BIG DATA MANAGEMENT

CZ/CE4123

Tutorial 4
Cache Conscious Designs

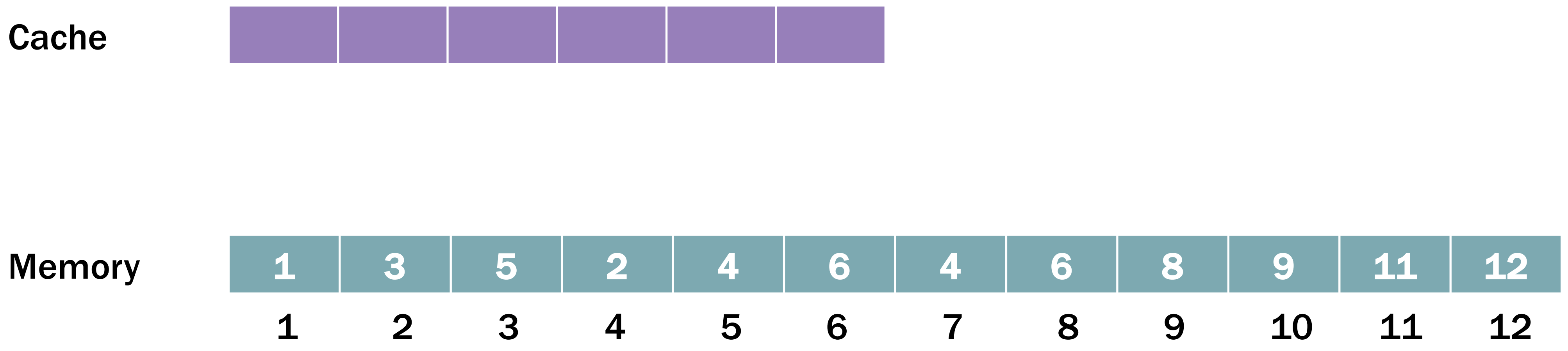
QUESTION1

We have a 12-integer array in main memory as follows
1, 3, 5, 2, 4, 6, 4, 6, 8, 9, 11, 12
Let cache size be the size of 6 integers, and cache line size (transfer unit) be the size of 3 integers.
Suppose initially the cache is empty, and there is a program sequentially accessing the whole array.
The cache replacement mechanism is the same as that in the lecture notes: i.e., first cached first evicted.

(1) After the execution of the program, what are the final values stored in the cache? Please give the cache state after every access of the array element.
(2) How many cache hits and cache misses during the program execution? Please give the hit/miss state after every access of the array element.

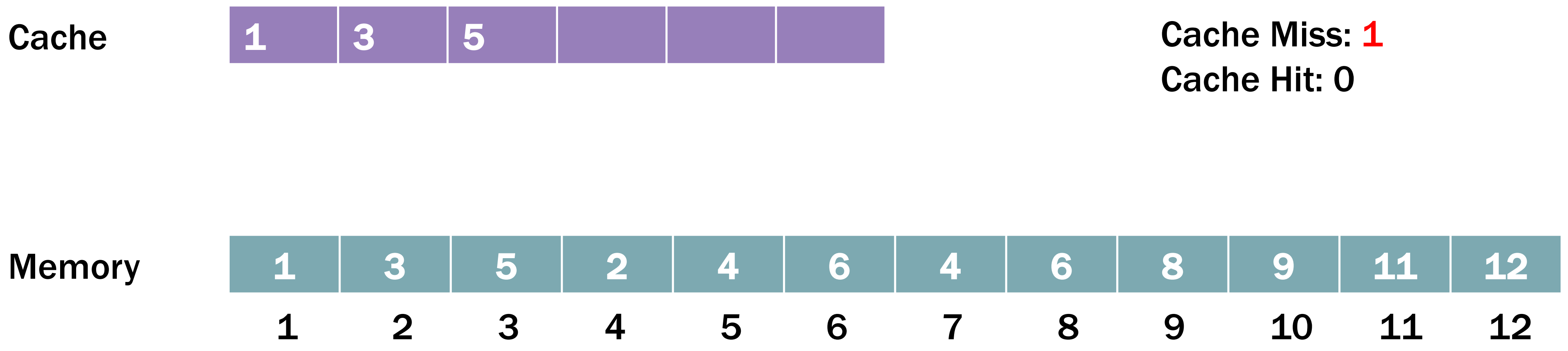
QUESTION1 SOLUTION

Access pattern: 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12



QUESTION1 SOLUTION

Access pattern: 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12



Cache Miss: 1
Cache Hit: 0

QUESTION1 SOLUTION

Access pattern: 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12

Cache

1	3	5			

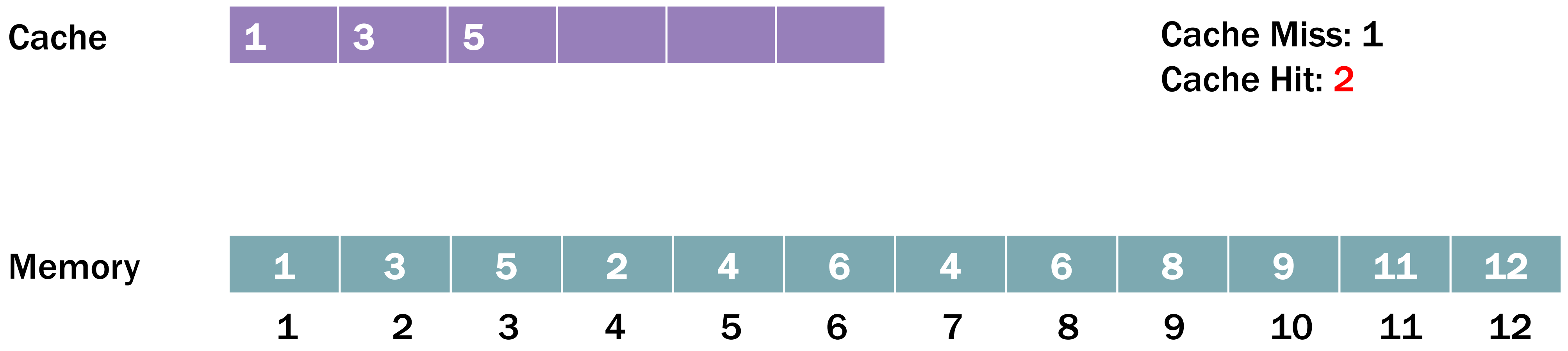
Cache Miss: 1
Cache Hit: 1

Memory

1	3	5	2	4	6	4	6	8	9	11	12
1	2	3	4	5	6	7	8	9	10	11	12

QUESTION1 SOLUTION

Access pattern: 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12



Cache Miss: 1
Cache Hit: 2

QUESTION1 SOLUTION

Access pattern: 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12

Cache

1	3	5	2	4	6

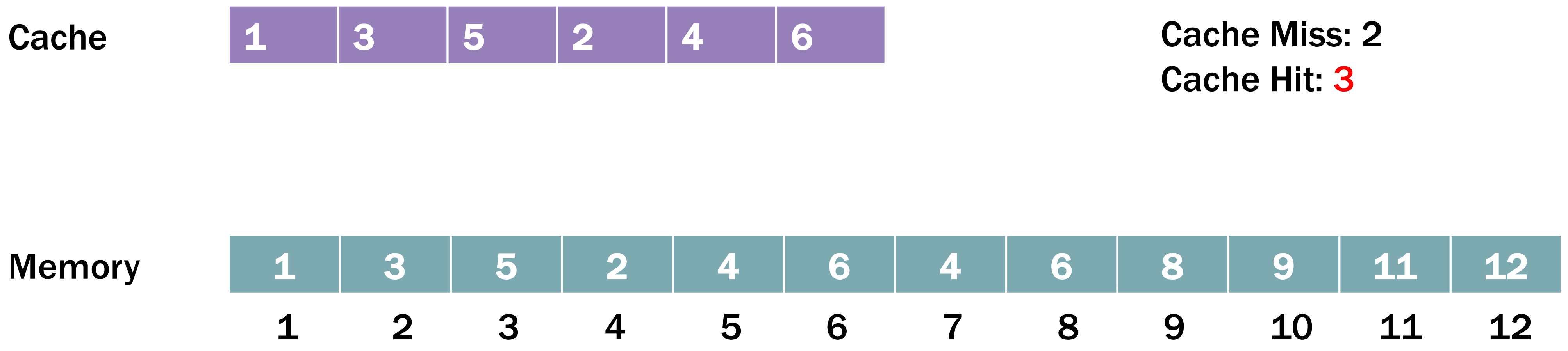
Cache Miss: 2
Cache Hit: 2

Memory

1	3	5	2	4	6	4	6	8	9	11	12
1	2	3	4	5	6	7	8	9	10	11	12

QUESTION1 SOLUTION

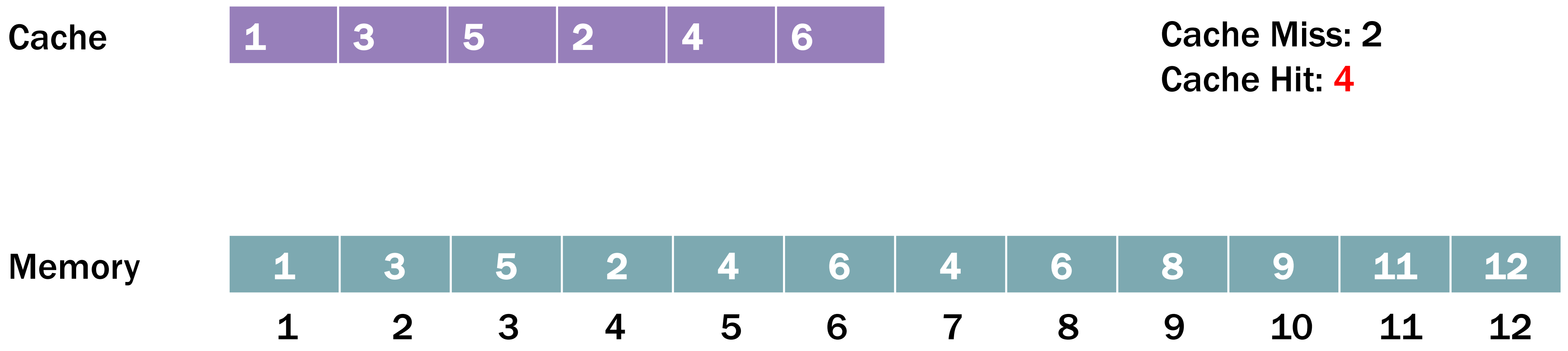
Access pattern: 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12



Cache Miss: 2
Cache Hit: 3

QUESTION1 SOLUTION

Access pattern: 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12



Cache Miss: 2
Cache Hit: 4

QUESTION1 SOLUTION

Access pattern: 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12

Cache

4	6	8	2	4	6

Cache Miss: 3
Cache Hit: 4

Memory

1	3	5	2	4	6	4	6	8	9	11	12
1	2	3	4	5	6	7	8	9	10	11	12

QUESTION1 SOLUTION

Access pattern: 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12

Cache

4	6	8	2	4	6

Cache Miss: 3
Cache Hit: 5

Memory

1	3	5	2	4	6	4	6	8	9	11	12
1	2	3	4	5	6	7	8	9	10	11	12

QUESTION1 SOLUTION

Access pattern: 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12

Cache

4	6	8	2	4	6

Cache Miss: 3
Cache Hit: 6

Memory

1	3	5	2	4	6	4	6	8	9	11	12
1	2	3	4	5	6	7	8	9	10	11	12

QUESTION1 SOLUTION

Access pattern: 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12

Cache	4	6	8	9	11	12

Cache Miss: 4
Cache Hit: 6

Memory	1	3	5	2	4	6	4	6	8	9	11	12
	1	2	3	4	5	6	7	8	9	10	11	12

QUESTION1 SOLUTION

Access pattern: 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12

Cache

4	6	8	9	11	12

Cache Miss: 4
Cache Hit: 7

Memory

1	3	5	2	4	6	4	6	8	9	11	12
1	2	3	4	5	6	7	8	9	10	11	12

QUESTION1 SOLUTION

Access pattern: 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12

Cache	4	6	8	9	11	12

Cache Miss: 4
Cache Hit: 8

Memory	1	3	5	2	4	6	4	6	8	9	11	12
	1	2	3	4	5	6	7	8	9	10	11	12

QUESTION2

Suppose we have a 2-dimentional integer array A[N][N]. We consider the two ways of array scanning: row-by-row and column-by-column. Let cache size = 5000 integers.

(1) If N>5000 and cache line size (transfer unit)=100 integers, please give a formal analysis of the cache hits/misses of the two ways of writing codes.

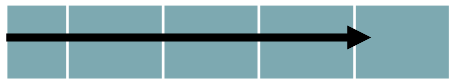
(2) If N=250 and cache line size=500 integers, please give a formal analysis of the cache hits/misses of the two ways of writing codes.

(note: for easy analysis, you can assume the 1st cache line starts from A[0][0])

SOLUTION FOR (1)

Solution X

```
for (int i=0;i<N;i++)
    for(int j=0;j<N;j++)
        A[i][j]=1;
```



Cache friendly

1 cache miss will bring 99 cache hits

SOLUTION FOR (1)

Ans:

For solution X, whenever there is a cache miss, it brings consecutive 100 integers (along the row) into the main memory, getting 99 cache hits.

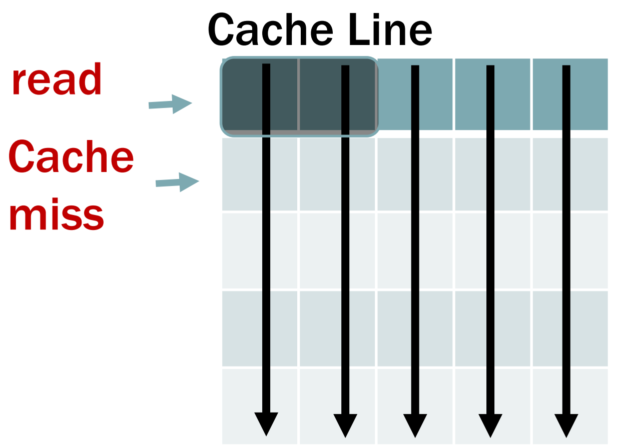
So, the overall cache misses: NxN/100

the overall cache hits: NxNx99/100

SOLUTION FOR (1)

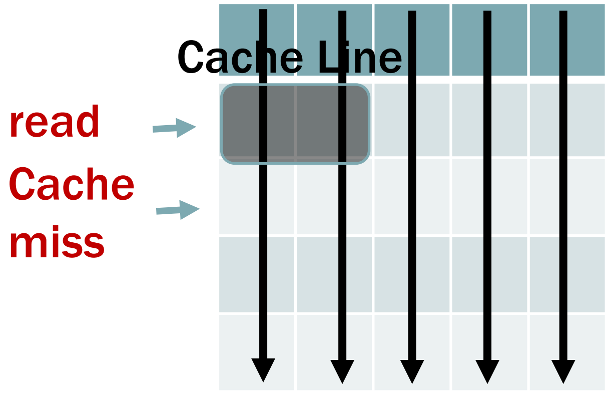
Solution Y

```
for (int j=0;j<N;j++)
      for(int i=0;i<N;i++)
            A[i][j]=1;
```



Cache unfriendly

All are cache misses



Cache unfriendly

All are cache misses

SOLUTION FOR (1)

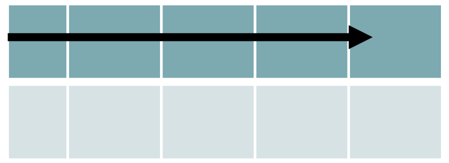
Ans:

For solution Y, since N>5000/100 (cache size/cache-line size), any cache miss at A[i][j] cannot guarantee a cache hit at A[i][j+1]. So every access incurs one cache miss.

SOLUTION FOR (2)

Solution X

```
for (int i=0;i<N;i++)
    for(int j=0;j<N;j++)
        A[i][j]=1;
```

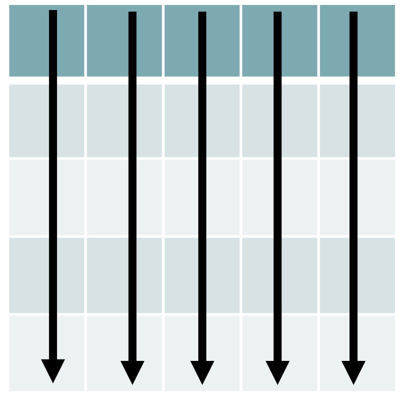


N=250

1 cache line can contain two rows.
1 miss will still bring 499 cache hits

Solution Y

```
for (int j=0;j<N;j++)
    for(int i=0;i<N;i++)
        A[i][j]=1;
```



N=250

1 cache line can contain two rows.
1 cache miss will bring 1 cache hit.

SOLUTION FOR (2)

Ans:

For solution X, the overall cache misses: NxN/500

the overall cache hits: NxNx499/500

For solution Y, since N=250>5000/250(cache size/row size), 1 cache miss will bring 1 cache hit.

So the overall cache misses: NxN/2

the overall cache hits: NxN/2

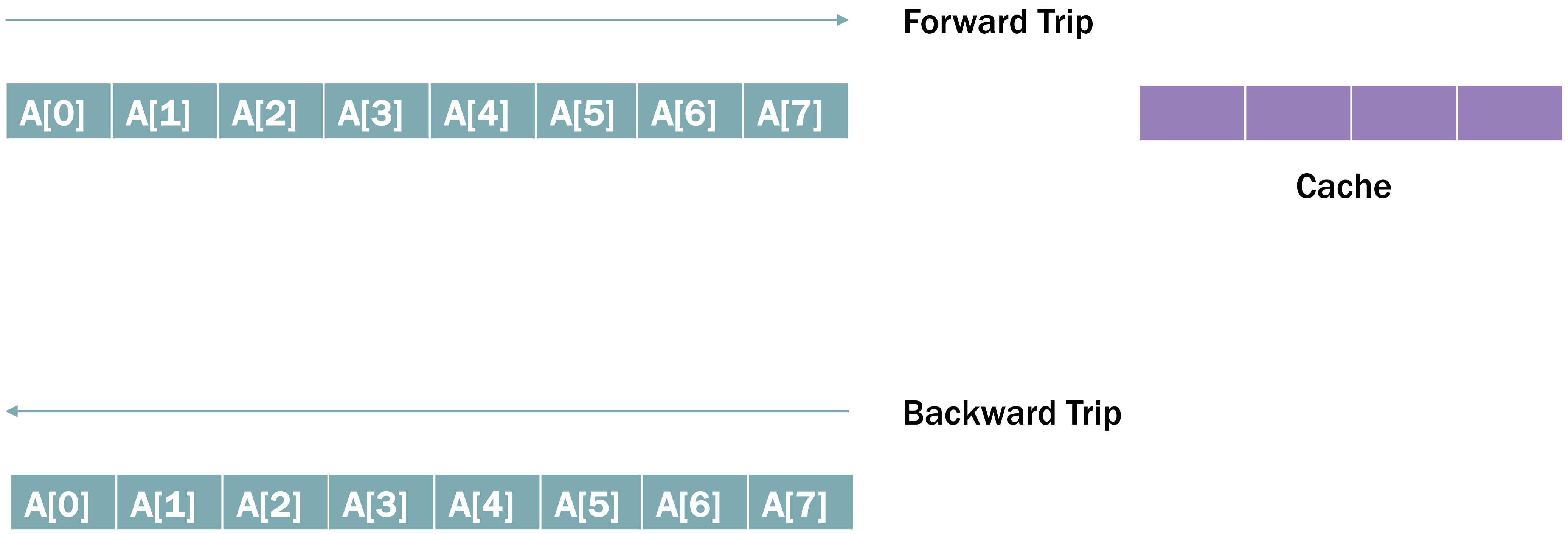
QUESTION 3

We have an 8-integer array A in the main memory. Let cache size be 4 (integers), and cache line size be 2 (integers). Suppose that initially the cache is empty, and the cache replacement policy is the same as the one introduced in the lecture notes, i.e., first cached first evicted.

Let the round-trip scanning be scanning an array in the order from the beginning to the end (i.e., A[0] to A[7]), and then from the end to the beginning (i.e., A[7] to A[0]). How many cache hits and cache misses during the round-trip scanning? Please explain your answer.

A[0]
A[1]
A[2]
A[3]
A[4]
A[5]
A[6]
A[7]

QUESTION 3


Forward Trip
A[0]
A[1]
A[2]
A[3]
A[4]
A[5]
A[6]
A[7]
Cache
Backward Trip
A[0]
A[1]
A[2]
A[3]
A[4]
A[5]
A[6]
A[7]

QUESTION 3

We have an 8-integer array A in the main memory. Let cache size be 4 (integers), and cache line size be 2 (integers). Suppose that initially the cache is empty, and the cache replacement policy is the same as the one introduced in the lecture notes, i.e., first cached first evicted.

Let the round-trip scanning be scanning an array in the order from the beginning to the end (i.e., A[0] to A[7]), and then from the end to the beginning (i.e., A[7] to A[0]). How many cache hits and cache misses during the round-trip scanning? Please explain your answer.

Ans: 10 hits, 6 misses.

1. The 1st trip (forward), one miss always brings one hit. We call it "1 miss 1 hit" procedure. So there are 4 hits and 4 misses.
2. The 2nd trip (backward), The first 4 integers of backward trip give 4 hits; the last 4 integers resume the cases of 1 miss and 1 hit.
3. In total, 10 hits and 6 misses.