

一、问答题

1、下列字符表示成相应的 ASCII 码是多少？

- (1) 换行 0AH (2) 字母 “Q” 51H
(3) 空格 20H

2、下列各机器数所表示数的范围是多少？

- (1) 8 位二进制无符号定点整数；
0~255
(2) 8 位二进制无符号定点小数；0.996094
(3) 16 位二进制无符号定点整数；0~65535
(4) 用补码表示的 16 位二进制有符号整数；-32768~32767

3、 $(111)_x=273$ ，基数 $X=?$ 16

4、有一个二进制小数 $X=0.X_1X_2X_3X_4X_5X_6$

- (1) 若使 $X \geq 1/2$ ，则 $X_1 \cdots X_6$ 应满足什么条件？ $X_1=1$
若使 $X > 1/8$ ，则 $X_1 \cdots X_6$ 应满足什么条件？ $X_1 \vee X_2 \vee X_3=1$
(2)

5、有两个二进制数 $X=01101010$ ， $Y=10001100$ ，试比较它们的大小。

- (1) X 和 Y 两个数均为无符号数； $X > Y$
(2) X 和 Y 两个数均为有符号的补码数。 $X < Y$

二、填空题

1、对于 R 进制数来说，其基数（能使用的数字符号个数）是 R 个，能使用的最小数字符号是 0。

2、十进制数 72 转换成八进制数是 110Q。

3、与十进制数 67 等值的十六进制数是 43H。

4、二进制数 101.011 转换成十进制数是 5.375。

5、十六进制数 0E12 转换成二进制数是 110000010010B。

三、选择题

1、在计算机内部，一切信息的存取、处理和传送都是以 D 形式进行的。

- A) EBCDIC 码 B) ASCII 码 C) 十六进制编码 D) 二进制编码

2、与十进制数 56 等值的二进制数是 A。

- A) 111000 B) 111001 C) 101111 D) 110110

3、十进制数 59 转换成八进制数是 A。

- A) 73Q B) 37Q C) 59Q D) 112Q

4、与十进制数 58.75 等值的十六进制数是 B。

- A) A3.CH B) 3A.CH C) 3A.23H D) C.3AH

5、二进制数 1101.01 转换成十进制数是 C。

- A) 17.256 B) 13.5 C) 13.25 D) 17.5

6、二进制数 111010.11 转换成十六进制数是 B。

- A) 3ACH B) 3A.CH C) 3A3H D) 3A.3H

7、十六进制数 1CB.D8 转换成二进制数是 D。

- A) 110001011.11001 B) 111101011.11101
C) 111101011.11101 D) 111001011.11011

8、将十六进制数 FF.1 转换成十进制数是 C。

- A) 255.625 B) 250.625 C) 255.0625 D) 250.0625

9、十进制正数 43 的八位二进制原码 B。

- A) 00110101 B) 00101011 C) 10110101 D) 10101011

10、十进制正数 38 的八位二进制补码是 D。

- A) 00011001 B) 10100110 C) 10011001 D) 00100110

四、是非判断题

- 1、在第三代计算机期间出现了操作系统。 (√)
2、对于种类不同的计算机，其机器指令系统都是相同的。(×)
3、在计算机中，数据单位 bit 的意思是字节。 (×)
4、八进制数的基数为 8，因此在八进制数中可以使用的数字符号是 0、1、2、3、4、5、6、7、8。(×)
5、在汉字国标码 GB2312-80 的字符集中，共收集了 6763 个常用汉字。(√)

答案：

一、问答题

- 1、(1) 0AH (2) 51H (3) 20H
2、(1) 0~255 (2) 0.996094 (3) 0~65535 (4) -32768~32767
3、16
4、(1) $X_1=1$ (2) $X_1 \vee X_2 \vee X_3=1$
5、(1) $X>Y$ (2) $X<Y$

二、填空题

- 1、R 个，0
2、110Q
3、43H
4、5.375
5、110000010010B

三、选择题

- 1、D 2、A 3、A 4、B 5、C 6、B 7、D 8、C 9、B 10、D

四、是非判断题

- 1、√ 2、× 3、× 4、× 5、√

1. 微处理器，微型计算机和微型计算机系统三者之间有何区别？

答：微处理器即 CPU，它包括运算器、控制器、寄存器阵列和内部总线等部分，用于实现微型计算机的运算和控制功能，是微型计算机的核心；一台微型计算机由微处理器、内存储器、I/O 接口电路以及总线构成；微型计算机系统则包括硬件系统和软件系统两大部分，其中硬件系统又包括微型计算机和外围设备；由此可见，微处理器是微型计算机的重要组成部分，而微型计算机系统又主要由微型计算机作为其硬件构成。

2. CPU 在内部结构上由哪几部分构成？CPU 应具备哪些主要功能？

答：CPU 在内部结构上由运算器、控制器、寄存器阵列和内部总线等各部分构成，其主要功能是完成各种算数及逻辑运算，并实现对整个微型计算机控制，为此，其内部又必须具备传递和暂存数据的功能。

3. 累加器和其它通用寄存器相比有何不同？

答：累加器是通用寄存器之一，但累加器和其它通用寄存器相比又有其独特之处。累加器除了可用做通用寄存器存放数据外，对某些操作，一般操作前累加器用于存放一个操作数，操作后，累加器用于存放结果。

4. 微型计算机的总线有哪几类？总线结构的特点是什么？

答：微型计算机的总线包括地址总线、数据总线和控制总线三类，总线结构的特点是结构简单、可靠性高、易于设计生产和维护，更主要的是便于扩充。

5. 举出 10 个微型计算机应用的例子。

答：略

6. 计算机 I/O 接口有何用途？试列出 8 个 I/O 接口。

答：计算机 I/O 接口是连接计算机和外部设备的纽带和桥梁，它主要用于协调和控制计算机与外设之间的信息流通和交换。例如：串行通讯口（COM 口）、并行打印机口、软盘驱动器接口、硬盘驱动器接口、光驱接口、显示器接口、音响设备接口、其它通用设备接口（USB、SCSI 等）。

7. 现在，计算机主板上一般都包括哪些 I/O 接口？I/O 接口卡如何与主板相连？

答：现在，计算机主板上一般包括串行通讯口、并行打印口、软盘驱动器接口、硬盘驱动器接口、光驱接口、USB 接口等。象显示器适配器、网卡、modem 卡等 I/O 接口卡一般通过总线插槽与主板相连。

8. 简述系统总线，AGP 总线，PCI 总线及 ISA 总线的作用。

答：系统总线是 CPU 与存储器及桥接器之间传递信息的通道，AGP 总线专门用于连接 CPU 与显示器适配器，PCI 总线一般用于连接一些高速外设接口作为高速外设与 CPU 或内存交换信息的通道，而 ISA 总线一般用于连接一些非高速外设接口作为非高速外设与 CPU 或内存交换信息的通道。

9. 试说明计算机用户，计算机软件，计算机硬件三者的相互关系。

答：计算机用户，计算机软件系统，计算机硬件系统共同构成一个计算机应用系统，三者在该系统中处于三个不同的层次。计算机用户处于最高层，计算机软件处于中间层，计算机硬件系统处于最下层。在这里计算机用户是系统的主宰，他们通过软件系统与硬件系统发生关系，指挥计算机硬件完成指定的任务。即，计算机用户使用程序设计语言编制应用程序，在系统软件的干预下使用硬件系统进行工作。

10. 简述 DOS 操作系统各部分的作用及相互关系。

答：DOS 操作系统包括 3 个模块:DOS-Shell(COMMAND.COM)、DOS-Kernel(IBM DOS.COM)、DOS-BIOS(IBM BIOS.COM)。

DOS-Shell 模块对用户输入的 DOS 命令行或应用程序行作出响应。即负责 DOS 命令的解释和任务的分配,具体工作则要靠 DOS-Kernel 模块所提供的系统功能完成。

DOS-Kernel 模块尽管提供了许多的系统功能，但由于执行每一个系统功能过程中，完全依赖各种设备实现指定的功能，因此，它还要进一步调用 DOS-BIOS 模块中的设备驱动程序才能工作。

DOS-BIOS 模块对 DOS-Kernel 传送的"请求"进行解释，最终转换为对固化在 ROM-BIOS 中的设备控制程序的请求并由它们去控制硬件,完成指定的操作。

11. 存储单元的选择由什么信号控制？读、写靠什么信号区分？

答：存储单元的选择由地址信号控制，而对存储单元进行读操作还是写操作则要靠读、写信号区分。

12. 以一个可执行文件的运行为例，简述程序执行过程。

答：当在 DOS 提示符下键入一个可执行文件名称（或在其它操作系统环境下执行有关操作）后，操作系统自动将该文件从外存装入内存并使指令指针指向其第一条指令，从而启动文件执行过程。首先将第一条指令从内存取入 CPU 中译码执行，同时指令指针自动加 1 或按指令的要求作出相应改变，指向下一条要执行的指令，接着将下一条指令从内存取入 CPU 译码执行，这样不断重复取指令和执行指令的过程，逐条执行指令，直至程序结束。

1.把下列各数化成二进制数和八进制数(二进制取 3 位小数，八进制取一位小数): $7+3/4$, $\pm 3/64$, 73.5, 725.9375, 25.34

答： $7+3/4=111.110B$; $7+3/4=7.6O$; $\pm 3/64=\pm 0.000B$; $\pm 3/64=\pm 0.0O$; $73.5=1001001.100B$; $73.5=111.4O$;

$725.9375=1011010101.111B$; $725.9375=1325.7O$; $25.34=11001.011B$; $25.34=31.3O$

2.把下列各数化成十进制数: 101.10011B , 22.2O, AD.4H

答： $101.10011B=5.59375$; $22.2O=18.25$; $AD.4H=173.25$

3.完成下列二进制数运算: $101.111+11.011$, $1001.10-110.01$, $101.11*11.01$, $101110111/1101$

答： $101.111+11.011=1001.01$; $1001.10-110.01=11.01$;

$101.11*11.01=10010.1011$; $101110111/1101=11100.11$

4.完成下列各十六进制数的运算: A39E+28DC, D5AB-7CE5, 2BF*4C, C16E/3A

答： $A39E+28DC=CC7A$; $D5AB-7CE5=58C6$; $2BF*4C=D0B4$; $C16E/3A=355.C2$

5.先将 $15B*8E/2A$ 中的十六进制数化成十进制数，再进行计算，最后再将结果化为十六进制数。

答： $15BH*8EH/2AH=347*142/42=49274/42=1173.19=495.30AH$

6.试分别判断下列各组数据中哪个数据最大？哪个数据最小？

(1) $A=0.1001B$, $B=0.1001D$, $C=0.1001H$

(2) A=10111101B, B=1001D, C=111H

答: (1) A 最大, C 最小; (2) B 最大, A 最小;

1.写出下列各数的二进制原码和补码(最后两个用双字节): 0, 96, -128, -38H, 127, 105, 879H, -32768

答: 上述各数的原码依次为: 00000000(10000000), 01100000, 无, 10111000, 01111111, 01101001, 0000100001111001, 无;

上述各数的补码依次为: 00000000, 01100000, 10000000, 11001000, 01111111, 01101001, 0000100001111001, 1000000000000000;

2.分别列出下述 10 进制数的 16 进制数、非压缩的 BCD 数、压缩的 BCD 数、ASCII 数字串(用 16 进制形式写出):10, 64, 78, 81, 92, 100, 125, 255

答: 上述各数的 16 进制数依次为: AH,40H,4EH,51H,5CH,64H,7DH,FFH;

上述各数的非压缩的 BCD 数依次为: 0100H,0604H,0708H,0801H,0902H,010000H,010205H,020505H;

上述各数的压缩的 BCD 数依次为: 10H,64H,78H,81H,92H,0100H,0125H,0255H;

上述各数的 ASCII 数字串依次为: 3130H,3634H,3738H,3831H,3932H,313030H,313235H, 323535H;

3.用 10 进制数写出下列补码表示的机器数的真值:71H, 1BH, 80H, F8H, 397DH, CF42H, 9350H

答: 上述补码表示的各机器数的真值用 10 进制数分别表示为: +113,+27,-128,-8,+14717,-20670,-27828

4.若用一个字节来表示带符号数,判断下列各运算在机内进行时是否会产生溢出,写出判断过程。

A. 5BH+32H; B. -08H-15H; C. -51H+(-3DH); D. 2DH+3CH

答：A. 产生溢出，5BH=01011011B 其补码表示的机器数为:01011011

32H=00110010B 其补码表示的机器数为:00110010

相加的结果为:10001101

数值最高位向符号位进位,但符号位向前无进位,故产生溢出。

B. 不产生溢出，-08H=-00001000B 其补码表示的机器数为:11111000

-15H=-00010101B 其补码表示的机器数为:11101011

相加的结果为:111100011

数值最高位向符号位进位,符号位同时也向前进位,故不产生溢出。

C.产生溢出,-51H=-01010001B 其补码表示的机器数为:10101111

-3DH=-00111101B 其补码表示的机器数为:11000011

相加的结果为:101110010

数值最高位向符号位无进位,但符号位向前进位,故产生溢出。

D.不产生溢出,2DH=00101101B 其补码表示的机器数为:00101101

3CH=00111100B 其补码表示的机器数为:00111100

相加的结果为:01101001

数值最高位向符号位无进位,符号位向前也无进位,故不产生溢出。

5.从键盘敲入一个大写字母，如何转换为与其相对应的小写字母？从键盘敲入 16 进制数字符 0~F，如何转换为其相对应的二进制数(00000000~00001111)？

答：从键盘敲入一大写字母后,将其 ASCII 码加上 20H,就转换成了与其相对应的小写字母。

从键盘敲入 16 进制数字 0~9 后,将其 ASCII 码值减去 30H,就转换成了与其相对应的二进制数.

从键盘敲入 16 进制数字 A~F 后,将其 ASCII 码值减去 37H,就转换成了与其相对应的二进制数.

6.详细叙述总线缓冲器的作用。

答：总线缓冲器的作用主要是控制各路数据在总线上的交叉传送避免相互冲突，当几路数据都要向总线上传送时，就通过各路的缓冲器来解决，当一路传送时，缓冲器使其它各路数据与总线断开。

7.锁存器和寄存器有什么不同？

答：锁存器与寄存器都是用来暂存数据的器件，在本质上没有区别，不过寄存器的输出端平时不随输入端的变化而变化，只有在时钟有效时才将输入端的数据送输出端（打入寄存器），而锁存器的输出端平时总随输入端变化而变化，只有当锁存器信号到达时，才将输出端的状态锁存起来，使其不再随输入端的变化而变化。

1. 8086 从功能上分成了 EU 和 BIU 两部分。这样设计的优点是什么？

答：传统计算机在执行程序时，CPU 总是相继地完成取指令和执行指令的动作，即，指令的提取和执行是串行进行的。而 8086CPU 在功能上分成了 EU 和 BIU 两部分，BIU 负责取指令，EU 负责指令的执行，它们之间既互相独立又互相配合，使得 8086 可以在执行指令的同时进行取指令的操作，即实现了取指令和执行指令的并行工作，大大提高了 CPU 和总线的利用率，从而提高了指令的处理速度。

2. 8086 CPU 中地址加法器的重要性体现在哪里？

答：地址加法器是 8086 CPU 的总线接口单元中的一个器件，在 8086 存储器分段组织方式中它是实现存储器寻址的一个关键器件，地址加法器将两个 16 位寄存器中的逻辑地址移位相加，得到一个 20 位的实际地址，把存储器寻址空间从 64 K 扩大到 1M，极大地扩大了微型计算机的程序存储空间，从而大大提高了程序运行效率。

3. 8086 CPU 中有哪些寄存器？分组说明用途。哪些寄存器用来指示存储器单元的偏移地址？

答：8086 CPU 中有 8 个通用寄存器 AX、BX、CX、DX、SP、BP、SI、DI；两个控制寄存器 IP、FL；四个段寄存器 CS、DS、SS、ES。8 个通用寄存器都可以用来暂存参加运算的数据或中间结果，但又有各自的专门用途。例如，AX 专用做累加器，某些指令指定用它存放操作数和运算结果；CX 为计数寄存器，在某些指令中做计数器使用；DX 为数据寄存器；BX 为基址寄存器，BP 为基址指针，SI 为源变址寄存器，DI 为目的变址寄存器，这 4 个寄存器在数据寻址中用来存放段内偏移地址（有效地址）或段内偏移地址的一部分；SP 为堆栈指示器，用来存放栈顶有效地址。两个控制寄存器用来存放有关的状态信息和控制信息。例如，标志寄存器 FL 用来存放状态标志和控制标志；而指令指针用来存放下一条要取指令的有效地址。四个段寄存器用来存放段地址。例如，CS 寄存器用来存放代码段的段地址；DS 寄存器用来存放数据段的段地址；SS 寄存器用来存放堆栈段的段地址；ES 寄存器用来存放扩展段的段地址。

4. 8086 系统中存储器的逻辑地址由哪两部分组成？物理地址由何器件生成？如何生成？每个段的逻辑地址与寄存器之间有何对应关系？

答：8086 系统中存储器的逻辑地址由段地址（段首址）和段内偏移地址（有效地址）两部分组成；存储单元的物理地址由地址加法器生成，寻址时，CPU 首先将段地址和段内偏移地址送入地址加法器，地址加法器将段地址左移 4 位并与段内偏移地址相加，得到一个 20 位的物理地址。数据段的段地址在 DS 寄存器中，段内偏移地址可能在 BX、BP、SI 或 DI 寄存器中。代码段的段地址在 CS 寄存器中，段内偏移地址在 IP 寄存器中。堆栈段的段地址在 SS 寄存器中，段内偏移地址在 SP 寄存器中。扩展段的段地址在 ES 寄存器中，段内偏移地址可能在 BX、BP、SI 或 DI 寄存器中。

5. 设 CPU 中各有关寄存器的当前状况为：SS=0a8bH、DS=17ceH、CS=dc54H、BX=394bH、IP=2f39H、SP=1200H，BX 给出的是某操作数的有效地址，请分别写出该操作数、下一条要取的指令及当前栈顶的逻辑地址和物理地址。

答：该操作数的逻辑地址为 DS: BX=17CE: 394BH, 物理地址=17CEH*10H+394BH=1B62BH; 下一条要取的指令的逻辑地址为 CS: IP=DC54: 2F39H, 物理地址=DC54H*10H+2F39H=DF479H; 当前栈顶的逻辑地址=SS: SP=0A8B: 1200H, 物理地址=0A8BH*10H+1200H=0BAB0H。

6. 若 DS=157DH 时, 某操作数的物理地址是 215FAH, 当 DS=18DEH 时, 该操作数的物理地址是多少?

答：该操作数的段内偏移地址=该操作数的物理地址-DS=215FAH-157D0H=BE2AH, 故当 DS=18DEH 时, 该操作数的物理地址=DS*10H+BE2AH=18DE0H+BE2AH=24C0AH

7. 设 AX=2875H、BX=34DFH、SS=1307H、SP=8H, 依此执行 PUSH AX、PUSH BX、POP AX、POP CX 后栈顶指针变为多少? AX=? BX=? CX=?

答：当前栈顶指针=SS*10H+SP=13070H+8H=13078H, 依此执行 PUSH AX、PUSH BX、POP AX、POP CX 后栈顶指针仍为 13078H。但 AX=34DFH, BX=34DFH, CX=2875H。

1. 假定 (BX) =637DH, (SI) =2A9BH, 位移量 D=3237H, 试确定在以下各种寻址方式下的有效地址是什么?

- (1) 立即寻址
- (2) 直接寻址
- (3) 使用 BX 的寄存器寻址
- (4) 使用 BX 的间接寻址
- (5) 使用 BX 的寄存器相对寻址
- (6) 基址变址寻址
- (7) 相对基址变址寻址

答：(1) 立即数寻址的有效地址是当前 IP 的内容;

(2) 直接寻址, 若使用位移量 D=3237H 进行, 则有效地址为 3237H;

(3) 使用 BX 的寄存器寻址时, 操作数在 BX 寄存器中, 因此无有效地址;

(4) 使用 BX 的间接寻址时, 有效地址在 BX 寄存器中, 即有效地址=637DH;

(5) 使用 BX 的寄存器相对寻址的有效地址=(BX) +D=637DH+3237H=95B4H;

(6) 基址变址寻址的有效地址 = (BX) + (SI) = 637DH + 2A9BH = 8E18H;

(7) 相对基址变址寻址的有效地址 = (BX) + (SI) + D = C050H;

2. 写出把首地址为 BLOCK 的数组的第 6 个字送到 DX 寄存器的指令。要求使用以下几种寻址方式:

(1) 寄存器间接寻址

(2) 寄存器相对寻址

(3) 基址变址寻址

答: (1) 使用寄存器间接寻址, 把首地址为 BLOCK 的数组的第 6 个字送到 DX 寄存器的指令为:

```
MOV BX, BLOCK
```

```
ADD BX, 12
```

```
MOV DX, [BX]
```

(2) 使用寄存器相对寻址, 把首地址为 BLOCK 的数组的第 6 个字送到 DX 寄存器的指令为:

```
MOV BX, BLOCK
```

```
MOV DX, [BX+12]
```

(3) 使用基址变址寻址, 把首地址为 BLOCK 的数组的第 6 个字送到 DX 寄存器的指令为:

```
MOV BX, BLOCK
```

```
MOV SI, 12
```

```
MOV DX, [BX+SI]
```

3. 现有 (DS)=2000H, (BX)=0100H, (SI)=0002H, (20100H)=12H, (20101H)=34H, (20102H)=56H, (20103H)=78H, (21200H)=2AH, (21201H)=4CH, (21202H)=B7H, (21203H)=65H, 试说明下列各条指令执行完后 AX 寄存器的内容。

(1) MOV AX, 1200H

(2) MOV AX, BX

(3) MOV AX, [1200H]

(4) MOV AX, [BX]

(5) MOV AX, [BX+1100]

(6) MOV AX, [BX+SI]

(7) MOV AX, [BX+SI+1100]

答：(1) 指令 MOV AX, 1200H 执行完后 AX 寄存器的内容为 1200H；

(2) 指令 MOV AX, BX 执行完后 AX 寄存器的内容为 0100H；

(3) 指令 MOV AX, [1200H] 是将从物理地址 = (DS) * 10H + 1200H = 21200H 开始的两个单元内容送 AX，执行完后 AX 寄存器的内容为 4C2AH；

(4) 指令 MOV AX, [BX] 是将从物理地址 = (DS) * 10H + (BX) = 20100H 开始的两个单元内容送 AX，故执行完后 AX 寄存器的内容为 3412H；

(5) 指令 MOV AX, [BX+1100] 是将从物理地址 = (DS) * 10H + (BX) + 1100H = 21200H 开始的两个单元内容送 AX，故执行完后 AX 寄存器的内容为 4C2AH；

(6) 指令 MOV AX, [BX+SI] 是将从物理地址 = (DS) * 10H + (BX) + (SI) = 20102H 开始的两个单元内容送 AX，故执行完后 AX 寄存器的内容为 7856H；

(7) 指令 MOV AX, [BX+SI+1100] 是将从物理地址 = (DS) * 10H + (BX) + (SI) + 1100H = 21202H 开始的两个单元内容送 AX，故执行完后 AX 寄存器的内容为 65B7H；

4. 假设已知 (DS)=2900H, (ES)=2100H, (SS)=1500H, (SI)=00A0H, (BX)= 0100H, (BP)=0010H, 数据段中变量名 VAL 的偏移地址值为 0050H, 试指出下列源操作数字段的寻址方式是什么？其物理地址值是多少？

(1) MOV AX, 0ABH

(2) MOV AX, BX

(3) MOV AX, [100H]

(4) MOV AX, VAL

(5) MOV AX, [BX]

(6) MOV AX, ES:[BX]

(7) MOV AX, [BP]

(8) MOV AX, [SI]

(9) MOV AX, [BX+10]

(10) MOV AX, VAL[BX]

(11) MOV AX, [BX][SI]

(12) MOV AX, [BP][SI]

答：（1）在指令 MOV AX, 0ABH 中，源操作数字段的寻址方式是立即数寻址，其物理地址值= (CS) *10H+ (IP) ；

（2）在指令 MOV AX, BX 中，源操作数字段的寻址方式是寄存器寻址，操作数在 BX 中，无物理地址；

（3）在指令 MOV AX, [100H] 中，源操作数字段的寻址方式是直接寻址，其物理地址值= (DS) *10H+100 =29000H+100H=29100；

（4）在指令 MOV AX, VAL 中，源操作数字段的寻址方式是直接寻址，其物理地址值= (DS) *10H+50H =29000H+50H=29050H；

（5）在指令 MOV AX, [BX] 中，源操作数字段的寻址方式是寄存器间接寻址，其物理地址值= (DS) *10H + (BX) =29000H+100H=29100H；

（6）在指令 MOV AX, ES: [BX] 中，源操作数字段的寻址方式是寄存器间接寻址，其物理地址值= (ES) *10H+ (BX) =21000H+100H=21100H；

（7）在指令 MOV AX, [BP] 中，源操作数字段的寻址方式是寄存器间接寻址，其物理地址值= (SS) *10H + (BP) =15000H+10H=15010H；

（8）在指令 MOV AX, [SI] 中，源操作数字段的寻址方式是寄存器间接寻址，其物理地址值= (DS) *10H + (SI) =29000H+0A0H=290A0H；

（9）在指令 MOV AX, [BX+10] 中，源操作数字段的寻址方式是寄存器相对寻址，其物理地址值= (DS) *10H+ (BX) +0AH= 29000H+100H+0AH =2910AH；

（10）在指令 MOV AX, VAL[BX] 中，源操作数字段的寻址方式是寄存器相对寻址，其物理地址值= (DS) *10H+ (BX) +50H= 29000H+100H+50H= 29150H；

（11）在指令 MOV AX, [BX][SI] 中，源操作数字段的寻址方式是基址变址寻址，其物理地址值= (DS) *10H+ (BX) + (SI) =29000H+100H+0A0H =291A0H；

（12）在指令 MOV AX, [BP][SI] 中，源操作数字段的寻址方式是基址变址寻址，其物理地址值= (SS) *10H+ (BP) + (SI) =15000H+10H+0A0H =150B0H

5. 分别指出下列指令中的源操作数和目的操作数的寻址方式。

（1）MOV SI, 200

（2）MOV CX, DATA[SI]

（3）ADD AX, [BX+DI]

（4）AND AX, BX

(5) MOV [SI], AX

(6) PUSHF

答：(1) 目的操作数字段的寻址方式是寄存器寻址，源操作数字段的寻址方式是立即数寻址；

(2) 目的操作数的寻址方式是寄存器寻址，源操作数的寻址方式是寄存器相对寻址；

(3) 目的操作数的寻址方式是寄存器寻址，源操作数的寻址方式是基址变址寻址；

(4) 目的操作数的寻址方式是寄存器寻址，源操作数的寻址方式也是寄存器寻址；

(5) 目的操作数的寻址方式是寄存器间接寻址，源操作数的寻址方式是寄存器寻址；

(6) 目的操作数的寻址方式是寄存器间接寻址，源操作数的寻址方式是寄存器寻址；

6. 试述指令 MOV AX, 2010H 和 MOV AX, DS:[2010H] 的区别。

答：指令 MOV AX, 2010H 是将立即数 2010H 送 AX 寄存器，而指令 MOV AX, DS:[2010H] 是将 DS 段有效地址为 2010H 的两个单元的内容送 AX。

7. 写出以下指令中内存操作数的所在地址。

(1) MOV AL, [BX+5]

(2) MOV [BP+5], AX

(3) INC BYTE PTR [SI+3]

(4) MOV DL, ES:[BX+DI]

(5) MOV BX, [BX+SI+2]

答：(1) 指令 MOV AL, [BX+5] 中内存操作数的所在地址 = (DS) * 10H + (BX) + 5；

(2) 指令 MOV [BP+5], AX 中内存操作数的所在地址 = (SS) * 10H + (BP) + 5 和 (SS) * 10H + (BP) + 6；

(3) 指令 INC BYTE PTR[SI+3] 中内存操作数的所在地址 = (DS) + (SI) + 3；

(4) 指令 MOV DL, ES:[BX+DI] 中内存操作数的所在地址 = (ES) * 10H + (BX) + (DI) ;

(5) 指令 MOV BX, [BX+SI+2] 中内存操作数的所在地址 = (DS) * 10H + (BX) + (SI) + 2 和 (DS) * 10H + (BX) + (SI) + 3;

8. 判断下列指令书写是否正确, 如有错误, 指出错在何处并用正确的程序段 (一条或多条指令) 实现原错误指令 (8、13 除外期望实现的操作)。

- | | |
|--------------------|-----------------------|
| (1) MOV AL, BX | (9) MOV ES, 3278H |
| (2) MOV AL, SL | (10) PUSH AL |
| (3) INC [BX] | (11) POP [BX] |
| (4) MOV 5, AL | (12) MOV [1A8H], 23DH |
| (5) MOV [BX], [SI] | (13) PUSH IP |
| (6) MOV BL, F5H | (14) MOV [AX], 23DH |
| (7) MOV DX, 2000H | (15) SHL AX, 5 |
| (8) POP CS | (16) MUL AX, BX |

答: (1) MOV AL, BX 错, 源操作数为字类型, 目的操作数为字节类型, 二者不一致。应改为: MOV AX, BX 或 MOV AL, BL ;

(2) MOV AL, SL 错, SI 寄存器不能分为高 8 位和低 8 位使用, 即没有 SL 寄存器。应改为: MOV AX, SI

(3) INC [BX] 错, 未指定操作数的类型。应改为: INC BYTE PTR [BX]

(4) MOV 5, AL 错, 目的操作数使用了立即数, 在指令中一般不允许。应改为: MOV DS:[5], AL

(5) MOV [BX], [SI] 错, 源操作数和目的操作数均为内存单元, 不允许。

应改为: MOV AX, [SI]

MOV [BX], AX

(6) MOV BL, F5H 错, 源操作数错, 以 A~F 开头的数字前应加 0。应改为: MOV BL, 0F5H

(7) MOV DX, 2000H 正确。

(8) POP CS 错，不能将栈顶数据弹至 CS 中。

(9) MOV ES, 3278H 错，立即数不能直接送 ES 寄存器。

应改为：MOV AX, 3278H

MOV ES, AX

(10) PUSH AL 错，栈操作不能按字节进行。应改为：PUSH AX

(11) POP [BX] 正确。

(12) MOV [1A8H], 23DH 错，源操作数是立即数，目的操作数必须使用寄存器指出。应改为：

MOV BX, 1A8H

MOV [BX], 23DH

(13) PUSH IP 错，不能用 IP 寄存器做源操作数。

(14) MOV [AX], 23DH 错，不能用 AX 寄存器间接寻址。应改为：MOV BX, AX

MOV [BX], 23DH

(15) SHL AX, 5 错，不能用大于等于 1 的立即数指出移位位数。应改为：

MOV CL, 5

SHL AX, CL

(16) MUL AX, BX 错，目的操作数 AX 是隐含的，不能在指令中写出。应改为：
MUL BX

9. 设堆栈指针 SP 的初值为 2000H, AX=3000H, BX=5000H, 试问：

(1) 执行指令 PUSH AX 后 (SP)=?

(2) 再执行 PUSH BX 及 POP AX 后 (SP)=? (AX)=? (BX)=?

答：(1) 执行指令 PUSH AX 后 (SP)=2000H-2=1FFEH;

(2) 再执行 PUSH BX 及 POP AX 后 (SP)=1FFEH, (AX)=5000H, (BX)=5000H

10. 要想完成把[2000H]送[1000H]中，用指令:MOV [1000H], [2000H]是否正确？如果不正确，应用什么方法？

答：把[2000H]送[1000H]中，用指令 MOV [1000H], [2000H]不正确，应改为：
MOV AX, [2000H] MOV [1000H], AX

11. 假如想从 200 中减去 AL 中的内容，用 SUB 200, AL 是否正确？如果不正确，应用什么方法？

答：想从 200 中减去 AL 中的内容，用 SUB 200, AL 不正确，应改为：MOV
BL, 200 SUB BL, AL

12 分别写出实现如下功能的程序段

- (1) 双字减法（被减数 7B1D2A79H，减数 53E2345FH）。
- (2) 使用移位指令实现一个字乘 18 的运算。
- (3) 使用移位指令实现一个字除以 10 的运算。
- (4) 将 AX 中间 8 位，BX 低四位，DX 高四位拼成一个新字。
- (5) 将数据段中以 BX 为偏移地址的连续四个单元的内容颠倒过来
- (6) 将 BX 中的四位压缩 BCD 数用非压缩 BCD 数形式顺序放在 AL、BL、CL、DL 中。

答：(1) 双字减法的程序段是：

```
MOV AX, 2A79H    ; 被减数的低位字送 AX
SUB AX, 345FH    ; 低位字相减，结果送 AX
MOV BX, 7B1DH    ; 被减数的高位字送 BX
SBB BX, 53E2H    ; 高位字相减处并减去低位字相减产生的借位，结果送 BX
```

(2) 使用移位指令实现一个字乘 18 的程序段是：

```
MOV AX, 05F7H    ; 被乘数送 AX
SHL AX, 1        ; 被乘数乘以 2，结果在 AX 中
MOV BX, AX       ; 被乘数乘以 2 的结果暂存到 BX
MOV CL, 3        ; 设置移位位数 3
SHL AX, CL       ; 被乘数再乘以 8（共乘以 16），结果在 AX 中
ADD AX, BX       ; 被乘数再乘以 18，结果在 AX 中
```

(3) 使用移位指令实现一个字除以 10 的运算，必须将 $X/10$ 拆分成多项的和，而每一项都应是 2 的某次幂的倒数。利用等比级数的前 N 项和公式，可求出 $A0=X/8$ ，公比 $Q=-1/4$ ，故 $X/10=X/8-X/32+X/128-X/512+\dots$ ，

所求的程序段是：

```
MOV AX, FE00H    ; 被除数送 AX
MOV CL, 3        ; 设置移位位数 3
SHR AX, CL       ; 被乘数除以 8，结果在 AX 中
MOV BX, AX       ; 被乘数除以 8 的结果暂存到 BX
MOV CL, 2        ; 设置移位位数 2
SHR AX, CL       ; 被乘数除以 4（累计除 32），结果在 AX 中
SUB BX, AX       ; 被除数/8-被除数/32，结果在 BX 中
MOV CL, 2        ; 设置移位位数 2
SHR AX, CL       ; 被乘数除以 4（累计除 128），结果在 AX 中
ADD BX, AX       ; 被除数/8-被除数/32+被除数/128，结果在 BX 中
MOV CL, 2        ; 设置移位位数 2
SHR AX, CL       ; 被乘数除以 4（累计除 512），结果在 AX 中
SUB BX, AX       ; 被除数/8-被除数/32+被除数/128-被除数/512，结果在 BX 中
```

(4) 将 AX 中间 8 位，BX 低四位，DX 高四位拼成一个新字的程序段是：

```
AND DX, 0F000H   ; 将 DX 的低 12 位清零，高 4 位不变
AND AX, 0FF0H    ; 将 AX 的低 4 位清零，高 4 位清零，中间 8 位不变
AND BX, 0FH      ; 将 BX 的高 12 位清零，低 4 位不变
ADD AX, BX
ADD AX, DX       ; 按要求组成一个新字，结果放在 AX 中。
```

(5) 将数据段中以 BX 为偏移地址的连续四个单元的内容颠倒过来的程序段是：

MOV AL, [BX] ; 数据段中 BX 为偏移地址的字单元内容送 AX

XCHG AL, [BX+3] ; 数据段中 BX+3 为偏移地址的字单元内容与 AX 的内容交换

MOV [BX], AL ; 数据段中 BX+3 为偏移地址的字单元内容送 BX 为偏移地址的字单元

MOV AL, [BX+1] ; 数据段中 BX+1 为偏移地址的字单元内容送 AX

XCHG AL, [BX+2] ; 数据段中 BX+2 为偏移地址的字单元内容与 AX 的内容交换

MOV [BX+1], AL ; 数据段中 BX+2 为偏移地址的字单元内容送 BX+1 为偏移地址的字单元

(6)将 BX 中的四位压缩 BCD 数用非压缩 BCD 数形式顺序放在 AL、BL、CL、DL 中的程序段是：

MOV DL, BL ; 四位压缩 BCD 数的低位字节送 DL

AND DL, 0FH ; DL 的高 4 位清零，得四位非压缩 BCD 数的最低位，放入 DL 中

MOV CL, 4 ; 设置移位位数 4

SHR BX, CL ; BX 中的数据逻辑右移 4 位，使四位压缩 BCD 数的次低位位于 BL 的低 4 位

MOV CH, BL ; 将 BL 的内容暂存到 CH 中保留

AND CH, 0FH ; CH 的高 4 位清零，得四位非压缩 BCD 数的次低位，放 CH 中

MOV CL, 4 ; 设置移位位数 4

SHR BX, CL ; BX 中的数据逻辑右移 4 位，使四位压缩 BCD 数的次高位位于 BL 的低 4 位

MOV AL, BL ; 将 BL 的内容暂存到 AL 中保留

AND BL, 0FH ; BL 的高 4 位清零，得四位非压缩 BCD 数的次高位，放 BL 中

MOV CL, 4 ; 设置移位位数 4

SHR AL, CL ; 使四位压缩 BCD 数的最高位位于 AL 的低 4 位，得四位非压缩 BCD 数的次高

; 位，放入 BL 中

MOV CL, CH ; 将四位非压缩 BCD 数的次低位移入 CL 中

1. 假设 OP1, OP2 是已经用 DB 定义的变量, W_OP3 和 W_OP4 是已经用 DW 定义的变量, 判断下列指令书写是否正确? 如有错误, 指出错在何处? 并写出正确的指令 (或程序段) 实现原错误指令期望实现的操作 ((19)、(20) 不改)。

- (1) PUSH OP1
- (2) POP [W_OP4]
- (3) MOV AX, WORD PTR [SI][DI]
- (4) MOV AX, WORD PTR ES:BX
- (5) MOV BYTE PTR [BX], 1000
- (6) MOV BX, OFFSET [SI+200H]
- (7) MOV OP2, [BX]
- (8) CMP HIGH W_OP3, 25
- (9) CMP OP1, OP2
- (10) CMP AX, OP2
- (11) MOV W_OP3[BX+4*3][DI], SP
- (12) ADD W_OP3, W_OP4
- (13) MOV AX, W_OP3[DX]
- (14) MOV OP1, LOW DS
- (15) MOV SP, OP2[BX][SI]
- (16) MOV AX, W_OP3+W_OP4
- (17) MOV AX, W_OP3-W_OP4+100
- (18) SUB AL, W_OP3+7
- (19) MOV AX, BX SHL 2
- (20) MOV BX, W_OP3 AND 8FD7H

答: 1. (1) PUSH OP1 错, OP1 为字节类型, 栈操作不能按字节进行, 应改为: PUSH WORD PTR OP1

(2) POP [W_OP4] 正确。

(3) MOV AX, WORD PTR [SI][DI] 错, 源操作数寻址方式有问题。

应改为: MOV BX, SI

MOV AX, WORD PTR [BX][DI]

(4) MOV AX, WORD PTR ES:BX 错, 若源操作数为寄存器寻址是不能加段说明及属性修改的。显然, 原意应为寄存器间接寻址, 故应改为: MOV AX, WORD PTR ES:[BX]

(5) MOV BYTE PTR [BX], 1000 错, 源操作数为字类型, 目的操作数为字节类型, 二者不一致。应改为: MOV WORD PTR [BX], 1000

(6)MOV BX, OFFSET [SI+200H] 错, OFFSET 运算应在汇编时完成,但[SI+200H]在执行指令时才可获得。应改为: LEA BX,[SI+200H]

(7)MOV OP2,[BX] 错, 源操作数和目的操作数均为内存单元, 不允许。

应改为: MOV AL, [BX]

MOV OP2, AL

(8)CMP HIGH W_OP3, 25 错, 目的操作数中使用 HIGH 来分离存储器操作数, 这是不允许的。应改为: CMP BYTE PTR W_OP3+1,25

(9)CMP OP1, OP2 错, 源操作数和目的操作数均为内存单元, 不允许。

应改为: MOV AL, OP2

CMP AL, OP1

(10)CMP AX, OP2 错, 源操作数为字节类型, 目的操作数为字类型, 二者不一致。

应改为: CMP AX, WORD PTR OP2

(11)MOV W_OP3[BX+4*3][DI], SP 正确。

(12)ADD W_OP3, W_OP4 错, 源操作数和目的操作数均为内存单元, 不允许。

应改为: MOV AX, W_OP4

ADD W_OP3, AX

(13)MOV AX, W_OP3[DX] 错, 不能用 DX 寄存器间接寻址。

应改为: MOV BX, DX
MOV AX, W_OP3[BX]

(14)MOV OP1, LOW DS 错, 源操作数中使用 LOW 来分离寄存器操作数, 这是不允许的。

应改为: MOV WORD PTR OP1,DS

(15)MOV SP, OP2[BX][SI] 错, 源操作数为字节类型, 目的操作数为字类型, 二者不一致。
应改为:MOV SP, WORD PTR OP2[BX][SI]

(16)MOV AX, W_OP3+W_OP4 错, 两个标识符相加无意义。应改为: MOV AX, W_OP1

ADD AX,
W_OP2

(17)MOV AX,W_OP3-W_OP4+100 正确,源操作数为两个标识符(地址)相减再加一数,汇编时得到一个数。

(18)SUB AL, W_OP3+7 错, 两个操作数类型不一致。应改为: SUB AL, BYTE PTR W_OP3+7

(19)MOV AX,BX SHL 2 错, SHL 只能对常量进行运算,左边不能使用寄存器

(20)MOV BX,W_OP3 AND 8FD7H 错, AND 只能对常量进行运算,左边不能使用变量

4. 设已定义数据段

```
DATA SEGMENT
.....
VAR2 DW 10H, 7889H
.....
VAR5 DB 20H, 0
ADDR DW VAR2, VAR5
DATA ENDS
```

为使 ADDR 字存储单元中存放内容为“0010H”, ADDR+2 字存储单元中存放内容为“0020H”, 上述省略号位置应分别填写一条什么语句? 说明理由。

答: 为使 ADDR 字存储单元中存放内容为“0010H”, ADDR+2 字存储单元中存放内容为“0020H”, 上述省略号位置应分别填写 ORG 10H 和 ORG 20H, 或者分别填写语句

VAR1 DB 10H DUP(?) 和 VAR3 6 DUP(?)

5. 下面的数据段中, 有数据为‘4100H’的字存储单元有几个? 它们的偏移量分别是多少?

```
DATA SEGMENT
DA1 DB 0
      ORG 41H
DA2 DB 0, 'A', 41H, 0, 41H
ADR DW DA1, DA2
DATA ENDS
```

答: 给出的数据段中, 有数据‘4100H’的字存储单元有 3 个, 它们的偏移量分别是 41H、44H、47H。

7. 试用数据定义语句 DB 或 DW 改写下述两语句中的某一个, 使它们在存储器中有完全相同的存储情况。

```
VAR1 DB 'abcdefghij'
VAR2 DW 6162H, 6364H, 6566H, 6768H, 696AH
```

答: 将第一个伪指令语句改写为 VAR1 DW ‘ab’, ‘cd’, ‘ef’, ‘gh’, ‘ij’, 第二个伪指令语句不变。或第一个伪指令语句不变, 而将第二个伪指令语句改写为:

```
VAR2 DB 61H, 62H, 63H, 64H, 65H, 66H, 67H, 68H, 69H, 6AH
```

9. 按下面的要求写出程序的框架

- (1)数据段的位置从 0E000H 开始，数据段中定义一个 100 字节的数组，其类型属性既是字又是字节；
- (2)堆栈段从小段开始，段组名（类别名）为 STACK；
- (3)代码段中指定段寄存器，指定主程序从 1000H 开始，给有关段寄存器赋值；
- (4)程序结束。

答：所求程序框架为：

```
Data SEGMENT AT 0E000H
    Arr_w LABEL WORD
    Arr_b DB 100 DUP(?)

Data ENDS

Sta_seg SEGMENT PARA 'STACK'

    DB 80 DUP(?)

    Top LABEL WORD

Sta_seg ENDS

Code SEGMENT

    ASSUME CS:Code, DS:data, SS:sta_seg

    ORG 1000H

Start: MOV AX, data

    MOV DS, AX

    MOV AX, Sta_seg

    MOV SS, AX

    MOV SP, OFFSET Top
```

.....

MOV AH, 4CH

INT 21H

Code ENDS

END Start

10. 编写一个程序，要求运行时屏幕显示‘BELL’，同时响铃一次(响铃的 ASCII 码为 07)。

答：所求程序为：

Code SEGMENT

ASSUME CS:code

Start: MOV DL, 42H

MOV AH, 2

INT 21H

MOV DL, 45H

MOV AH, 2

INT 21H

MOV DL, 4CH

MOV AH, 2

INT 21H

MOV DL, 4CH


```
MOV AH, 2

INT 21H

MOV DL, 7

MOV AH, 2

INT 21H

MOV AH, 4CH

INT 21H

Code ENDS

END Start
```

11. 假设在数据段 X_SEG、附加段 Y_SEG 和堆栈段 Z_SEG 中分别定义了字变量 X、Y 和 Z，试编制一完整的程序计算 $X+Y+Z$ ，并将结果送 X。

答：所求程序为：

```
X_SEG SEGMENT

    X DW 12eH

X_SEG ENDS

Y_SEG SEGMENT

    Y DW 4d2H

Y_SEG ENDS

Z_SEG SEGMENT STACK    'STACK'

    Z DW 2CAH
```

```
Z_SEG ENDS
```

```
Code SEGMENT
```

```
    ASSUME CS:Code, DS:X_SEG, ES:Y_SEG, SS:Z_SEG
```

```
Start: MOV AX, X_SEG
```

```
    MOV DS, AX
```

```
    MOV AX, Y_SEG
```

```
    MOV ES, AX
```

```
    MOV AX, ES:Y
```

```
    ADD X, AX
```

```
    MOV BP, 0
```

```
    MOV AX, [BP+Z]
```

```
    ADD X, AX
```

```
    MOV AH, 4CH
```

```
    INT 21H
```

```
Code ENDS
```

```
    END Start
```

12. 写一个完整的程序放在代码段 C_SEG 中，要求把数据段 D_SEG 中的 D_ADD1 和附加段 E_SEG 中的 D_ADD2 相加，并把结果存放在 D_SEG 段中的 SUM 中。其中 D_ADD1、D_ADD2 和 SUM 均为双字类型，D_ADD1 赋值为 98765，D_ADD2 赋值为-15893。

答：所求程序为：

D_SEG SEGMENT

DADD1 DD 98765

SUM DD ?

D_SEG ENDS

E_SEG SEGMENT

DADD2 DD -15893

E_SEG ENDS

C_SEG SEGMENT

ASSUME CS: C_SEG, DS:D_SEG, ES:E_SEG

Start: MOV AX, D_SEG

MOV DS, AX

MOV AX, E_SEG

MOV ES, AX

MOV AX, WORD PTR DADD1

MOV BX, WORD PTR DADD1+2

ADD AX, WORD PTR ES:DADD2

ADC BX, WORD PTR ES:DADD2+2

MOV WORD PTR SUM, AX

MOV WORD PTR SUM+2, BX

MOV AH, 4CH

INT 21H

C_SEG ENDS

END Start

一、填空题

- 1、执行部件 EU 的组织有：____4 个通用寄存器____，____4 个专用寄存器____和 1 个标志寄存器和算术逻辑部件_____。
- 2、8086CPU 从偶地址访问内存 1 个字时需占用 ____1 个总线____周期，而从奇地址访问内存 1 个字操作需占用 ____2 个总线____周期。
- 3、IBM-PC 机中的内存是按段存放信息的，一个段最大存贮空间为__64K____字节。
- 4、8086 微处理器在最小模式下，用____ $\overline{M}/\overline{IO}$ ____来控制输出地址是访问内存还是访问 I/O。
- 5、一台计算机能执行多少种指令，是在____设计_____ 时确定的。

二、单项选择题

- 1、微型计算机的性能主要由____B____来决定。
A、价钱 B、CPU C、控制器 D、其它
- 2、对微处理器而言，它的每条指令都有一定的时序，其时序关系是____C____
A、一个时钟周期包括几个机器周期，一个机器周期包括几个指令周期。
B、一个机器周期包括几个指令周期，一个指令周期包括几个时钟周期。
C、一个指令周期包括几个机器周期，一个机器周期包括几个时钟周期。
D、一个指令周期包括几个时钟周期，一个时钟周期包括几个机器周期。
- 3、属于数据寄存器组的寄存器是 ____C____
A、AX, BX, CX, DS B、SP, DX, BP, IP
C、AX, BX, CX, DX D、AL, DI, SI, AH
- 4、微型计算机的 ALU 部件是包含在____D____ 之中。
A、存储器 B、I/O 接口 C、I/O 设备 D、CPU
- 5、在 8086 和 8088 汇编语言中，一个字能表示的有符号数的范围是__B____
A、 $-32768 \leq n \leq 32768$ B、 $-32768 \leq n \leq 32767$
C、 $-65535 \leq n \leq 65535$ D、 $-65536 \leq N \leq 65535$
- 6、80386 微型计算机是 32 位机，根据是它的 ____D____。
A、地址线是 32 位 B、数据线为 32 位
C、寄存器是 32 位的 D、地址线和数据线都是 32 位
- 7、某数存于内存数据段中，已知该数据段的段地址为 2000H，而数据所在单元的偏移地址为 0120H，该数的在内存的物理地址为（ B ）
A. 02120H B. 20120H C. 21200H D. 03200H
- 8、在存储器读周期时，根据程序计数器 PC 提供的有效地址，使用从内存中取出（ D ）

- A. 操作数 B. 操作数地址 C. 转移地址 D. 操作码
- 9、8086/8088 系统中，对存储器进行写操作时，CPU 输出控制信号有效的是 (A)
- A. $\overline{W}/\overline{IO}=1, \overline{WR}=0$ B. $\overline{WR}=1$
- C. $\overline{M}/\overline{IO}=0, \overline{RD}=0$ D. $\overline{RD}=0$
- 10、在 8086/8088 微机系统中，将 AL 内容送到 I/O 接口中，使用的指令是 (D)
- A. IN AL, 端口地址 B. MOV AL, 端口地址
- C. OUT AL, 端口地址 D. OUT 端口地址, AL

三、简答题

1. 微处理器内部结构由哪些部件组成？
2. 论述 8086 CPU 的最小方式和最大方式的区别。
3. 论述指令周期、机器周期和时钟周期之间的关系。
4. 试比较 8086CPU 与 8088CPU 的异同之处。
5. 8086CPU 从功能上分为几部分？各部分由什么组成？各部分的功能是什么？
6. 8086 系统中的物理地址是如何得到的？假如 CS=2500H, IP=2100H, 其物理地址是多少？
7. 什么是最大模式？什么是最小模式？用什么方法将 8086/8088 置为最大模式和最小模式？
8. 存储器分段组织有何优越性？
9. 试述 BHE 与 A₀ 组合如何选择数据字？
10. CPU 响应中断时，为什么要执行两个连续的中断响应周期？
- 11.

答案：

一、填空题

- 1、4 个通用寄存器、4 个专用寄存器、1 个标志寄存器和算术逻辑部件
- 2、1 个总线、2 个总线
- 3、64K
- 4、 $\overline{M}/\overline{IO}$
- 5、设计

二、单项选择题

- 1、B 2、C 3、C 4、D 5、B 6、D 7、B 8、D 9、A 10、D

三、简答题

- 1、 答案：微处理器由两部分组成，总线接口单元和执行单元

2、8086CPU 的最小方式和最大方式的区别是：最小方式，MN/MX 接+5V (MX 为低电平)，用于构成小型单处理机系统，支持系统工作的器件：(1)时钟发生器，8284A，(2)总线锁存器，74LS373，(3)总线收发器，74LS245，控制信号由 CPU 提供。最大方式，MN/MX 接地 (MX 为低电平)，用于构成多处理机和协处理机系统，支持系统工作的器件：(1)时钟发生器，8284A，(2)总线锁存器，74LS373，(3)总线收发器，74LS245，(4)总线控制芯片，8288，控制信号由 8288 提供。

3、答：(1) .执行一条指令所需要的时间称为指令周期；(2) .一个 CPU 同外部设备和内存存储器之间进行信息交换过程所需要的时间称为总线周期；(3) .时钟脉冲的重复周期称为时钟周期；(4) .一个指令周期由若干个总线周期组成，一个总线周期又由若干个时钟周期组成；(5) . 8086CPU 的总线周期至少由 4 个时钟周期组成；(6) .总线周期完成一次数据传输包括：传送地址，传送数据；(7) 等待周期----在等待周期期间，总线上的状态一直保持不变；(8) 空闲周期----总线上无数据传输操作

4、 8088CPU 和 8086CPU 内部寄存器都是 16 位，数据总线都是有 16 位，地址总线都是有 20 位，指令系统相同。主要不同点有：

- (1) 8086 指令队列是 6 个字节长，而 8088 指令队列是 4 个字节长。
- (2) 外部数据总线位数不同，8086 为 16 位 $AD_0 \sim AD_{15}$ ，8088 为 $AD_0 \sim AD_7$
- (3) 34 脚和 28 脚功能不同，34 脚 8086 中是数据总线允许 \overline{BHE}/S_7 ，8088 是 \overline{SSO} ，

28 脚，8086 是 M/\overline{IO} ，8088 是 IO/\overline{M}

5、8086 从功能上可分为执行单元和总线接口单元。执行单元的功能是负责指令的执行，将指令译码并利用内部寄存器和 ALU 对数据进行处理。它由 4 个通用寄存器（AX、BX、CX、DX），4 个专用寄存器（BP、SP、SI 和 DI），标志寄存器和算术逻辑部件组成。总线接口单元的功能是存储器、I/O 端口传送数据。它由 4 个段寄存器（CS、DS、ES、SS），指令指针寄存器 IP，20 位地址加法器和 6 个字节的指令队列组成。

6、在 8086 系统中，用 16 位的段地址和 16 位的偏移地址来表示 20 位的物理地址，物理地址=段地址*16+偏移址得到物理地址。

该题 CS=2500H，IP=2100H，则物理地址=CS*16+IP=2500H*16+2100H=27100H

7、最大模式是相对最小模式而言，系统中包含两个或多个处理品，其中一个主处理器就是 8088/8086，其它处理器是协处理器，它是协助主处理器工作的。将 MN/MX*置 0 即为最大模式。

最小模式又是相对于最大模式，就是在系统中只有一个处理器，所有的总线控制信号都有由 8088/8086 产生，系统中总线控制逻辑电路少，将 MN/MX*置 1 即为最大模式。

8、减少了指令的长度，提高了执行速度。内存分段为程序浮动加载创造了条件。使程序保持完整的相对性，具备了浮动性。

9、 \overline{BHE} 的作用是高 8 位数据允许， $A_0=0$ 是低 8 位数据允许，在 T_1 状态时， \overline{BHE} 和 A_0 组合起来表示连接在数据总线上的存储器和 I/O 端口当前的数据在总线上出现有格式。

10、CPU 响应中断时，执行两个连续的中断响应周期，每个响应周期都给出中断响应信号 \overline{INTA} 。这是因为第一个总线周期通知中断请示设备，微处理器准备响应中断，应该准备好中断类型码，第二个总线周期中，微处理器接收中断类型码，以便得到中断向量，即中断服务程序的入口地址。

一、问答题：

- 1、微型计算机是由五部分组成，包括控制器，运算器，存储器、输入设备，输出设备；控制器控制计算机运转，计算机主要是算术运泉和逻辑运算，存储器主要是存储信

息，输入设备主要是输入信息给计算机，输出设备主要是计算机输出信息。

- 2、IBM PC 机的主存主要有只读存储器和随机存储器。只读存储器只能读出，不能写入，随机存储器可以写入也可以读出。
- 3、8086 微处理器的字长为 16 位，能直接访问的存储单元可以是 2 个字节。
- 4、8088 微处理器的逻辑地址两部分组成，包括段地址和偏移地址；物理地址=段地址*16+偏移地址。
- 5、如果一个程序在执行前（CS）=0A7F0H，（IP）=2B40H，该程序的起始地址=0A7F0H*16+2B40H=0AAA40H。
- 6、

78H	02003H
56H	02002H
34H	02001H
12H	02000H

- 7、寻址方式就是指：寻找操作数或者操作地址的各种方法，寻址方式主要有 7 种
 - (1). 立即数寻址
 - (2). 寄存器寻址
 - (3). 直接寻址
 - (4). 寄存器间接寻址
 - (5). 寄存器相对寻址方式
 - (6). 基址变址寻址方式
 - (7). 相对基址变址寻址方式

二、填空题

- 1、CPU、存储器
- 2、4、4、ALU
- 3、逻辑地址、物理地址
- 4、偏移地址，段基址*16+偏移地址，段寄存器
- 5、码、操作数
- 6、寄存器相对寻址方式，寄存器间接寻址，直接寻址，基址变址寻址方式，寄存器寻址，立即数寻址，相对基址变址寻址方式，寄存器相对寻址方式
- 7、ZF，CF，SF，OF
- 8、1，有符号
- 9、05AEH，4000H
- 10、0AB0H，1，1，0F8B0H，1，1

三、选择题

- 1、C 2、A 3、C 4、B 5、A 6、D 7、C 8、B 9、C 10、D

四、是非判断题

- 1、× 2、× 3、√ 4、√ 5、× 6、×

五、根据以下要求选用相应的指令或指令序列

- 1、MOV AX, 4629H
- 2、LDS BX, DATA
- 3、ADD DX, BX
- 4、SUB AX, 0360H
- 5、MOV BX, ES: [0500H]

6、MOV CL, 2
SHL AL, CL
7、IMUL BL
8、MOV CX, 0
9、OR DX, 0E00H
5、AND AL, 0FH
6、XOR CL, 0F0H
7、MOV CL, 2
SAR AX, CL
13、MOV CL, 4
ROL AL, CL
14、SHL AX, 1
RCL DX, 1
15、NEG DX
NEG AX
SBB DX, 0

一、问答题

1、下列字符表示成相应的 ASCII 码是多少？

- (1) 换行_____ (2) 字母“Q”_____
- (3) 空格_____

2、下列各机器数所表示数的范围是多少？

- (1) 8 位二进制无符号定点整数；
(2) 8 位二进制无符号定点小数；
(3) 16 位二进制无符号定点整数；
(4) 用补码表示的 16 位二进制有符号整数；

3、 $(111)_x=273$ ，基数 $X=?$

4、有一个二进制小数 $X=0.X_1X_2X_3X_4X_5X_6$

(3) 若使 $X \geq 1/2$ ，则 $X_1 \cdots X_6$ 应满足什么条件？

(4) 若使 $X > 1/8$ ，则 $X_1 \cdots X_6$ 应满足什么条件？

4、有两个二进制数 $X=01101010$ ， $Y=10001100$ ，试比较它们的大小。

- (1) X 和 Y 两个数均为无符号数；
(2) X 和 Y 两个数均为有符号的补码数。

二、填空题

1、对于 R 进制数来说，其基数（能使用的数字符号个数）是_____，能使用的最小数字符号是_____。

2、十进制数 72 转换成八进制数是_____。

3、与十进制数 67 等值的十六进制数是_____。

4、二进制数 101.011 转换成十进制数是_____。

5、十六进制数 0E12 转换成二进制数是_____。

三、选择题

1、在计算机内部，一切信息的存取、处理和传送都是以_____形式进行的。

- A) EBCDIC 码 B) ASCII 码 C) 十六进制编码 D) 二进制编码

2、与十进制数 56 等值的二进制数是_____。

- A) 111000 B) 111001 C) 101111 D) 110110

3、十进制数 59 转换成八进制数是_____。

- A) 73Q B) 37Q C) 59Q D) 112Q

4、与十进制数 58.75 等值的十六进制数是_____。

- A) A3.CH B) 3A.CH C) 3A.23H D) C.3AH

5、二进制数 1101.01 转换成十进制数是_____。

- A) 17.256 B) 13.5 C) 13.25 D) 17.5

6、二进制数 111010.11 转换成十六进制数是_____。

- A) 3ACH B) 3A.CH C) 3A3H D) 3A.3H

7、十六进制数 1CB.D8 转换成二进制数是_____。

- A) 110001011.11001 B) 111101011.11101
C) 111101011.11101 D) 111001011.11011

8、将十六进制数 FF.1 转换成十进制数是_____。

- A) 255.625 B) 2501625 C) 255.0625 D) 250.0625

9、十进制正数 43 的八位二进制原码_____。

- A) 00110101 B) 00101011 C) 10110101 D) 10101011

10、十进制正数 38 的八位二进制补码是_____。

- A) 00011001 B) 10100110 C) 10011001 D) 00100110

四、是非判断题

1、在第三代计算机期间出现了操作系统。 ()

2、对于种类不同的计算机，其机器指令系统都是相同的。()

3、在计算机中，数据单位 bit 的意思是字节。 ()

4、八进制数的基数为 8，因此在八进制数中可以使用的数字符号是 0、1、2、3、4、5、6、7、8。()

5、在汉字国标码 GB2312-80 的字符集中，共收集了 6763 个常用汉字。()

答案：

一、问答题

1、(1) 0AH (2) 51H (3) 20H

2、(1) 0~255 (2) 0.996094 (3) 0~65535 (4) -32768~32767

3、16

4、(1) $X_1=1$ (2) $X_1 \vee X_2 \vee X_3=1$

5、(1) $X>Y$ (2) $X<Y$

二、填空题

6、R 个，0

7、110Q

8、43H

9、5.375

10、110000010010B

三、选择题

1、D 2、A 3、A 4、B 5、C 6、B 7、D 8、C 9、B 10、D

四、是非判断题

1、√ 2、× 3、× 4、× 5、√

一、 回答下列问题

1、 设 VALA EQU 200

VALB EQU 30

VALC EQU 1BH

下列表达式的值各为多少？

(1) (VALA*VALC+VALB)/VALC

(2) (VALB AND 0FH) OR (VALB XOR 0FH)

(3) (VALA GE VALB) AND 0FH

2. 根据下列要求编写一个汇编语言程序.:

(1) 代码段的段名为 COD_SEG

(2) 数据段的段名为 DAT_SEG

(3) 堆栈段的段名为 STK_SEG

(4) 变量 HIGH_DAT 所包含的数据为 95

(5) 将变量 HIGH_DAT 装入寄存器 AH, BH 和 DL

(6) 程序运行的入口地址为 START

3、定义一个数据段的字符变量和数据变量。

(1) TRING 为字符串变量: 'Personal computer'

(2) VALD 为十进制数字变量 12, 99

(3) VALH 为十六进制数字变量 32H, 0FAH

(4) VALZ 为 10 个零的字节变量

(5) VALW 为 20 个字单元

4、定义一个数据段, 要求如下:

(1) 段界起始于字边界

(2) 该段与同名逻辑段相邻连接成一个物理段

(3) 类别号为 'DATA'

(4) 定义数据 12, 30, 'ABCD'

(5) 保留 20 个字的存储区

5、下列标号为什么是非法的?

(1) GET.DATA (2) 1_NUM (3) TEST-DATA (4) RET (5) NEW ITEM

6、下面的数据项设置了多少个字节?

(1) ASC_DATA DB '1234' (2) HEX_DATA DB 1234H

7、 DATA SEGMENT

TABLE_ADDR DW 1234H

DATA ENDS

.

.

.

MOV BX, TABLE_ADDR

LEA BX, TABLE_ADDR

请写出上述两条指令执行后, BX 寄存器中的内容。

8、写出分配给下列中断类型号在中断向量表中的物理地址。

(1) INT 12H

(2) INT 8

9、AR1 和 VAR2 为字变量, LAB 为标号, 试指出下列指令的错误之处。

(1) ADD VAR1, VAR2 (2) SUB AL, VAR1

(3) JMP LAB[SI] (4) JNZ VAR1

二、填空题

1、假设 (DS)=2000H, (BX)=1256H, (SI)=528FH; 位移量 TABLE=20A1H, (232F7H)=3280H, (264E5H)=2450H, 则

执行指令 JMP BX 后, (IP) =_____;

执行指令 JMP TABLE[BX]后, (IP) =_____;

执行指令 JMP [BX][SI]后, (IP) =_____。

2、在 1000H 单元中有一条二字节指令 JMP SHORT LAB, 如果其中偏移量分别为 30H、6CH、0B8H, 则转向地址 LAB 的值分别为_____; _____; _____;

3、数据段中有以下定义:

ARRAY1 EQU 16H

ARRAY2 DW 16H

请指出下面两条指令的寻址方式:

MOV AX, ARRAY1; 寻址方式_____。

MOV AX, ARRAY2; 寻址方式_____。

4、下面两个语句的区别在于_____、_____。

X1 EQU 1000H

X2 =1000H

5、指令“AND AX, STR1 AND STR2”中, STR1 和 STR2 是两个已赋值的变量, 两个 AND 的区别是_____、_____。

6、指令 LOOPZ/LOOPE 是结果_____或_____发生转移的指令。而指令 LOOPNZ/LOOPNE 则是结果_____或_____发生转移的指令。

7、调用指令 CALL 可分为_____、_____、_____和_____等四种调用。

8、在汇编语言中, 一个过程有 NEAR 和 FAR 两种属性。NEAR 属性表明主程序和子程序_____, FAR 属性表示主程序和子程序_____。

9、串处理指令规定源寄存器使用_____, 源串在_____段中; 目的寄存器使用_____, 目的串必须在_____段中。

10、_____是过程定义结束伪指令, 与_____配套使用。_____是段定义结束伪指令, 与_____配套使用。

三、选择题

1、执行下列程序, 选择正确的结果:

MOV SP, 2000H

MOV AX, 0F0H

MOV SI, 1234H

MOV DX, 5678H

PUSH SI

POP DI

SHL DX, 1

TEST AX, DX

PUSH DX

HLT

(1) SP=_____;

A) 2000H B) 1FFFH

C) 2001H D) 1FFEh

(2) DH 内容所在存储器的偏移地址为_____;

- A) 1FFFH B) 1FFEh
C) 1FFDH D) 1FFCH

(3) DL 内容所在存储器的偏移地址为_____;

- A) 1FFFH B) 1FFEh
C) 2000H D) 1FFCH

(4) (AX) =_____;

- A) 0H B) 78H
C) 0F0H D) 56H

(5) (DI) =_____。

- A) 1234H B) 5678H
C) 2000H D) 00F0H

(6) (DX) =_____;

- A) 5678H B) 0ACF0H
C) 2B3CH D) 0ABF0H

(7) 标志位 ZF=_____。

- A)0 B)1 C)不定 D) 无值

2、下面是多字节加法程序，第一个数和是 8A0BH，第二个数是 D705H。

```
DATA SEGMENT
FIRST DB  __ (1), __ (2), 0H
SECOND DB __ (3), __ (4)
DATA ENDS
CODE SEGMENT
    ASSUME CS: CODE, DS: DATA
START: MOV AX, DATA
        MOV DS, AX
        MOV CX, __ (5)
        MOV SI, 0
        __ (6)
NEXT:  MOV AL, SECOND[SI]
        ADC FIRST[SI], AL
        INC SI
        LOOP NEXT
        MOV AL, 0
        ADC AL, __ (7)
        MOV FIRST[SI], AL
        MOV AH, 4CH
        INT 21H
CODE ENDS
        END START
```

请选择正确的答案填入空格中：

- (1) A) 8AH B) 0BH
 C) D7H D) 05H
(2) A) 8AH B) 0BH

- C) D7H D) 05H
(3) A) 8AH B) 0BH
 C) D7H D) 05H
(4) A) 8AH B) 0BH
 C) D7H D) 05H
(5) A) 3 B) 2
 C) 1 D) 4
(6) A) CLC B) CLD
 C) STC D) CLI
(7) A) 1 B) -1
 C) 0 D) 0FFH

四、完成下列练习

1、 设(DS)=0300H, (CS)=0200H, (IP)=0100H, (BX)=1000H, DISPLACE=2000H, (4000H)=5000H, (6000H)=8000H
确定无条件转移指令的段内转移地址

- (1) JMP BX
(2) JMP [BX]
(3) JMP DISPLACE[BX]

2、 设 AX 和 BX 寄存器中是有符号数, CX 和 DX 寄存器中是无符号数, 写出下列条件转移指令。

- (1) CX 值高于或等于 DX 值, 则转移 HIEQU
(2) AX 值大于 BX 值, 则转移 GREAT
(3) CX 值为 0, 则转移 ZERO
(4) AX 值小于或等于 BX 值, 则转移 SMAEQU
(5) CX 值低于或等于 DX 值, 则转移 LOWEQU
(6) AX 为负值, 则转移 NEGAT
(7) AX 为偶数, 则转移 IMAGE

3、 写出下列串操作指令所替代的指令序列。

- (1) REP MOVSB
(2) REP LODSB
(3) REP STOSB
(4) REP SCASB

五、是非判断题

程序在数据段中的定义如下:

```
DATA      SEGMENT
NAMES     DB  'ABCD'
NAMET     DW  20
DATA      ENDS
```

请判断下列指令是否正确? 正确的填 'A', 错误的填 'B'。

- (1) MOV BX, OFFSET NAMES ()
(2) MOV AL, [BX+2] ()
(3) MOV AX, NAMES ()
(4) MOV AX, WORD PTR NAMES+1 ()

- (5) MOV AL, NAMET ()
- (6) MOV AL, BYTE PTR NAMET ()
- (7) MOV AX, DATA ()
- (8) MOV DS, AX ()
- (9) MOV AX, SEG NAMES ()
- (10) MOV BX, DATA ()

六、请阅读下列程序，分析其功能或结果

- 1、 MOV AH, 00
INT 16H
MOV AH, 0AH
MOV CX, 00
MOV BH, 00
INT 10H
- 2、 MOV AH, 00
MOV AL, 66
MOV DX, 00
INT 17H
MOV AH, 00
MOV AL, 0DH
INT 17H
- 3、 MOV AH, 07
MOV AL, 00
MOV BL, 70H
MOV CH, 02
MOV CL, 05
MOV DH, 22
MOV DL, 38
INT 10H
- 4、 MESS DB 'PRESS ANY KEY', 0DH, 0AH, '\$'
MOV AH, 09
MOV DX, SEG MESS
MOV DS, DX
MOV DX, OFFSET MESS
INT 21H
- 5、 MOV AH, 2
MOV DL, 1
INT 21H

七、编写程序

- 1. 试编写一个汇编语言程序，要求对键盘输入的小写字母用大写字母显示出来。
- 2. 编写程序，比较两个字符串 STRING1 和 STRING2 所含字符是否完全相同，若相同则显示“MATCH”，若不同则显示“NO MATCH”。
- 3. 试编写程序，要求从键盘输入 3 个 16 进制数，并根据对 3 个数的比较显示如下信息：
 - (1) 如果 3 个数都不相等则显示 0；
 - (2) 如果 3 个数中有 2 个数相等则显示 2；

- (3) 如果 3 个数都相等则显示 3。
4. 已知整数变量 A 和 B，试编写完成下述操作的程序：
- (1) 若两个数中有一个是奇数，则将该奇数存入 A 中，偶数存入 B 中；
 - (2) 若两个数均为奇数，则两数分别加 1，并存回原变量；
 - (3) 若两个数均为偶数，则两变量不变。
5. 把 0~10010 之间的 30 个数，存入首地址为 GRAD 的数组中，GRAD+i 表示学号为 i+1 的学生成绩。另一个数组 RANK 是 30 个学生的名次表，其中 RANK+i 的内容是学号为 i+1 的学生的名次。试编写程序，根据 GRAD 中的学生成绩，将排列的名次填入 RANK 数组中（提示：一个学生的名次等于成绩高于这个学生的人数加 1）。
6. 试编写程序，它轮流测试两个设备的状态寄存器，只要一个状态寄存器的第 0 位为 1，则与其相应的设备就输入一个字符；如果其中任一状态寄存器的第 3 位为 1，则整个输入过程结束。两个状态寄存器的端口地址分别是 0024 和 0036，与其相应的数据输入寄存器的端口则为 0026 和 0038，输入字符分别存入首地址为 BUFF1 和 BUFF2 的存储区中。
7. 编写一个程序，接收从键盘输入的 10 个十进制数字，输入回车符则停止输入，然后将这些数字加密后（用 XLAT 指令变换）存入内存缓冲区 BUFFER。加密表为：
- | | |
|-------|------------------------------|
| 输入数字： | 0, 1, 2, 3, 4, 5, 6, 7, 8, 9 |
| 密码数字： | 7, 5, 9, 1, 3, 6, 8, 0, 2, 4 |

答案

一、回答下列问题

1、(1) C9H

(2) 1FH

(3) 0FH

2、答案:

```
DAT_SEG SEGMENT
    HIGH_DAT DB 95
DAT_SEG ENDS
;
STK_SEG SEGMENT
    DW 64 DUP(?)
STK_SEG ENDS
;
COD_SEG SEGMENT
MAIN PROC FAR
ASSUME CS: COD_SEG, DS: DAT_SEG, SS: STK_SEG
START: MOV AX, DAT_SEG
        MOV DS, AX
        MOV AH, HIGH_DAT
        MOV BH, AH
        MOV DL, AH
        MOV AH, 4CH
        INT 21H
MAIN ENDP
COD_SEG ENDS
```

END START

3、答案：

DATA SEGMENT

TRING DB 'Personal computer'

VALD DB 12, 99

VALH DB 32H, 0FAH

VALZ DB 10 DUP ('0')

VALW DW 20 DUP (?)

4、答案：

(1) 1, 表示 BUFA 所占的内存空间

(2) 20, 表示 STA 中 DUP 前重复的次数。

5、非法标号：(1)因为 '.' 只允许是标号的第一个字符

(2)第一个字符不能为数字

(3)不允许出现 '-'

(4)不能是保留字,如助记符

(5)不能有空格

6、答案：

(1) 设置了 4 个字节 (2) 设置了 2 个字节

7、答案：

MOV BX, TABLE_ADDR ; 执行后(BX)=1234H

LEA BX, TABLE_ADDR ; 执行后(BX)=OFFSET TABLE_ADDR

8、答案：

(1) 00048h

(2) 00020h

9、答案：

(1) 源和目的不能同为内存变量

(2) 两个类型不匹配

(3) LAB 为标号不为符号常量

(4) JNZ 可跳转到标号，不能跳转到变量。

二、填空题

1、1256H、3280H、2450H

2、1030H, 106CH, 10B8H

3、立即数寻址方式、直接寻址方式

4、=可以二次赋值, EQU 不能直接下赋值, 如果要二次赋值, 必须使用 PURGE 解除后才行。

5、前一个 AND 是一个命令, 后一个 AND 是一个运算符。

6、相等、零, 不相等, 不为零

7、段内直接, 段内间接, 段间直接, 段间间接

8、在同一个段内, 不在同一个段内。

9、SI, DS, DI, ES

10、ENDP, 必须与过程名, ENDS, 必须与段名

三、选择题

1、(1)、D (2)、A (3)、B (4)、C (5)、A (6)、B (7)、A

2、(1)、B (2)、A (3)、D (4)、C (5)、B (6)、A (7)、C

四、完成下列练习

1、1000H, 5000H, 8000H

2、答案：

```
(1)  CMP  CX, DX
      JNB  HIEQU
      :
      :
      HIEQU:
(2)  CMP  AX, BX
      JG   GREAT
      :
      :
      GREAT:
(3)  CMP  CX, 0
      JZ   ZERO
      :
      :
      ZERO:
(4)  CMP  AX, BX
      JLE  SMAEQU
      :
      :
      SMAEQU:
(5)  CMP  CX, DX
      JBE  LOWEQU
      :
      :
      LOWEQU:
(6)  CMP  AX, 0
      JS   NEGAT
      :
      :
      NEGAT:
(7)  TEST AX, AX
      JP   IMAGE
      :
      :
      IMAGE:
```

3、答案：

```
(1) MOV  DS, SEG SOURCE
MOV  ES, SEG DEST
MOV  SI, OFFSET SOURCE
MOV  DI, OFFEST DEST
MOV  CX, N
```

```
AGAIN:  MOV  AL, [SI]
        MOV  ES: [DI], AL
        INC SI
        INC DI
        LOOP AGAIN
```

```
(2) MOV  DS, SEG SOURCE
    MOV  SI, OFFSET SOURCE
    MOV  CX, N
AGAIN:  MOV  AL, [SI]
        INC SI
        LOOP AGAIN
```

```
(3)  MOV  ES, SEG DEST
    MOV  DI, OFFSET DEST
    MOV  CX, N
AGAIN:  MOV  ES: [DI], AL
        INC DI
        LOOP AGAIN
```

```
(4)  MOV  ES, SEG DEST
    MOV  DI, OFFSET DEST
    MOV  CX, N
AGAIN:  CMP  AL, ES: [DI]
        INC DI
        LOOP AGAIN
```

五、是非判断题

(1) A (2) B (3) B (4) A (5) B (6) A (7) A (8) A (9) A (10) A

六、请阅读下列程序，分析其功能或结果

- 1、读取键盘输入，并显示在显示器上。
- 2、在打印机一输出字符 'B'，并回车。
- 3、屏幕初始化，左上角行号为 2，左上角列号为 5，右下角行号为 22，右下角列号为 38，使用整个空白窗口。
- 4、在屏幕上输出 PRESS ANY KEY 并换行回车。
- 5、在屏幕上输出 1

七、编写程序

1、答案：

```
abc:  mov     ah,1
      int     21h
      cmp     al,'a'
      jb      stop
      cmp     al,'z'
      ja      stop
      sub     al,20h
      mov     dl,al
      mov     ah,2
      int     21h
```

```
        jmp      abc
stop: ret
2、答案:
dataarea segment
    string1 db 'asfioa'
    string2 db 'xcviyoaf'
    mess1   db 'MATCH','$'
    mess2   db 'NO MATCH','$'
dataarea ends
prognam segment
main proc far
    assume cs:prognam,ds:dataarea
start:
pushds
sub     ax,ax
pushax
mov     ax,dataarea
mov     ds,ax
mov     es,ax
begin:
    mov     cx, string2-string1
    mov     bx, mess1-string2
    cmp     bx,cx
    jnz     dispno
    lea     dx,addr
    lea     si,string1
    lea     di,string2
    repe    cmpsb
    jne     dispno
    mov     ah,9
    lea     dx,mess1
int      21h
    ret
dispno:
    mov     ah,9
    lea     dx, mess2
    int     21h
ret
main     endp
prognam  ends
        end start
```

3、答案:

```
data segment
array dw 3 dup(?)
```

```
data    ends
code    segment
main    proc far
        assume cs:code,ds:data
start:
pushds
sub     ax,ax
pushax
mov     ax,data
mov     ds,ax
mov     cx,3
lea     si,array
begin:
        pushcx
        mov     cl,4
        mov     di,4
        mov     dl,' '
        mov     ah,02
        int     21h
        mov     dx,0
input:
        mov     ah,01
int      21h
and     al,0fh
shl     dx,cl
or      dl,al
dec     di
jne     input
mov     [si],dx
add     si,2
pop     cx
loop    begin
comp:
        lea     si,array
        mov     dl,0
        mov     ax,[si]
        mov     bx,[si+2]
        cmp     ax,bx
        jne     next1
        add     dl,2
next1:
        cmp     [si+4],ax
        jne     next2
        add     dx,2
```

```
next2:
    cmp    [si+4],bx
    jne     num
    add     dl,2
num:
    cmp     dx,3
    jl      disp
    mov     dl,3
disp:
    mov     ah,2
    add     dl,30h
    int     21h
    ret
main      endp
code ends
                end        start
```

4、答案:

```
dseg segment
    a    dw  ?
    b    dw  ?
dseg ends
cseg      segment
mainproc  far
assume    cs:cseg,ds:dseg
start:push ds
        sub     ax,ax
        push ax
        mov     ax,dseg
        mov     ds,ax
begin:   mov     ax,a
        mov     bx,b
        xor     ax,bx
        test    ax,0001
        jz      class
        test    bx,0001
        jz      exit
        xchgbx,a
        mov     b,bx
        jmp     exit
class:   test    bx,0001
        jz      exit
        inc     b
        inc     a
exit: ret
```

```
mainendp
cseg      ends
          end      start
5、 答案:
dseg segment
grade     dw  30 dup(?)
rank      dw  30 dup(?)
dseg ends
cseg      segment
mainproc  far
assume    cs:cseg, ds:dseg, es:dseg
start:push ds
          sub      ax,ax
          pushax
          mov      ax,dseg
          mov      ds,ax
          mov      es,ax
begin:    mov      di,0
          mov      cx,30
loop1:    pushcx
          mov      cx,30
          mov      si,0
          mov      ax,grade[di]
          mov      dx,0
loop2:    cmp      grade[si],ax
          jbe      go_on
          inc      dx
go_on:    add      si,2
          loop     loop2
          pop      cx
          inc      dx
          mov      rank[di],dx
          add      di,2
          loop     loop1
          ret
mainendp
cseg      ends
          end      start
```

```
6、 答案:
          mov      si, 0
          mov      di, 0
test12:   in       al, 0024h
          test     al, 08
          jnz      exit
```

```
        in     al,0036h
        test   al, 08
        jnz    exit
dev1:    in     al, 0024h
        test   al, 01
        jz     dev2
        in     al, 0026h
        mov    buffer[si], al
        inc    si
dev2:    in     al, 0036h
test     al, 01
jz       test12
in       al, 0038h
mov      buff2[di],al
inc      di
jmp      test12
exit: ret
```

7、答案:

```
scode    db    7,5,9,1,3,6,8,0,2,4
buffer   db    10 dup(?)
; ...    ...
        mov     si,0
        mov     cx,10
        lea     bx,scode
input:   mov     ah,01
        int     21h
        cmp     al,0ah
        jz      exit
        and     al,0fh
        xlat
        mov     buffer[si],al
        inc     si
        loop    input
exit: ret
```