Assignment #3

Monday, April 21, 2025 12:11 PM

1. Explain what is static and dynamic relocation and their tradeoffs (5 pts).

Static is when the OS performs a one-time change of addresses (memory address allocation) in program when loaded (load time) into memory. An OS can not move it once a program is assigned a place in memory. The disadvantage of this is 1) it slows down loading and 2) it is difficult to get it done right.

Dynamic is with the OS is able to relocate a program at runtime. The OS uses two hardware registers, base and limit, and it does this every time a process references memory. The disavantage of Dynamic is that two operations are performed on every memory access and the addition is slow.

2. Consider a system that uses 16-bit virtual addresses. For each of the following *decimal* virtual addresses, compute the *binary* virtual page number (VPN) and offset for page size of 2KB and 4KB respectively: 2049, 4096, 14577. You CANNOT omit the leading 0's in the virtual page numbers (10 pts).

2KB = 2^11 and 4 KB = 2^12
so the offset is 11 and 12 bits respectively
16 bit = 2^16 virtual addresses
so the VPN is 16 - 11 and 16 - 12
VPN of 2KB = 5 bits
VPN of 4KB = 4 bits

Binary
```
2049 =  0000100000000001
4096 =  0001000000000000
14577 = 0011100011110001
```

VPN for 2KB Offset
```
2049  = 00001 00000000001
4096  = 00010 00000000000
14577 = 00111 00011110001
```

VPN for 4KB Offset
```
2049  = 0000  100000000001
4096  = 0001  000000000000
14577 = 0011  100011110001
```

3. Given a virtual address 0110101000101010 and the (incomplete) page table below. The last bit in each page table entry is the present/absent bit, and the other bits are page frame numbers (10 pts).

(1) Can you translate this virtual address to a physical address when the page size is 4KB? If yes, please show how to translate; if no, please describe what happens.

(2) What is the total size of the physical memory?

6	011	1

3. Given a virtual address 0110101000101010 and the (incomplete) page table below. The last bit in each page table entry is the present/absent bit, and the other bits are page frame numbers (10 pts).

(1) Can you translate this virtual address to a physical address when the page size is 4KB? If yes, please show how to translate; if no, please describe what happens.

(2) What is the total size of the physical memory?

6	011	1
5	010	1
4	110	1
3	001	1
2	000	0
1	100	1
0	000	0

(1) 4KB = 2^12 offset and 0110101000101010 has 16 bits in it so the VPN is 4 bits, this means that 0110 is our VPN, and 0110 = 6 which means 011 (3) is our PFN, and we have a present/absent bit = 1, so it is currently loaded in physical memory. We get 011101000101010.

(2) Since we only have a 3 bit PFN, that means our physical memory has 2^3 = 8 page frames, so 8 page frames at 4KB each means we have 32KB of physical memory.

4. Consider a virtual memory system that uses 32-bit virtual addresses and a page size of 8KB. Assuming a single-level page table, how many bits are used for the offset and virtual page number, respectively? Assuming there are 32 bits in each page table entry, what is the size of the page table? (10 pts)

8KB = 2^13 so 13 bit offset
so 2^32/2^13 is 32 - 13 which gives us a 19 bit VPN
so we have 2^19 entries in the page table

32 bits = 32/8 = 4 bytes

so, 2^19 x 4 = 2^2 x 2^19 = 2^21 which is 2 MB

5. Given a two-level page table with 4KB pages. Assume that each level occupies 10 bits in the virtual address. What is the total size of the virtual address space? (5 pts)

4KB = 2^12 so 12 bit offset
level 1 = 2^10 bits
level 2 = 2^10 bits
so we have a 32 bit virtual address
the VPN is 2^20
so the total size is 2^20 x 2^12 = 2^32 so 4 GB

6. Consider a virtual memory system that uses 32-bit virtual addresses and a page size of 1KB. Assuming a two-level page table, where the two levels use the same number of bits in the virtual address, how many bits will be used for the page offset, PT1 and PT2, respectively? The size of the page table entries is 4 bytes for both levels. If a total of 4 entries are currently used in the top-level page table, what is the size of page tables currently needed in memory? (10 pts)

1 KB = 2^10 so 10 bit offset
2^32/2^10 = 2^22 so VPN is 22 bit
each page table gets 2^11 so PT1 and PT2 is 11 bits

Each entry is 4 bytes and there are only 4 entries in the top level
so top level PT1 is 2^11 x 2^2 = 2^13 which is 8KB
the second level PT2 is 2^13 x 2^2 = 2^15 which is 32KB
so the total memory needed is 32KB + 8KB = 40KB