Studente	
Nome	
Cognome	
Matricola	

1.1) Si completi la mappa della memoria con i due vettori A e B e variabili C e D. (2 punti)		
Dato	Indirizzo	
	LSB	MSB
A[0]	0x0000 1000	0x0000 1007
A[511]	0x0000 1FF7	0x0000 1FFF
B[0]	0x0000 2000	0x0000 2007
B[511]	0x0000 2FF7	0x0000 2FFF
C	0x0000 3000	0x0000 3007
D	0x0000 4000	0x0000 4007

1.2) Specificare il numero e il valore dei bit di TAG e Set ID del primo ed ultimo elemento di ognuno dei vettori A e B e variabili C e D. (2 punti)		
Proprieta	Vettore	
	A	B
# Bit TAG	20	20
# Bit SetID	7	7
TAG [0]	0x0000 1	0x0000 2
SetID [0]	0x0	0x0
Tag [511]	0x0000 1	0x0000 2
SetID [511]	0x7F (127)	0x7F (127)
Proprieta	Variabile	
	C	D
TAG	0x0000 3	0x0000 4
SetID	0x0	0x0

1.3 Quanta memoria occupano complessivamente i vettori A e B? È possibile che tutti e due i vettori e variabili e variabili C e D si trovino contemporaneamente ed interamente in cache nello stesso momento? (1 punto)		
Proprieta	Vettore	
	A	B
# Bytes	4KiB	4KiB
RISPOSTA: No I due vettori e le due variabili non possono essere contenute contemporaneamente in cache. I soli due vettori possono esse		

Studente	
Nome	
Cognome	
Matricola	

2.1.Si mostri la dinamica del contenuto della cache, dello stato MESI e del valore del bit LRU durante l'esecuzione del primo loop della cache.

CORE 0								EXEC	CORE 1							
	Istruzione: sd zero,0(x12)									Istruzione: -						
	Via 0			LRU	Via 1					Via 0			LRU	Via 1		
Set ID	TAG	Data	MESI		TAG	Data	MESI		Set ID	TAG	Data	MESI		TAG	Data	MESI
0x00	0x3000	C	M	1			I		0x0			I	0			I
	Istruzione: -							Init		Istruzione: sd zero,0(x12)						
	Via 0			LRU	Via 1					Via 0			LRU	Via 1		
Set ID	TAG	Data	MESI		TAG	Data	MESI		Set ID	TAG	Data	MESI		TAG	Data	MESI
0x00	0x0000 3	C	M	1			I		0x0	0x0000 4	D	M	1			I
	Istruzione: fld f14,0(x15)									Istruzione: -						
	Via 0			LRU	Via 1					Via 0			LRU	Via 1		
Set ID	TAG	Data	MESI		TAG	Data	MESI		Set ID	TAG	Data	MESI		TAG	Data	MESI
0x00	0x0000 3	C	M	0	0x0000 1	A[0..3]	E		0x0	0x0000 4	D	M	1			I
	Istruzione: fld f15,0(x11)									Istruzione: -						
	Via 0			LRU	Via 1					Via 0			LRU	Via 1		
Set ID	TAG	Data	MESI		TAG	Data	MESI		Set ID	TAG	Data	MESI		TAG	Data	MESI
0x00	0x0000 2	B[0..3]	E	1	0x0000 1	A[0..3]	E		0x0	0x0000 4	D	M	1			I
	Istruzione: fld f15,0(x12)							Task = T0 i = 0		Istruzione: -						
	Via 0			LRU	Via 1					Via 0			LRU	Via 1		
Set ID	TAG	Data	MESI		TAG	Data	MESI		Set ID	TAG	Data	MESI		TAG	Data	MESI
0x00	0x0000 2	B[0..3]	E	0	0x0000 3	C	E		0x0	0x0000 4	D	M	1			I
	Istruzione: fsd f15,0(x12)									Istruzione: -						
	Via 0			LRU	Via 1					Via 0			LRU	Via 1		
Set ID	TAG	Data	MESI		TAG	Data	MESI		Set ID	TAG	Data	MESI		TAG	Data	MESI
0x00	0x0000 2	B[0..3]	E	0	0x0000 3	C	M		0x0	0x0000 4	D	M	1			I
	Istruzione: -									Istruzione: fld f15,0(x12)						
	Via 0			LRU	Via 1					Via 0			LRU	Via 1		
Set ID	TAG	Data	MESI		TAG	Data	MESI		Set ID	TAG	Data	MESI		TAG	Data	MESI
									0x0	0x0000 4	D	M	1			I
	Istruzione: -									Istruzione: fld f14,0(x15)						
	Via 0			LRU	Via 1			Task = T1 i = 0		Via 0			LRU	Via 1		
Set ID	TAG	Data	MESI		TAG	Data	MESI		Set ID	TAG	Data	MESI		TAG	Data	MESI
									0x0	0x0000 4	D	M	0	0x0000 1	A[0..3]	E
	Istruzione: -									Istruzione: fsd f15,0(x12)						
	Via 0			LRU	Via 1					Via 0			LRU	Via 1		
Set ID	TAG	Data	MESI		TAG	Data	MESI		Set ID	TAG	Data	MESI		TAG	Data	MESI
									0x0	0x0000 4	D	M	1	0x0000 1	A[0..3]	E

Studente	
Nome	
Cognome	
Matricola	

2.2.Si mostri il contenuto della cache (per la prima e l'ultima linea di cache), lo stato MESI e il valore del bit LRU al termine del programma. (3 punti).

CORE 0								EXEC
	Istruzione:							
	Via 0			LRU	Via 1			
Set ID	TAG	Data	MESI		TAG	Data	MESI	
0x00	0x0000 3	C	M	1	0x0000 2	B[0..3]	E	Core 0
0x01	0x0000 1	A[4..7]	S	0	0x0000 2	B[4..7]	E	
...				...				
0x7E	0x0000 1	A[504..507]	S	0	0x0000 2	B[504..507]	E	
0x7F	0x0000 1	A[508..511]	S	0	0x0000 2	B[508..511]	E	

CORE 0								EXEC
	Istruzione:							
	Via 0			LRU	Via 1			
Set ID	TAG	Data	MESI		TAG	Data	MESI	
0x0	0x0000 4	D	M	1	0x0000 1	A[0..3]	E	Core 1
0x01	0x0000 1	A[4..7]	S	1				
...				...				
0x7E	0x0000 1	A[504..507]	S	1				
0x7F	0x0000 1	A[508..511]	S	1				

Studente	
Nome	
Cognome	
Matricola	

2.3.Calcolare il numero di accessi, HIT, MISS, HIT rate, MISS rate e cicli di WB per ciascun core al termine dell'esecuzione del programma (3 punti)

Dato	Task	
	T0	**T1**
# Accessi	2049	1539
# HIT	1782	1410
# MISS	267	129
# HIT rate [%]	87%	92%
# MISS rate [%]	13%	8%
# WB	4	0

Motivare i valori ottenuti:

Durante le prime 4 iterazioni:

- nel task 0 avvengono 1 miss in scrittura su C durante l'inizializzazione, 3x4 miss in lettura corrispondenti alle letture di C, A[0..3], B[0..3] e 4 hit in scrittura di C. 4 cicli di writeback su C (17 accessi, 13 miss, 4 hit)

- nel task 1 avvengono 1 miss in scrittura su D durante l'inizializzazione, 4 hit in lettura e 4 hit in scrittura per gli accessi in lettura e scittura di D ed 1 miss + 3 hit in lettura per gli accessi in A[0..3] (13 accessi, 2 miss, 11 hit)

Nelle successive 508 iterazioni:

- nel task 0 per ciascun gruppo di 4 iterazioni si hanno 4 hit in lettura e 4 hit in scrittura corrispondenti agli accessi di C (1 ld e 1sd per iterazione), 1x2 miss in lettura e 3x2 hit in lettura per A, B. (127x16 accessi, 127x2 miss, 127x14 hit => 2032 accessi, 254 miss, 1778 hit)

- nel task 1 per ciascun gruppo di 4 iterazioni si hanno 4 hit in lettura e 4 hit in scrittura corrispondenti agli accessi di D (1 ld e 1sd per iterazione), 1 miss in lettura e 3 hit in lettura per A. A questi si aggiungono due accessi uno in lettura ed uno in scrittura che danno origine a hit. (127x12 + 2accessi, 127 miss, 127x11 + 2 hit => 1524 accessi, 127 miss, 1397 hit)

2.4.Si descriva in modo qualitativo come si modificherebbe l'analisi al punto 3 se la variabile D fosse stata allocata all'indirizzo 0x0000_3010? (2 punti)

Il cambio di indirizzo di D avrebbe come primo impatto una differente allocazione nella cache dovuta ad un diverso TAG (ora identico a C e pari a 0x0000 3) lo stesso SetId (0x00). A seguito di ciò il dato D sarebbe contenuto all'interno dello stesso blocco di cache di C.

Gli accessi in lettura e scrittura del task0 corrispondenti a C e del task1 corrispondenti a D verranno gestiti dal protocollo MESI. Al termine di ciascuna iterazione per il task0 il dato C si trova in stato modified (M) in una delle due vie della cache del core0. A seguito della prima istruzione di ld su D relativa al task1, C dovrà modificare il suo stato da modified a invalid (M->S) ed emettere un ciclo di WB per aggiornare il valore del blocco di cache corrispondente a C nei livelli superiori della gerarchia di memoria. Contestualmente il blocco di cache contenente D nella via 0 della cache del core1 si porterà nello stato mesi Shared. L'istruzione di store eseguita dal task1 nel core1 porterà la linea di cache contenente D nello stato modified (S-->M), ed invaliderà il blocco di cache contenente C nel core 0 (S->I).

Alla successiva iterazione del task0 il core0 avrà una miss in lettura su C che porterà ad una transizione dello stato mesi corrispondente allo stesso blocco di memoria nel core1 da M-->S e stato mesi S nel core0. La scrittura di C comporterà la transizione nello stato M sul