1 Warm up. Explain the purpose of this code.

```
01   void acquire(pmt*mutexA, pmt*mutexB){
02     assert(mutexA&&mutexB && mutexA != mutexB);
03     if( mutexA < mutexB ) p_m_lock(mutexA);
04     p_m_lock(mutexB);
05     if( mutexB < mutexA) p_m_lock(mutexA);
06   }
```

2. Virtual Memory Addressing

A Running Process: My expensive and useful RAM:

| Argv, Env |
| Stack |
| ~ |
| ~ |
| Heap |
| Heap |
| Heap |
| Globals |
| Program |
| Program |

| ... |
| 4096 bytes |
| 4096 bytes |
| 4096 bytes |
| 4096 bytes |
| 4096 bytes |
| 4096 bytes |
| 4096 bytes |
| 4096 bytes |
| 4096 bytes |
| 4096 bytes |
| 4096 bytes |
| 4096 bytes |
| ... |

32 bit address:

[ ]

3. What is a page table? How is it used?

20 bits
12 bits

4. For a 32 bit system if each page is 4KB. How many entries are there? How many bytes are required to store the entire page table?

5. Using the above page design, how many memory reads are required to read a byte at address 0x200 ?

6. Using the two-level page table with page size 4KB, how many memory reads are required to read a byte at address 0x200 ?

7. What is an TLB? Why is it useful? How is it used?

How much memory does a 2 level page table require (assume data segment requirement requires 2 pages and the stack requires 2 pages; assume each entry in the directory contains 1024 page entries and each entry is 4 bytes)

20 bits
12 bits

Why might a page be missing in memory? Where can it be found?

| Argv,, Env |
| Stack |
| ~ |
| ~ |
| Heap |
| Heap |
| Heap |
| Globals |
| Program |
| Program |

...
4096 bytes
4096 bytes
4096 bytes
4096 bytes
4096 bytes
4096 bytes
4096 bytes
4096 bytes
4096 bytes
4096 bytes
4096 bytes
4096 bytes
4096 bytes
...

What is the dirty bit? Why is it useful?

What else can we store about each page?

Can two processes shared the same piece of RAM?

Why is this useful?

Practice Interview Questions:

What is IPC? Give some examples and explain how they work!

Amdahl's law.

My problem takes 100 seconds to calculate an answer? 25 seconds of that is non-||izable.

How can I make it run twice as fast ?