

[GROUP BY 列名序列] (分组子句)

[HAVING 组条件表达式] (组条件子句)

[ORDER BY列名[ASC|DESC]…] (排序子句)

###### 3. SELECT语句中的限定

本部分内容主要是对SELECT语句的进一步使用进行深入学习，领会下列各种限定的使用目的和方法。

* 要求输出表格中不出现重复元组，则在SELECT后加DISTINCT。
* SELECT 语句中允许出现加减乘除及列名、常数的算术表达式。
* WHERE 语句中可以用BETWEEN…AND…来限定一个值的范围。
* 同一个基本表在 SELECT语句中多次引用时，可用AS来增加别名。
* WHERE语句中字符串匹配用LIKE和两个通配符%和下划线\_来实现。
* 查询结果的结构完全一致时可将两个查询进行并(UNION)、交(INTERSECT)和差(EXCEPT)操作。
* 查询空值操作不是用='null'，而是用 IS NULL来测试。
* 集合成员资格比较用 IN/NOT IN ，集合成员算术比较用元组θSOME/ALL。
* 可以用子查询结果取名(表名(列名序列))来作为导出表使用。
* 基本表的自然连接操作是用 NATURAL INNER JOIN来实现的。

#### 五、SQL数据更新

###### 1. 插入语句

插入语句的语法如下：

INSERT INTO 基本表名(字段名[,字段名]…)

VALUES(常量[,常量]…);查询语句

INSERT INTO 基本表名(列名表)

SELECT 查询语句

其中元组值可以连续插入。用查询语句可以按要求插入所需数据。

###### 2. 删除语句

删除语句的语法如下：

DELETE FROM 基本表名

[WHERE条件表达式]

###### 3. 修改语句

修改语句的语法如下：

UPDATE 基本表名

SET 列名=值表达式(,列名=值表达式…)

[WHERE 条件表达式]

#### 六、SQL的访问控制

SQL的访问控制功能主要是指对用户访问数据的控制。数据库中的数据由多个用户共享，为保证数据库的安全，SQL语言提供数据控制语句(Data Control Language，DCL)对数据库进行统一的控制管理。

###### 1. 权限

权限机制的基本思想是：给用户授予不同类型的权限，在必要时，可以收回授权。使用户能够进行的数据库操作及所操作的数据限定在指定的范围内，禁止用户超越权限对数据库进行非法的操作，从而保证数据库的安全性。在SQL Server中，权限可分为系统权限和对象权限。

###### 2. 权限的授予与收回

1) 授予权限的语句格式

SQL语言使用GRANT语句为用户授予系统权限，其语法格式为：

GRANT <权限> [,<权限>]…

[ON <对象类型><对象名>]

TO <用户> [,<用户>]…

[WITH GRANT OPTION]

其语义为：将指定的系统权限授予指定的用户或角色。其中，用户可以是单个或多个，也可以是PUBLIC，PUBLIC代表数据库中的全部用户；WITH GRANT OPTION为可选项，指定后则允许被授权的用户将指定的系统特权或角色再授予其他用户或角色。例如，为用户ZHANGSAN授予CREATE TABLE系统权限的语句如下：

GRANT CREATE TABLE

TO ZHANGSAN

2) 收回权限的语句格式

数据库管理员可以使用REVOKE语句收回权限，其语法格式为：

REVOKE <权限> [,<权限>]…

[ON <对象类型><对象名>]

FROM <用户> [,<用户>]…

例如，收回用户USER1对C表查询权限的语句如下：

REVOKE SELECT

ON C

FROM USER1

#### 七、嵌入式SQL

由于SQL是基于关系模型的语言，而高级语言基于整数、实数、字符、记录、数组等数据类型，因此两者之间尚有很大差别。譬如SQL语句不能直接使用指针、数组等数据结构，而高级语言一般不能直接进行集合的操作。为了能在宿主语言的程序中嵌入SQL语句，必须对使用方式作某些规定。嵌入式SQL与宿主语言之间的通信有以下3种方式。

###### 1. SQL通信区

SQL通信区向宿主语言传递SQL语句执行的状态信息，使宿主语言能够根据此信息控制程序流程。

###### 2. 主变量

宿主变量也称共享变量。宿主语言向SQL语句提供参数主要通过主变量来实现，主变量由宿主语言的程序定义，并用SQL的DECLARE语句说明。

###### 3. 游标SQL

SQL语言是面向集合的，一条SQL语句可产生或处理多条记录。而宿主语言是面向记录的，一组主变量一次只能放一条记录，所以引入游标，通过移动游标指针来决定获取哪一条记录。

### 3.2.7 数据库设计



#### 一、概述

###### 1. 软件生存期

软件生存期是指从软件的规划、研制、实现、投入运行后的维护，直到它被新的软件所取代而停止使用的整个期间。

###### 2. 数据库系统的生存期

数据库系统的生存期是指数据库系统从开始规划、分析、设计、实现、投入运行及维护，直到它被新的系统取代而停止使用的整个期间。

###### 3. 数据库设计的特点

数据库设计主要有以下两大特点。

* 数据库建设是硬件、软件和干件(技术与管理的界面)的结合。
* 数据库设计是结构(数据)设计和行为(处理)设计的结合。

###### 4. 数据库设计的方法

数据库设计有许多方法，主要有以下7种。

* 视图模型化及视图汇总设计法。
* 关系模式设计法。
* 新奥尔良(New Orleans)方法。
* 基于E-R模型的数据库设计方法。
* 基于3NF的设计方法。
* 基于抽象语法规范的设计方法。
* 计算机辅助设计方法。

###### 5. 数据库设计的基本过程

按照规范的设计方法，将数据库设计分为以下6个阶段。

* 需求分析。
* 概念结构设计。
* 逻辑结构设计。
* 物理结构设计。
* 数据库实施。
* 数据库的运行和维护。

注意：数据库设计还包含了应用系统的设计过程。

#### 二、数据库设计的基本步骤

###### 1. 需求分析

需求分析的任务是通过详细调查现实世界要处理的对象(组织、部门、企业等)，充分了解原系统(手工系统或计算机系统)的工作概况，明确用户需求，确定新系统的功能。

调查的重点是“数据”和“处理”，以获得用户对数据库的以下4点要求。

* 信息要求：指用户需要从数据库中获得信息的内容与性质。通过信息要求可以导出数据要求。
* 处理要求：指用户要完成什么处理功能，对处理的响应时间有什么要求。采用批处理还是联机处理方式。
* 安全性和完整性要求。
* 企业的环境特征：包括企业的规模与结构，部门的地理分布，主管部门对机构的规定与要求，对系统费用/利益的限制。

调查的方法主要有：跟班作业、开调查会、请专人介绍、设计调查表请用户填写、查阅原系统有关记录。

需求分析阶段生成的结果如下。

* 数据：数据字典(通常包括数据项、数据结构、数据流、数据存储和处理过程)，全系统中的数据项、数据流、数据存储的描述。
* 处理：数据流图和判定表、数据字典中处理过程的描述。

###### 2. 概念结构设计

概念结构设计的主要特点有：能真实地反映现实世界，包括事物和相互之间的联系，能满足用户对数据的处理要求，是对现实世界的一个真实模型；易于理解；易于更改；易于向关系、网状、层次等各种数据模型转换。

一般通过E-R模型来描述概念结构。

概念结构设计的基本方法有自顶向下、自底向上、逐步扩张、混合策略。

扩充的E-R模型概念主要包括以下内容。

* 数据的抽象。对象之间两种基本联系是聚集和概括。
* 依赖关系。一个实体的存在必须以另一个实体的存在为前提。通常将前者称为弱实体，用双线框表示，用指向弱实体的箭头表明依赖关系。

###### 3. 逻辑结构设计

逻辑结构设计的目的是把概念设计阶段的基本E-R图转换成与选用的具体机器上的DBMS所支持的数据模型相符合的逻辑结构(包括数据库模式和外模式)。逻辑设计有以下3个步骤。

* 将概念模型(E-R图)转换为一般的关系、网状或层次模型。
* 将关系、网状或层次模型向特定的DBMS支持下的数据模型转换。
* 对数据模型进行优化。

###### 4. 物理结构设计

对于给定的基本数据模型选取一个最适合应用环境的物理结构的过程，称为物理结构设计。对于一个给定的逻辑数据模式选取一个最适合应用环境的物理结构的过程，称为数据库的物理结构设计。

#### 三、数据库的实施与维护

###### 1. 数据库实现阶段的工作

数据库实现阶段的工作主要包括建立实际数据库结构，试运行，装入数据。

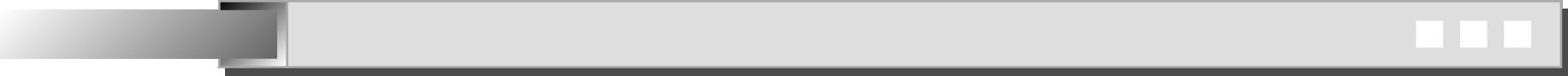
###### 2. 其他有关的设计工作

其他有关的设计工作主要包括数据库的重新组织设计，故障恢复方案设计，安全性考虑，事务控制。

###### 3. 运行与维护阶段的工作

运行与维护阶段的工作主要包括数据库的日常维护(安全性、完整性控制、数据库的转储和恢复)，性能的监督、分析与改进，扩充新功能，修改错误。

## 3.3 真题详解



### 综合知识试题

试题1(2010年下半年试题57)

在数据库设计中，E-R模型常用于 (1) 阶段。

(1) A. 需求分析　　 B. 概念设计　　 C. 逻辑设计　　 D. 物理设计

**参考答案：**(1)A。

**要点解析：**本题考查的知识点是E-R图。E-R图也即实体-联系图(Entity-Relationship Diagram)，提供了表示实体型、属性和联系的方法，用来描述现实世界的概念模型。E-R图设计属于数据库设计的需求分析阶段。

试题2~4(2010年下半年试题58~60)

某公司数据库的两个关系：部门(部门号, 部门名, 负责人, 电话)和员工(员工号, 姓名, 住址)。假设每个部门有若干名员工，一名负责人，一部电话；员工号为员工关系的主键。

a. 若部门名是唯一的，请将下述SQL语句的空缺部分补充完整。

CREATE TABLE部门(部门号CHAR(3)PRIMARY KEY,  
      部门名CHAR(10) (2) ,  
      负责人CHAR(4),  
      电话CHAR(20))  
          (3) );

(2) A. NOT NULL　　 B. UNIQUE　　C. KEY UNIQUE　　D. PRIMARY KEY  
 (3) A. PRIMARY KEY(部门号)NOT NULL UNIQUE  
 B. PRIMARY KEY(部门名)UNIQUE  
 C. FOREIGN KEY(负责人)REFERENCES员工(姓名)  
 D. FOREIGN KEY(负责人)REFERENCES员工(员工号)

b. 查询各部门负责人的姓名及住址的SQL语句如下：

SELECT部门名, 姓名, 住址  
      FROM部门, 员工 WHERE (4) ;

(4) A. 员工号=负责人　B. 员工号=‘负责人’ C. 姓名＝负责人　D. 姓名=‘负责人’

**参考答案：(2)B；(3)D；(4)D。**

**要点解析：部门关系中，部门名唯一，所以应使用UNIQUE来约束；定义了部门关系中负责人为外码，其取值区域来自于员工关系。WHERE后表示要查找的条件为“姓名=‘负责人’”。**

试题5和试题6(2010年下半年试题61和试题62)

设学生关系Students(Sno, Sname, Sex, Sdept, Sage, Sadd), Sno、Sname、Sex、Sdept、Sage、Sadd分别表示学生的学号、姓名、性别、所在系、年龄和住址。查询比计算机系(CS)所有学生年龄都小于其他系的学生姓名及年龄的SQL语句为：

SELECT Sname,Sage  
FROM Students  
WHERE Sage (5)   
    (SELECT Sage  
        FROM Students  
        WHERE (6) )  
    AND Sdept<>'CS';

(5) A. <MAX 　　　 B. >MIN　　　　 C. <ALL 　　　 D. <ANY  
 (6) A. Sdept=‘CS’ B. Sdept<>‘CS’ C. ‘Sdept’=CS　　 D. ‘Sdept’<>CS

**参考答案：(5)C；(6)A。**

**要点解析：该语句说明查询条件为所有学生年龄都小于其他系的学生的信息；该语句说明，我们需要从除了计算机系的其他所有关系中调出信息。**

试题7(2010年上半年试题56)

关系模型是采用 (7) 结构表达实体类型及实体间联系的数据模型。

(7) A. 树型       B. 网状      C. 线型      D. 二维表格

**参考答案：(7)**D。

**要点解析：数据模型主要有网状模型、层次模型、关系模型和面向对象模型。尽管采用网状模型和层次模型的网状数据库和层次数据库已经很好地解决了数据的集中和共享问题，但是在数据独立性和抽象级别上仍有很大欠缺。用户在对这两种数据库进行存取时，仍然需要明确数据的存储结构，指出存取路径。而后来出现的采用关系模型的关系数据库则较好地解决了这些问题，因为，关系模型是采用二维表格结构表达实体类型及实体间联系的数据模型。关系模型有严格的数学基础，抽象级别比较高，而且简单清晰，便于理解和使用。**

试题8~13(2010年下半年试题57~62)

设有商品关系P(商品名, 条形码, 供应商号, 价格, 数量)，“条形码”唯一标识关系P中的每一个元组，商品名不能为空，供应商号是关系P的外键。另有供应商关系S(供应商号, 供应商名, 地址, 电话)。关系P中的商品名是唯一的。建立商品关系 P 的 SQL语句如下所示：

CREATE TABLE P( 商品名CHAR(30) (8) ,

  条形码CHAR(30) (9) ,

  供应商号  CHAR(5) ,

  价格  CHAR(20) ,

  数量CHAR(20)

(10) (供应商号) REFERENCES S(供应商号));

(8) A. NULL  B. UNIQUE        C. NOT NULL           D. NOT NULL UNIQUE

(9) A. NULL  B. NOT NULL    C. PRIMARY KEY   D. FOREIGN KEY

(10) A. NULL  B. NOT NULL    C. PRIMARY KEY   D. FOREIGN KEY

查询供应商及价格小于等于 2500 元且大于等于 1280 元的“电冰箱”的数量的SQL语句为：

SELECT商品名,供应商名,价格,数量

FROM  P

WHERE商品名= '电冰箱' AND (11) ;

将供应商号“12021”所供应的商品价格上涨3%的SQL语句为：

UPDATE  P

(12)

WHERE 供应商号= '12021';

(11) A. 价格Between 1280 AND 2500   B. 价格From 1280 TO 2500

C. 价格≤'2500' AND价格≥'1280'  D. '价格'≤2500 AND '价格'≥1280

(12) A. SET价格= '价格\*1.03'            B. Modify价格= '价格\*1.03'

C. SET价格=价格\*1.03               D. Modify价格=价格\*1.03

查询供应商地址包含“西安”的供应商名及电话的SQL语句为：

SELECT供应商名,电话

FROM  S

WHERE   (13)  ;

(13) A. 地址IN(西安)          B. 地址like '%西安%'

C. 地址IN('西安')        D. 地址like '西安'

**参考答案：**(8) D；(9) C；(10) D；(11) A；(12) C；(13) B。

**要点解析：试题(8)选D。关系P中商品名不能为空和商品名是唯一的，故应用NOT NULL UNIQUE来约束。**

**试题(9)选C。“条形码”唯一标识关系P中的每一个元组，故条形码是主键，故应用PRIMARY KEY来约束。**

**试题(10)选D。供应商号是关系P的外键，故应用FOREIGN KEY来约束。**

**试题(11)选A。查询供应商及价格小于等于2500元且大于等于1280元的“电冰箱”的数量的SQL语句应采用Between … AND …，故完整的SQL语句为：**

**SELECT 商品名, 供应商名, 价格, 数量**

**FORM P**

**WHERE 商品名='电冰箱' AND 价格 Between 1280 AND 2500;**

**试题(12)选C。将供应商号“12021”所供应的商品价格上涨3%的SQL语句应该采用SET…，故完整的SQL语句为：**

**UPDATE P**

**SET 价格=价格\*1.03**

**WHERE 供应商号='12021';**

**试题(13)选B。查询供应商地址包含“西安”的供应商名及电话的SQL语句应该采用like…，故完整的SQL语句为：**

**SELECT 供应商名, 电话**

**FORM S**

**WHERE 地址 like '%西安%';**

试题14(2010年上半年试题64)

数据测量时，对同一对象进行多次测量可能得到多个数值。精确度是指多次所测得的数值彼此接近的程度；准确度是指所测得的数值与真值符合的程度。实际测量时，不可能出现的情况是(14) 。

(14) A. 精确度与准确度都很好    B. 精确度很好但准确度不好

C. 精确度与准确度都不好    D. 准确度很好但精确度不好

**参考答案：(14)**D。

**要点解析：本题考查数据处理方面的基础知识。当准确度很好时，测量值都与真值符合的很好，相对应的精确度也很好。**

试题15(2009年下半年试题58)

对表3-2和表3-3进行 (15) 关系运算可以得到表3-4。

表3-2 表3-3

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 项目号 | 项目名 |  |  |  | 项目号 | 项目成员 |
| 00111 | ERP管理 |  |  |  | 00111 | 张小军 |
| 00112 | 搜索引擎 |  |  |  | 00112 | 李 华 |
| 00013 | 数据库建设 |  |  |  | 00013 | 王志敏 |
| 00211 | 软件测试 |  |  |  | 00211 | 李 华 |
| 00311 | 校园网规划 |  |  |  | 00311 | 王志敏 |

##### 表3-4

|  |  |  |
| --- | --- | --- |
| 项目号 | 项目名 | 项目成员 |
| 00111 | ERP管理 | 张小军 |
| 00112 | 搜索引擎 | 李 华 |
| 00013 | 数据库建设 | 王志敏 |
| 00211 | 软件测试 | 李 华 |
| 00311 | 校园网规划 | 王志敏 |

(15) A . 投影 B 选择 C.自然连接 D . 笛卡儿积

**参考答案：(15)**C。

**要点解析：**很显然，表3-4是表3-2和表3-3的自然连接运算的结果，将两表不同字段整合到一个表中。

试题16~19(2009年下半年试题59~62)

设有员工关系 Emp (员工号, 姓名, 性别, 部门, 家庭住址)，其中，属性“性别”的取值只能为 M 或F；属性“部门”是关系 Dept 的主键。要求可访问“家庭住址”的某个成分，如邮编、省、市、街道以及门牌号。关系 Emp 的主键和外键分别是 (16) 。“家庭住址”是个 (17) 属性，创建 Emp 关系的 SQL 语句如下：

CREATE TABLE Emp (

员工号 CHAR ( 4 ) ,

姓名 CHAR ( 10 ) ,

性别 CHAR ( l ) (18) ,

部门 CHAR ( 4 ) (19) ,

家庭住址 CHAR ( 30 ) ,

PRIMARY KEY (员工号)

);

(16) A. 员工号、部门 B. 姓名、部门

C. 员工号、家庭住址 D. 姓名、家庭住址

(17) A. 简单 B. 复合 C. 多值 D. 派生

(18) A. IN ( M , F ) B. LIKE ('M' , 'F' )

C. CHECK ('M' , ' F' ) D. CHECK (性别 IN ('M' , 'F' ) )

(19) A. NOT NULL B. REFERENCES Dept(部门)

C. NOT NULL UNIQUE D. REFERENCES Dept('部门')

**参考答案：(16)A；(17)B；(18)D；(19)B。**

**要点解析：**在员工关系 Emp中，显然唯一的员工号应作为主键，部门作为外键对应于部门表的主键。题中要求可访问“家庭住址”的某个成分，如邮编、省、市、街道以及门牌号。所以家庭住址是个复合属性。限制性别在M和F中，只需用Check语句：CHECK (性别 IN ('M', 'F'))，部门作为外键对应部门表主键部门只需使用REFERENCES 语句：REFERENCES Dept(部门)，字段名不能用引号。

试题20(2009年上半年试题57)

E-R图的三要素是 (20) 。

(20)A. 基本表、数据项和元组 B. 实体、联系和属性

C. 视图、基本表和数据项 D. 文件、记录和数据项

**参考答案：(20)**B。

**要点解析：**E-R图的三要素是实体、联系和属性。

试题21(2009年上半年试题58)

按国家规定，身份证号与公民之间存在 (21) 联系。

(21)A. 一对一 B.一对多 C.多对一 D.多对多

**参考答案：(21)**A。

**要点解析：**在我国，每个身份证号只与一个公民对应，同时一个公民也只能有一个身份证号。所以身份证号与公民之间存在一对一联系。

试题22~24(2010年下半年试题59~61)

零件关系P(零件名, 条形码, 供应商, 产地, 价格)中， (22) 属性可以作为该关系的主键。查询产于西安且名称为“P2”的零件，结果以零件名、供应商及零件价格分列表示，对应的SQL语句为：

SELECT 零件名, 供应商, 价格

FROM P

WHERE 零件名="P2" AND (23)

将供应商“S2”所供应的零件价格下调2%的SQL语句为：

UPDATE P

(24)

WHERE 供应商="S2"

(22) A. 零件名 B. 条形码 C. 产地 D. 供应商

(23) A. 条形码=西安 B. 条形码='西安'

C. 产地=西安 D. 产地='西安'

(24) A. SET 价格='价格\*0.98' B. SET 价格=价格\*0.98

C. Modify 价格='价格\*0.98' D. Modify价格=价格\*0.98

**参考答案：(22)B；(23)D；(24)B。**

**要点解析：**零件关系P(零件名, 条形码, 供应商, 产地, 价格)中，条形码可以唯一确定一个零件记录，所以条形码属性可以作为该关系的主键。查询产于西安且名称为“P2”的零件，结果以零件名、供应商及零件价格分列表示，对应的SQL语句为：

SELECT 零件名, 供应商, 价格

FROM P

WHERE 零件名="P2" AND产地='西安'

将供应商“S2”所供应的零件价格下调2%的SQL语句为：

UPDATE P

SET 价格=价格\*0.98

WHERE 供应商="S2"

试题25(2009年上半年试题62)

算法是问题求解过程的精确描述，它为解决某一特征类型的问题规定了一个运算过程。以下关于算法的叙述中，错误的是 (25) 。

(25) A. 流程图(flow chart)是算法的一种图形表示方法

B. 用伪代码描述的算法易于转换成程序

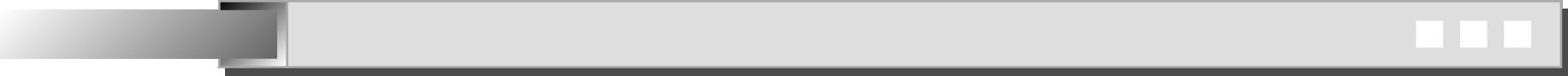
C. 用N/S盒图可以保证算法的良好结构(即由顺序、选择和重复结构来表示算法)

D. 用E-R图可以同时描述算法步骤和数据模型

**参考答案：(25)**D。

**要点解析：**用E-R图可以描述数据模型，但是不能描述算法步骤。

## 3.4 强化训练



### 3.4.1 综合知识试题

试题1和试题2

数据库是在计算机系统中按照一定的数据模型组织、存储和应用的 (1) 的集合。提供数据库各种操作的软件系统叫做 (2) 。

(1) A. 命令 B. 程序 C. 数据 D. 文件

(2) A. 数据库系统 B. 文件系统 C. 操作系统 D. 数据库管理系统

试题3

采用二维表格结构表达实体类型及实体间联系的数据模型称为 (3) 。

(3) A. 层次模型 B. 网状模型 C. 关系模型 D. 实体联系模型

试题4和试题5

设有关系Students(学号, 姓名, 年龄, 性别, 系名, 家庭住址)，其中，属性“系名”是关系DEPT的主键，用户要求访问“家庭住址”的某个成分，如邮编、省、市、街道以及门牌号。关系Students的主键和外键分别是 (4) 。“家庭住址”是一个 (5) 属性。

(4) A. 学号、家庭住址 B. 姓名、家庭住址

C. 学号、系名 D. 姓名、系名

(5) A. 简单 B. 复合 C. 多值 D. 派生

试题6

给定工程项目 PROJ 和职工 EMP 实体集，若一个职工可以参加多个项目，一个项目可以由多个职工参加，那么，PROJ 与 EMP 之间应是一个　(6)　的联系。

(6) A. 1∶1 　　　 B. N∶1 　　　 C. 1∶N 　　　　 D. M∶N

试题7和试题8

关系数据库是表的集合。对视图进行查询，本质上就是对从 (7) 中导出的数据进行查询；支持数据库各种操作的软件系统称为 (8) 。

(7) A. 一个或若干个基本表 B. 一个或若干个索引文件

C. 一个或若干个视图 D. 一个视图

(8) A. 数据库系统 B. 文件系统　　 C. 数据库管理系统 D. 操作系统

试题9和试题10

关系代数运算是以集合操作为基础的运算，其五种基本运算是并、差、 (9) 、投影和选择，其他运算可由这些运算导出。为了提高数据的操作效率和存储空间的利用率，需要对 (10) 进行分解。

(9) A. 交 B. 连接 C. 笛卡儿积 D. 自然连接

(10) A. 内模式 B. 视图 C. 外模式 D. 关系模式

试题11和试题12

通过　(11)　关系运算，可以从表3-5和表3-6获得表3-7；表3-7的主键为　(12)　。

##### 表3-5 表3-6 表3-7

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 课程号 | 课程名 |  | 课程号 | 教师名 |  | 课程号 | 课程名 | 教师名 |
| 10011 | 计算机文化 |  | 10011 | 赵军 |  | 10011 | 计算机文化 | 赵军 |
| 10024 | 数据结构 |  | 10024 | 李小华 |  | 10024 | 数据结构 | 李小华 |
| 20010 | 数据库系统 |  | 10024 | 林志鑫 |  | 10024 | 数据结构 | 林志鑫 |
| 20021 | 软件工程 |  | 20035 | 李小华 |  | 20035 | UML应用 | 李小华 |
| 20035 | UML应用 |  | 20035 | 林志鑫 |  | 20035 | UML应用 | 林志鑫 |

(11) A. 投影　 B. 选择　 C. 笛卡儿积　　　 D. 自然连接

(12) A. 课程号 B. 课程名 C. 课程号、教师名 D. 课程号、课程名

试题13~15

某银行信贷额度关系credit-in(C\_no, C\_name, limit, Credit\_balance)中的4个属性分别表示用户号、用户姓名、信贷额度和累计消费额。该关系的 (13) 属性可以作为主键。表3-8为关系credit-in的一个具体实例。

##### 表3-8 某银行信贷额度关系表

|  |  |  |  |
| --- | --- | --- | --- |
| C\_no | C\_name | limit | Credit\_balance |
| 1310001  1310002  2410003  2410004  4110041  4110042  4110812 | 张 静  陈继军  李丽莉  刘华东  赵庆民  范建华  赵庆民 | 3500  3500  23800  6600  9800  16000 | 1800  2000  2100  2000  5800  4500 |

查询累计消费额大于3000的用户姓名以及剩余消费额的SQL语句应为：

Select (14)

From credit-in

Where (15) ;

(13) A. C\_no B. C\_name　　 C. Credit\_balance D. limit

(14) A. C\_name, Credit\_balance - limit B. C\_name, limit - Credit\_balance

C. C\_name, limit, Credit\_balance D. C\_name, Credit\_balance

(15) A. limit>3000 B. Credit\_balance>3000

C. limit - Credit\_balance>3000 D. Credit\_balance - limit>3000

试题16~19

对于下面表3-9和表3-10所示的Students和SC关系，属性Sno表示学号，Sname表示姓名，Sex表示性别，SD表示专业，Age表示年龄，Cno表示课程号，Grade 表示成绩。

##### 表3-9 Students关系

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Sno | Sname | Sex | SD | Age |
| 310001  310002  410003  410004  110041  110042 | 张静  陈继军  李丽莉  刘华东  赵庆民  范建华 | 女  男  女  男  男  男 | 计算机  计算机  机械  机械  通信  通信 | 8  9  8  9  0  0 |

##### 表3-10 SC关系

|  |  |  |
| --- | --- | --- |
| Sno | Cno | Grade |
| 310002  310002  310002  410004  410004  110042  110042 | 1  2  3  2  3  1  2 | 93  84  84  83  93  84  82 |

a. 当用户查询“选了课程 2 的学生数”时，其结果应为 (16) ；当 Students 和 SC 关系进行自然连接时，其结果集的属性列数应为 (17) 。

(16) A. 3             B. 4             C. 6      D. 7

(17) A. 6      B. 7     C. 8      D. 9

b. 查询“至少选修了两门课程的每个学生的平均成绩”的SQL 语句如下：

SELECT Sno, (18)

FROM　SC

GROUP BY Sno

(19) >1;

(18) A. Grade         B. SUM (Grade)/ 7

C. AVG(Grade) D. SUM (Grade)/ COUNT(Cno)

(19) A. WHERE COUNT(Sno)      B. WHERE COUNT(DISTINCT (Sno))

C. HAVING COUNT(Sno)        D. HAVING COUNT(DISTINCT (Sno))

试题20和试题21

设有关系Students(学号, 姓名, 年龄, 性别, 系名, 家庭住址)，如果要查询姓“李”的且家庭住址包含“科技路”的学生的学号、姓名以及所在系，则对应的SELECT语句如下：

SELECT 学号, 姓名, 系名

FROM Students

WHERE (20) AND (21) ;

(20) A. 姓名LIKE '李\_ \_' B. 姓名LIKE '李%'

C. 姓名AS '李\_ \_' D. 姓名AS '李%'

(21) A. 家庭住址LIKE '%科技路%' B. 家庭住址LIKE '-科技路-'

C. 家庭住址AS '%科技路%' D. 家庭住址AS '-科技路-'

试题22~24

设有一个关系EMP(职工号, 姓名, 部门名, 工种, 工资)，若需查询不同部门中担任“钳工”的职工的平均工资，则相应的 SELECT 语句为：

　SELECT 部门名, AVG(工资 )AS 平均工资

　　　FROM EMP

　　　GROUP BY (22)

　　　HAVING (23)

将职工号为“23115”、姓名为“李军”、部门名为“硬件测试部”的元组插入 EMP 关系中的SQL 语句为：Insert (24) 。

(22) A. 职工号　　　　 B. 姓名　　　　 C. 部门名　　　　 D. 工种

(23) A. 工种= '钳工' B. '工种=钳工' C. '工种'= 钳工 D. 工种=钳工

(24) A. set to EMP Values(23115, 李军 , 硬件测试部,,)

　 B. set to EMP Values('23115', ' 李军 ', ' 硬件测试部',,)

　 C. into EMP Values(23115, 李军 , 硬件测试部,,)

　 D. into EMP Values('23115', ' 李军 ', ' 硬件测试部',,)

### 3.4.2 综合知识试题参考答案



【试题1和试题2】答案：(1)C；(2)D。

解析：数据库是在计算机系统中按照一定的数据模型组织、存储和应用的数据的集合。提供数据库各种操作的软件系统叫做数据库管理系统。

【试题3】答案：(3)C。

解析：采用二维表格结构表达实体类型及实体间联系的数据模型称为关系模型。

【试题4和试题5】答案：(4)C；(5)B。

解析：学号可以唯一标识一个学生，所以它是关系Students的主键。属性“系名”是关系DEPT的主键，由此可以得知，系名是Students的外键。简单属性不可再分，而复合属性可以分为更小的部分。由于“家庭住址”可以进一步分为省、市、区等，所以它是一个复合属性。

【试题6】答案：(6)D。

解析：本题考查的是E-R模型的基本概念。依题意可知，PROJ 与 EMP 之间应是多对多的联系，即M∶N。

【试题7和试题8】答案：(7)A；(8)C。

解析：关系数据库是表的集合。对视图进行查询，本质上就是对从一个或若干个基本表中导出的数据进行查询；支持数据库各种操作的软件系统称为数据库管理系统。

【试题9和试题10】答案：(9)C；(10)D。

解析：关系代数运算是以集合操作为基础的运算，其五种基本运算是并、差、笛卡儿积、投影和选择。为了提高数据的操作效率和存储空间的利用率，需要对关系模式进行分解。

【试题11和试题12】答案：(11)C；(12)D。

解析：显然表3-7是表3-5与表3-6自然连接的结果，表3-5和表3-6具有相同的属性课程号，进行等值连接后，去掉重复属性列就可以得到表3-7。从表3-7可以得知，“课程号、教师名”能决定表中记录的唯一性，所以“课程号、教师名”是表3-7的主键。

【试题13～试题15】答案：(13)A；(14)B；(15)B。

解析：从表3-8中可以看出，只有C\_no能唯一标识一条记录。所以主键应该为C\_no。剩余消费额应该为信贷额减去累计消费额，即limit-Credit\_balance，所以查询累计消费额大于3000的用户姓名以及剩余消费额的SQL语句应为：

　　Select C\_name ,limit - Credit\_balance

　　From credit-in

Where Credit\_balance>3000;

【试题16～试题19】答案：(16)A；(17)B；(18)C；(19)D。

解析：表3-10为学生选课表，表中有三个不同的学号，由此可知“选了课程2的学生数”为3，当学生表和选课表进行自然连接时，其结果集的属性列数为7 。求学生的平均值需要用到AVG函数。学生分组条件GROUP BY 后应该用HAVING，完整的SQL语句如下：

SELECT Sno，AVG(Grade)

FROM　SC

GROUP BY Sno

HAVING COUNT(DISTINCT (Sno))>1；

【试题20和试题21】答案：(20)B；(21)A。

解析：模糊查询通常使用LIKE操作符，“%”匹配任意字符串，“\_”匹配任意一个字符。所以完整的SQL语句为：

SELECT 学号, 姓名, 系名

FROM Students

WHERE 姓名LIKE '李%' AND家庭住址LIKE '%科技路%';

【试题22～试题24】答案：(22)C；(23)A；(24)D。

解析：按题意，查询不同部门中“钳工”的平均工资，需要按部门进行分组，而且必须满足的条件工种='钳工'。根据插入语句的基本格式可知，24题的A、B、C三项均有误。