1. (6 points) Consider virtual memory support through paging with a TLB. The virtual address size is 32 bits, the physical memory size is 64KB and the page size is 4KB. For the page table and a 2-entry TLB below (the given page table / TLB is the initial state before the access (a),(b),... are issued), indicate the changes to the page table and TLB, the calculated physical addresses and the effect on a memory reference (page fault *and* completion or access violation) for each of the following memory references (provided as binary address, leading zeros intentionally omitted!). Also indicate when a TLB miss occurs; you may use FIFO replacement for your TLB replacement policy (starting with the first entry).

   (a) 0000 0000 0000 0000 0010 1001 0000 0111 (read access)

   (b) 0000 0000 0000 0000 0011 0001 1100 0101 (write access)

   (c) 0000 0000 0000 0000 0001 0101 0110 1100 (write access)

   (d) 0000 0000 0000 0000 0010 0010 0000 0010 (write access)

   (e) 0000 0000 0000 0000 0000 0011 0101 0101 (read access)

   (f) 0000 0000 0000 0000 0100 0011 0000 0001 (read access)

   Page numbers are encoded in the most significant bits. The status bits (valid, dirty, read only, use) are 1 if true, 0 otherwise. If a page fault occurs, the corresponding page will be loaded into memory with write access (at the indicated frame number).

PAGE TABLE

virtual page	frame	valid	dirty	read-only	use
0	2	0	1	0	1
1	0	1	0	0	1
2	1	1	0	1	0
3	2	1	0	0	1
4	1	1	0	0	0

TLB

virtual page	frame	valid	dirty	read-only	use
2	1	1	0	1	0
1	0	1	0	0	0

a. 0000 0000 0000 0000 0010 1001 0000 0111 (read access)

Since '2' is in the TLB, there will be no changes, only use bit will be set.

TLB -

TLB

virtual page	frame	valid	dirty	read-only	use
2	1	1	0	1	**1**
1	0	1	0	0	0

Anly use bit will be set in TLB.

PAGE TABLE

virtual page	frame	valid	dirty	read-only	use
0	2	0	1	0	1
1	0	1	0	0	1
2	1	1	0	1	**1**
3	2	1	0	0	1
4	1	1	0	0	0

(b) 0000 0000 0000 0000 0011 0001 1100 0101 (write access)

Since '3' is not in the TLB, it will replace 2 by FIFO Replacement

New TLB -

TLB

virtual page	frame	valid	dirty	read-only	use
2	1	1	0	1	0
3	**2**	**1**	**0**	**0**	**1**

Use bit for '1' will be unset in the page table.

PAGE TABLE

virtual page	frame	valid	dirty	read-only	use
0	2	0	1	0	1
1	0	1	0	0	**0**
2	1	1	0	1	0
3	2	1	0	0	1
4	1	1	0	0	0

(c) 0000 0000 0000 0000 0001 0101 0110 1100 (write access)

Since '1' is not in the TLB, it will evict '2' -

TLB

virtual page	frame	valid	dirty	read-only	use
1	**0**	**0**	**0**	**0**	**1**
3	2	1	0	0	1

PAGE TABLE

virtual page	frame	valid	dirty	read-only	use
0	2	0	1	0	1
1	**0**	**0**	**0**	**0**	**1**
2	1	1	0	1	0
3	2	1	0	0	1
4	1	1	0	0	0

(d) 0000 0000 0000 0000 0010 0010 0000 0010 (write access)

Since '2' is not in the TLB, it will evict 3 -

TLB

virtual page	frame	valid	dirty	read-only	use
1	0	0	0	0	1
2	**1**	**1**	**0**	**0**	**1**

Read-only bit for 2 will be set in the TLB.

PAGE TABLE

virtual page	frame	valid	dirty	read-only	use
0	2	0	1	0	1
1	0	0	0	0	1
2	1	1	0	**0**	0
3	2	1	0	0	0
4	1	1	0	0	0

(e) 0000 0000 0000 0000 0000 0011 0101 0101 (read access)

Since '0' is not in the TLB, it will evict 1 -

TLB

virtual page	frame	valid	dirty	read-only	use
0	**2**	**0**	**0**	**0**	**1**
2	1	1	0	0	1

Dirty bit will be cleared for '0'. And use bit will be cleared for 1.

PAGE TABLE

virtual page	frame	valid	dirty	read-only	use
0	**2**	**1**	**0**	**0**	**1**
1	0	0	0	0	**0**
2	1	1	0	**0**	0
3	2	1	0	0	0
4	1	1	0	0	0

(f) 0000 0000 0000 0000 0100 0011 0000 0001 (read access)

Since '4' is not in the TLB, it will evict 2 -

TLB

virtual page	frame	valid	dirty	read-only	use
0	2	0	0	0	1
1	**0**	**0**	**0**	**0**	**1**

Page table will remain the same -

PAGE TABLE

virtual page	frame	valid	dirty	read-only	use
0	2	0	1	0	1
1	0	0	0	0	1
2	1	1	0	**0**	0
3	2	1	0	0	0
4	1	1	0	0	0