1. Our 32-bit uniprocessor machine has 1 GiB of RAM with 1 KiB pages, a fully-associative TLB that holds 8 entries and uses LRU.

a) What is the maximum number of valid entries in the page table for a single process? Answer in IEC.

Page table valid entries set by size of PM. 1 GiB / 1 KiB = 1 Mi-entries.

+0.5 points for 1MiB, +0.5 points for 2^{20}

b) What is the TLB Reach of our system?

8 TLB entries that refer to a 1 KiB page each. TLB Reach = 8*1KiB. No partial credit – full credit given for 2^{13} B._1 Mi-entries_ 8 KiB

___ ___

2. The system in question has 1MiB of physical memory, 32-bit virtual addresses, and 256 physical
pages. The memory management system uses a fully associative TLB with 128 entries and an LRU
replacement scheme.

a. What is the size of the physical pages in bytes?

物理页数256=2^8，内存1MB=2^20,2^20/2^7=2^12(页大小)

2^12 bytes

b. What is the size of the virtual pages in bytes?

2^12 bytes

c. What is the maximum number of virtual pages a process can use?

32位处理器，虚拟地址32位，2^32/2^12=2^20 pages

d. What is the minimum number of bits required for the page table base address register?

页表基址寄存器存储页表首地址，页表放在内存，内存地址一共20位，故需

20 bits

Everybody Got Choices

e. Answer “Yup!” (True) or “Nope!” (False) to the following questions

i. The page table is stored in main memory **Yup!**

ii. Every virtual page is mapped to a physical page **Nope!**

iii. The TLB is checked before the page table **Yup!**

iv. The penalty for a page fault is about the same as the penalty for a cache miss **Nope!**

v. A linear page table takes up more memory as the process uses more memory **Nope!**

3. For the following questions, assume the following:

- 16-bit virtual addresses
- 1 KiB pages
- 512 KiB of physical memory with LRU page replacement policy
- Fully associative TLB with 16 entries and an LRU replacement policy

a) How many virtual pages are there per process? **64** ________

b) How many bits wide is the *page table base register*? 19

4.某计算机系统有一个TLB和一个L1 Data Cache。该系统按字节编址，虚拟地址16位，物理地址14位，页大小为128B，TLB采用四路组相联方式，共16个页表项， L1 Data Cache采用直接映射方式，块大小为4B，共16行。系统运行到某一时刻时，TLB、页表和L1 Data Cache中部分内容如图示。请问:

(1)虚拟地址中哪几位表示虚拟页号？哪几位表示页内偏移？虚拟页号中哪几位表示TLB标记？哪几位表示TLB索引?

(2)物理地址中哪几位表示物理页号？哪几位表示页内偏移?

(3)主存物理地址如何划分标记字段、行索引字段和块地址字段?

(4)CPU从地址06FAH中取出的值是多少？说明CPU读取地址06FAH中内容的过程。

TLB

行号	标记	页框号	有效位	标记	页框号	有效位	标记	页框号	有效位	标记	页框号	有效位
0	03		0	09	0D	1	00		0	07	02	1
1	03	2D	1	02		0	04		0	0A		0
2	02		0	08		0	06		0	03		0
3	07		0	63	0D	1	0A	34	1	72		0

数据缓存 CACHE

行索引	标记	有效位	字节 3	字节 2	字节 1	字节 0
00	19	0	12	56	C9	AC
01	15	1				
02	1B	0	03	45	12	CD
03	36	0				
04	32	1	23	34	C2	2A
05	0D	1	46	67	23	3D
06		0				
07	16	1	12	54	65	DC
08	24	1	23	62	12	3A
09	2D	0				
0A	2D	1	43	62	23	C3
0B		0				
0C	12	1	76	83	21	35
0D	16	1	A3	F4	23	11
0E	5B	1	3D	5A	45	55
0F	14	0				

页表

虚页号	页框号	有效位
00	08	1
01	03	1
02	14	1
03	02	1
04		0
05	16	1
06		0
07	07	1
08	13	1
09	17	1
0A	09	1
0B		0
0C	19	1
0D	2D	1
0E	11	1
0F	0D	1

解：（1）由已知虚拟地址16位，页大小为128B，有虚拟地址中第15~7位表示虚拟页号，第6~0位表示页内偏移。

由于TLB采用四路组相联方式，共16个页表项，4组

所以虚拟页号中第15~9位表示TLB标记，第8~7位表示TLB索引。

（2）由已知物理地址14位，页大小为128B，有物理地址中第13~7位表示物理页号，第6~0位表示页内偏移。

（3）由于Cache采用直接映射方式，块大小为4B，共16行
所以主存物理地址第 11~6 位为标记字段、第 5~2 位为行索引字段和第 1~0 位为块地址字段。

虚拟地址06FAH

=0000011 01 1111010B

a)查TLB：1行，标记3，TLB Hit；查实页号2D

物理地址 0101101 1111010 16FA

00101101 1 1110 10在缓存的E行，标记5B

查到数据 3D5A4555