CS 31 Homework 9: Paging and Virtual Memory (due TUESDAY Nov. 30, 2021)

Names and Roles:

Question 1

For each of the following IA32 instructions, indicate whether the instruction **could** cause a page fault, whether it **could** cause a cache miss, and whether it **could** cause the dirty bit in the cache to be set to 1.

(a) `movl (%eax), %ebx`

`Page fault? YES or NO | Cache miss? YES or NO | Dirty bit? YES or NO`

(b) `movl $7, %ecx`

`Page fault? YES or NO | Cache miss? YES or NO | Dirty bit? YES or NO`

(c) `movl $7, (%edx)`

`Page fault? YES or NO | Cache miss? YES or NO | Dirty bit? YES or NO`

(d) `addl -8(%eax), %ebx`

`Page fault? YES or NO | Cache miss? YES or NO | Dirty bit? YES or NO`

For the next three questions you will be tracing memory accesses in a system with the following architecture:

- 8-bit virtual addresses
- 16-byte page size
- 4 pages of physical RAM

Question 2

How many bytes of data can a single process store in:

(a) physical memory (b) virtual memory

Question 3

For each the given virtual addresses, divide the address into the `page number` and `page offset`.

`1 0 0 0 1 0 1 0` `1 0 0 0 1 1 1 1` `1 0 1 0 1 0 1 0`

Question 4

On the accompanying Page Table diagram, show the results of the following memory operations on RAM and the Page Table. Assume first-in-first-out replacement. Within each box, time should progress downward, so the first address loaded appears at the top and subsequent changes are written below. To the right of the table, label each change with number of the operation that caused it. Annotate each operation below with *hit* or *page fault* to indicate whether the data was found in physical memory. Don't forget to update the valid bits, especially when a page is kicked out!

1. `read  0 0 0 1 1 0 1 0`
2. `write 0 0 0 1 1 0 1 1`
3. `read  1 1 1 1 1 0 0 0`
4. `read  1 1 1 1 1 0 1 0`
5. `read  0 1 1 0 1 0 0 0`
6. `write 0 0 0 0 1 0 0 1`
7. `read  0 0 0 0 0 0 0 0`
8. `read  0 1 0 1 0 1 1 1`
9. `write 0 0 0 1 1 0 1 0`
10. `read  0 0 0 1 1 1 0 1`

Table 1: Page Table

index	V	frame
0	0	
1	0	
...	...	...
5	0	
6	0	
...	...	...
15	0	