

## 3) (10 pts) ALG (Queues)

Consider the circular array implementation of a queue named Q, implemented with the structure shown below.

```
struct queue {
    int *array;
    int num_elements;
    int front;
    int capacity;
};
```

Suppose the queue is created with a capacity of 5 and front and num\_elements are initialized to 0. Trace the status of the queue by showing the valid elements in the queue and the position of front after each of the operations shown below. Indicate front by making bold the element at the front of the queue.

1. enqueue(Q, 50);
2. enqueue(Q, 34);
3. enqueue(Q, 91);
4. x = dequeue(Q);
5. enqueue(Q, 23);
6. y = dequeue(Q);
7. enqueue(Q, y);
8. enqueue(Q, 15);
9. enqueue(Q, x);
10. x = dequeue(Q);

After stmt #1:

front				
<b>50</b>				

After stmt #2:

front				
<b>50</b>	34			

After stmt #3:

front				
<b>50</b>	34	91		

After stmt #4:

front				
	<b>34</b>	91		

After stmt #5:

front				
	<b>34</b>	91	23	

After stmt #6:

front				
		<b>91</b>	23	

After stmt #7: front

		<b>91</b>	23	34
--	--	-----------	----	----

After stmt #8:

front				
15		<b>91</b>	23	34

After stmt #9:

front				
15	50	<b>91</b>	23	34

After stmt #10:

front				
15	50		<b>23</b>	34

**Grading: 1 pt per array, must be perfectly correct to get the point.**