

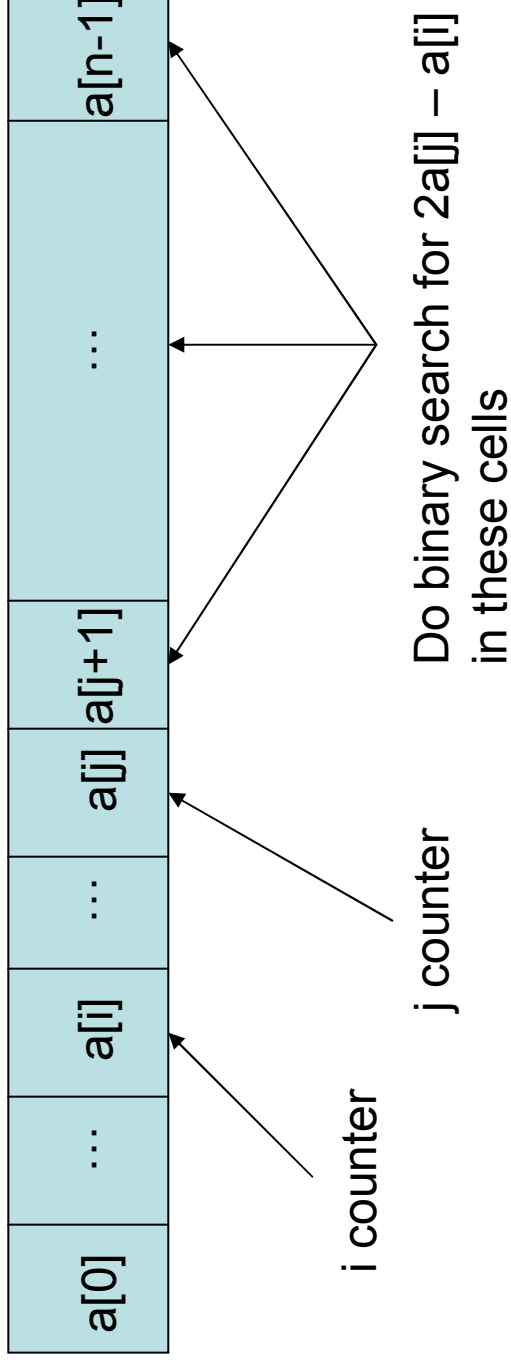
# Problem 4 and Problem 6 Solutions

# Pirates' Code

- Try all possible triples –  $O(n^3)$   
// Sort the array a  
for(int i = 0; i < n; i++)  
    for(int j = i+1; j < n; j++)  
        for(int k = j+1; k < n; k++) {  
            //check if a[i],a[j],a[k] form  
            //an arithmetic progression  
        }  
    }

# Pirates' Code

- Faster solution –  $O(n^2 \log(n))$

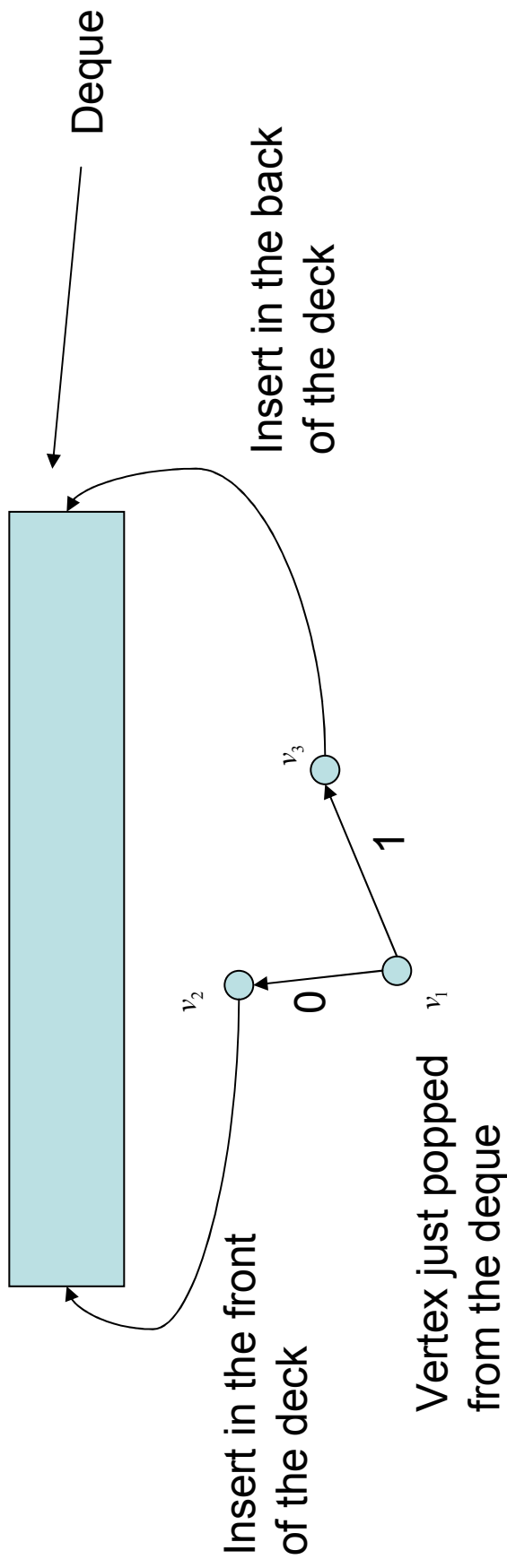


# Pirates' Path

- Dijkstra's algorithm
  - $O((V+E)\log(V))$  if you use a priority queue
  - $O(V^2)$  simple implementation with an array

# Pirates' Path

- BFS with a Deque –  $O(V+E)$



- Carefully keep track of visited vertices

# ACM ICPC

- Association of Computing Machinery  
International Collegiate Programming  
Contest
- <http://icpc.baylor.edu/icpc/>
- If you come to the University of Maryland  
send me an email
- [martin@cs.umd.edu](mailto:martin@cs.umd.edu)