E07：连续分配、页式分配及虚拟存储器

参考答案与说明

1. C
   【说明】：在固定分区的系统中，程序的重定位采用静态重定位，即程序的重定位是在程序装入内存时进行的，由装入程序完成。
2. C
3. A
4. A
5. A
6. D
7. B
8. D
9. ①固定分区　　②可变分区
   【说明】固定式分区就是把内存固定地划分为若干个大小不等的区域。与固定式分区相比，可变分区在作业执行前并不建立分区，分区的建立是在作业的处理过程中进行的，且其大小可随作业或进程对内存的要求而改变。

10. ①静态　　②动态
    【说明】采用固定分区时，由于不存在程序在内存的移动问题，因此通常采用静态重定位法；可变分区，一般要采用动态重定位法，其原因是由于可变分区系统需要有移动技术作为支持，以解决可变分区的存储碎片问题。

11. ①首次适应算法　②循环首次适应算法　③最佳适应算法　　④最坏适应算法
    【说明】四种不同的算法要求空闲区表按不同的方式排列。首次适应和循环首次适应算法要求空闲区表按空闲区的起始地址递增的次序排列；最佳适应算法要求空闲区表按空闲区尺寸从小到大排列；最坏适应算法要求空闲区按其大小递减的顺序组成空闲区表。

12. ①最坏适应算法
13. 错误
    【说明】分区分配所用的几种算法各有其特点，针对不同的请求队列，它们的效率和功能是不一样的。“最坏适应算法”是基于不留下碎片空闲区这一点出发的，它选择最大的空闲区来满足用户要求，以期分配后的剩余部分仍能进行再分配。

14. 正确
    【说明】这是为了适应程序在内存中移动，以解决内存碎片问题。
15. 正确
    【说明】单用户、单任务系统中，系统中的所有资源被一个任务独占。
16. 错误
    【说明】存储管理中“地址重定位”是指将程序中的相对地址转换为绝对地址的过程，分为“静态重定位”和“动态重定位”。

17. C
    【说明】：硬件自动把地址空间的地址分为页号和页内相对地址，通过页号在页表找到内存中的对应块号，内存的物理地址用下列公式确定：

    内存的物理地址 = 块号×每一页的字节数 + 页内相对地址

18. ①：B；　　②：E；　　③：G
    【说明】在页式管理中，作业的地址空间被分为页，而内存空间也被分为与页大小相等的块。页号和块号的对应是通过页表实现的。地址空间中的地址被地址变换机构自动分解为页号和页内相对地址，然后根据页号查找页表找到对应的块号：

块号×1024（即 IKB）+ 页内相对地址 = 内存空间的物理地址

19. D

【说明】页表一般是存放在内存中的，即划分某些内存区域存放页表，而它的起始地址是存放在专门的寄存器中以便地址转换机构能快速找到页表，这个寄存器称为页表始址寄存器。

20. A

【说明】在采用页式存贮管理系统中，页框的大小应选 2 的整次幂，这样可以加快地址转换速度。

21. C

22. A

23. C

24. ①页框（块）　　②页

【说明】页式管理中，页长的划分和内存外存之间数据传输速度以及内存大小等有关，一般每页长大约为 1~4KB。经过划分之后，进程的虚地址变为由页号 p 与页内地址 d 所组成。内存空间划分成与页相等的片或块后，用户进程在内存空间内除了在每个页内地址连续之外，每个页面之间不再连续。这样，不仅实现了内存中碎片的减少，而且实现了由连续存储到非连续存储的飞跃。

25. ①页表　　②地址变换机构

【说明】页表也称为页面映像表，最简单的页表由页号和内存块号组成。要实现页式虚地址到内存物理地址的转换，除了页表外，还需要其他的硬件支持，如：页表始址和页表长度寄存器等。

26. ①内存　　②2

【说明】由于页表放在内存中，一次访问页表以确定所取数据或指令的物理地址，另一次是根据地址取数据或指令。

27. ①页号和块号

28. 错误。

【说明】地址越界保护可由地址变换机构对“页表长度”和所要访问的虚地址的“页号”相比较完成，当要访问的虚地址的“页号”大于“页表长度”时发生越界中断。

29. 错误

【说明】页式存贮管理中地址空间是一维的，其页的划分对用户是透明的。

30. 正确。

【说明】系统抖动时，内存外存之间交换频繁，访问外存的时间和输入、输出处理的时间大大增大，造成 CPU 因等待数据空转，从而使系统效率急剧下降，这是虚拟存储系统中特有的一种现象。

31. 【参考答案】依题目所给条件，已知位示图如下所示：

	0	1	2	…		31
0	0	1	2	…		31
1	32	33	34	…		
2						
255				…		8191

4999÷32=156，余 7。所以 4999 块对应的字号为 156，位号为 7。

129 字的第 29 位对应的块号为：129 * 32 + 29 = 4157，即对应内存的第 4157 块。

32. 【参考答案】因为有 32 个页面：虚地址中高 5 位为页号；由于有 1KB 页长，所以虚地址中低 10 位为页内地址。

0A5C = 000 1010 0101 1100　虚页号为 2，物理页号为 4

000 1010 0101 1100 ⇨ 001 0010 0101 1100 =125C
0D3C = 000 1101 0011 1100　虚页号为 3，物理页号为 7
000 1101 0011 1100 ⇨ 001 1101 0011 1100 =1D3C

33. 【参考答案】(1) 由于主存容量 1M，所以主存地址需 20 位。(2) 1M 空间被分为 256 块，每一块的大小为 4K。所以虚地址中每一页的长度同样为 4K 字节，则页内地址应占用 12 位。(3) 如下表所示。(4) 略。

页号	起始地址
0	8192
1	16384
2	4096
3	20480

34. 【参考答案】因为物理地址为 20 位，所以该系统的内存空间大小为 1M，每块大小与页面大小相同，也为 1KB。逻辑地址为 16 位，其中，也好 6 位，页内偏移地址 10 位。
0420H = 0000 0100 0010 0000　虚页号为 1，物理页号为 7，则物理地址为：
0000 0100 0010 0000 ⇨ 0001 1100 0010 0000 = 1C20H

35. A
36. B
【说明】：有的也将 LRU 称为最近最久未使用页面的置换算法，根据一个作业在执行过程中过去的页面踪迹来推测未来的行为。该算法的思想是当需要淘汰一页时，选择离当前时间最近的一段时间最久没有使用过的页先淘汰，它认为过去一段时间里不曾被访问过的页，在最近的将来可能也不再会被访问。
37. B
【说明】M=4 时，采用 LRU 算法，系统的淘汰过程：

	4	3	2	1	4	3	5	4	3	2	1	5
是否缺页：	*	*	*	*			*			*	*	*
	4	4	4	4	4	4	4	4	4	4	4	5
内存中包含的		3	3	3	3	3	3	3	3	3	3	3
页面：			2	2	2	2	5	5	5	5	1	1
				1	1	1	1	1	1	2	2	2
被淘汰的页：							2			1	5	4

即 F=8（次缺页）
M=3 时，采用 LRU 算法，系统的淘汰过程：

	4	3	2	1	4	3	5	4	3	2	1	5
是否缺页：	*	*	*	*	*	*	*			*	*	*
	4	4	4	1	1	1	5	5	5	2	2	2
内存中包含		3	3	3	4	4	4	4	4	4	1	1
的页面：			2	2	2	3	3	3	3	3	3	5
被淘汰的页：				4	3	2	1			5	4	3

即 F=10（次缺页）
38. B
【说明】在其他几个因素确定的情况下，不同的置换算法其缺页率是不同的。
39. C
【说明】请求页式管理的原理是：当执行某条指令而又发现它不在内存时，或当执行某条指令需访问其他的数据和指令时，这些指令和数据不在内存中，从而发生缺页中断，系统将外存中相应的页面调入内存。
40. D
41. B　C

【说明】在请求页式管理中，发生缺页中断时如果内存有空闲的页面，就将当前要访问的页放入内存的空闲页中；如果此时内存满了，就应该根据置换算法淘汰页，然后将当前要访问的页放入内存。因此选项（A）错误。在单用户、单任务系统中，系统中的所有资源被一个任务独占。因此选项（B）正确。请求页式管理可以实现虚拟存储，因此选项（C）正确。置换算法是为了解决当发生缺页中断时，内存如果满了，按什么原则淘汰内存中的某一页。因此选项（D）错误。页式管理中，地址越界保护可由地址变换机构对“页表长度”和所要访问的虚地址的“页号”相比较完成，当要访问的虚地址的“页号”大于“页表长度”时发生越界中断。因此选项（E）错误。因此，正确的是 B、C。

42. ①页面置换

43. ①判断某页是否在内存

【说明】请求页式管理中，要解决的两个根本问题是：如何发现不在内存中的虚页以及如何处理。第一个问题通过在页表中增加页是否在内存的“标志位”和该页在“外存始址”可以解决。关于虚页不在内存的处理，涉及两个问题，第一，采用何种方式把缺的页调入内存；第二，如果内存中没有空闲的页面时，把调入的页放在哪里。也就是说，采用什么策略来淘汰已占据内存的页。如果选中某页应淘汰，而该页又因程序的执行被修改过，显然该页应被重新写到外存中加以保存。如果该页未被修改过，外存已保留有相同的副本，写回外存就没有必要，所以增加“改变位”来表征某页是否被修改。

44. ①13　　②14　　③14　　④12

45. 正确

【说明】标志位说明某页是否在内存中，磁盘地址位说明某页在磁盘上的地址，使得发生缺页中断时，系统能够在磁盘中得到该页。

46. 正确

47. 错误

【说明】置换算法是为了解决当发生缺页中断时内存如果满了，按什么原则淘汰内存中的某一页。

48. 【参考答案】

M=4 时，采用 LRU 算法，系统的淘汰过程：

	7	0	1	2	0	3	0	4	2	3	0	3	2	1	2	0	1	7	0
是否缺页：	*	*	*	*		*		*						*				*	
内存中包含页面：	7	7	7	7	7	3	3	3	3	3	3	3	3	3	3	3	3	7	7
		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
			1	1	1	1	1	4	4	4	4	4	4	1	1	1	1	1	1
				2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
被淘汰的页：						7		1						4			4	3	

即 F=8（次缺页）。

M=3 时，采用 LRU 算法，系统的淘汰过程：

	7	0	1	2	0	3	0	4	2	3	0	3	2	1	2	0	1	7	0
是否缺页	*	*	*	*		*		*	*	*	*			*		*		*	
内存中包含页	7	7	7	2	2	2	2	4	4	4	0	0	0	1	1	1	1	1	1
		0	0	0	0	0	0	0	0	3	3	3	3	3	3	0	0	0	0
			1	1	1	3	3	3	2	2	2	2	2	2	2	2	2	7	7
被淘汰的页				7		1		2	3	0	4			0		3		2	

即 F=12（次缺页）。

M=4 时，采用 FIFO 算法，系统的淘汰过程：

	7	0	1	2	0	3	0	4	2	3	0	3	2	1	2	0	1	7	0
是否缺页：	*	*	*	*		*		*			*			*	*			*	
内存中包含页面：	7	0	1	2	2	3	3	4	4	4	0	0	0	1	2	2	2	7	7
		7	0	1	1	2	2	3	3	3	4	4	4	0	1	1	1	2	2
			7	0	0	1	1	2	2	2	3	3	3	4	0	0	0	1	1
				7	7	0	0	1	1	1	2	2	2	3	4	4	4	0	0
被淘汰的页：						7		0			1			2	3			4	

即 F=10（次缺页）

M=3 时，采用 FIFO 算法，系统的淘汰过程：

	7	0	1	2	0	3	0	4	2	3	0	3	2	1	2	0	1	7	0
是否缺页	*	*	*	*		*	*	*	*	*	*			*	*			*	*
内存中包含页	7	0	1	2	2	3	0	4	2	3	0	0	0	1	2	2	2	7	0
		7	0	1	1	2	3	0	4	2	3	3	3	0	1	1	1	2	7
			7	0	0	1	2	3	0	4	2	2	2	3	0	0	0	1	2
被淘汰的页				7		0	1	2	3	0	4			2	3			0	1

即 F=14（次缺页）

49. 【参考答案】采用代码（1）其访问顺序与数组存放顺序一致，由于第一页已在内存中，所以除了访问第一页时不发生缺页，对其余 127 页的访问均发生缺页，所以共发生 128-1 次缺页中断。采用代码（2）其访问顺序是按列访问，与数组存放顺序不一致，经分析可知共发生 128*128-1 次缺页中断。

50. 【参考答案】

（1）FIFO 共发生 9 次缺页

1	2	1	0	4	1	3	4	2	1	4	1
*	*		*	*	*	*		*		*	*
1	2	2	0	4	1	3	3	2	2	4	1
	1	1	2	0	4	1	1	3	3	2	4
			1	2	0	4	4	1	1	3	2
				1	2	0		4		1	3

依次被淘汰的页为：1、2、0、4、1、3

LRU 共发生 7 次缺页

1	2	1	0	4	1	3	4	2	1	4	1
*	*		*	*		*		*	*		
1	1	1	1	1	1	1	1	2	2	2	2
	2	2	2	4	4	4	4	4	4	4	4
			0	0	0	3	3	3	1	1	1
				2		0		1	3		

依次被淘汰的页为：2、0、1、3

（2）0A4EH 对应的二进制为：0000 10,10 0100 1110，该地址表明它对应第 2 页，根据已知该页在内存，对应物理块为 10，所以，物理地址为：0010 10,10 0100 1110（十六进制为 2A4EH）。122AH 对应的二进制为：0001 00,10 0010 1010，该地址表明它对应第 4 页，根据已知该页不在内存中。