#include <stdio.h>

#include <string.h>

#include <stdlib.h>

#include <stdbool.h>

#define MAX 6

int intArray[MAX];

int front = 0;

int rear = -1;

int itemCount = 0;

int peek() {

return intArray[front];

}

bool isEmpty() {

return itemCount == 0;

}

bool isFull() {

return itemCount == MAX;

}

int size() {

return itemCount;

}

void insert(int data) {

if(!isFull()) {

if(rear == MAX-1) {

rear = -1;

}

intArray[++rear] = data;

itemCount++;

}

}

int removeData() {

int data = intArray[front++];

if(front == MAX) {

front = 0;

}

itemCount--;

return data;

}

int main() {

/\* insert 5 items \*/

insert(3);

insert(5);

insert(9);

insert(1);

insert(12);

// front : 0

// rear : 4

// ------------------

// index : 0 1 2 3 4

// ------------------

// queue : 3 5 9 1 12

insert(15);

// front : 0

// rear : 5

// ---------------------

// index : 0 1 2 3 4 5

// ---------------------

// queue : 3 5 9 1 12 15

if(isFull()) {

printf("Queue is full!\n");

}

// remove one item

int num = removeData();

printf("Element removed: %d\n",num);

// front : 1

// rear : 5

// -------------------

// index : 1 2 3 4 5

// -------------------

// queue : 5 9 1 12 15

// insert more items

insert(16);

// front : 1

// rear : -1

// ----------------------

// index : 0 1 2 3 4 5

// ----------------------

// queue : 16 5 9 1 12 15

// As queue is full, elements will not be inserted.

insert(17);

insert(18);

// ----------------------

// index : 0 1 2 3 4 5

// ----------------------

// queue : 16 5 9 1 12 15

printf("Element at front: %d\n",peek());

printf("----------------------\n");

printf("index : 5 4 3 2 1 0\n");

printf("----------------------\n");

printf("Queue: ");

while(!isEmpty()) {

int n = removeData();

printf("%d ",n);

}

}