Örnek Sorular

Veri Yapıları

Soru 1

```java
public static void main(String args[]) {
  Stack<String> stack = new Stack<>();
  stack.push("A");
  stack.push("B");
  stack.push("C");
  System.out.println(stack.pop());
  System.out.println(stack.peek());
  System.out.println(stack.remove("A"));
  System.out.println(stack.pop());
  System.out.println(stack.add("D"));
  System.out.println(stack.remove("B"));
}
```

Soru 2

```java
public static void main(String[] argv){
  int[] array = {6,9,8};
  List<Integer> list = new ArrayList<>();
  list.add(array[0]);
  list.add(array[2]);
  System.out.println(list);
  list.set(1, array[1]);
  list.remove(0);
  System.out.println(list);
}
```

Soru 3

```
public static void main(String[] argv){
  ArrayList<String> list = new ArrayList<>();
  list.add("dog");
  list.add("cat");
  list.add("frog");
  System.out.println(list.contains("cat"));
  //System.out.println(list.hasObject("cat"));
  System.out.println(list.indexOf("cat"));
  System.out.println(list.indexOf(1));
}
```

Soru 4

```
PriorityQueue<String> pq = new PriorityQueue<String>();
pq.add("7");
pq.add("4");
System.out.println(pq.peek() + " ");
pq.offer("2");
System.out.println(pq.peek());
pq.add("3");
System.out.println(pq.peek());
pq.offer("1");
System.out.println(pq.peek());
pq.remove("1");
System.out.println(pq.poll() + " ");
System.out.println(pq.remove("2"));
System.out.println(pq);
System.out.println(pq.poll() + " " + pq.peek());
```

Soru 5

```
int fonksiyon(ListIterator<String> it)
{
  if(it.hasNext()) {
    it.next();
    return 1 + fonksiyon(it);
  } return 0;
}
public static void main(String args[]) {
  List<String> list = new LinkedList<>();
  list.add("A"); list.add("B"); list.add("C");
  ListIterator<String> it = list.listIterator();
  System.out.println((new Ornek5()).fonksiyon(it));
}
```

Soru 6

```
public static void main(String[] args) {
  Stack<Integer> stack = new Stack<Integer>();
  int n = 12;
  while (n > 0) {
    stack.push(n % 2);
    n /= 2;
  }
  while (!stack.isEmpty())
    System.out.print(stack.pop());
}
```

Soru 7

```
private void shrinkStretch() {
  if (N < capacity / 4) {
    capacity /= 2;
    items = Arrays.copyOf(items, capacity);
  } else if (N > capacity / 2) {
    capacity *= 2;
    items = Arrays.copyOf(items, capacity);
  }
}
public void removeItem() {
  items[N--] = null;   shrinkStretch();
}
public void addItem(E e) {
  items[N++] = e;   shrinkStretch();
}
```

Soru 8

```
Stack<String> s = new Stack<>();
Queue<String> q = new LinkedList<>();
q.offer("a"); q.offer("b"); q.offer("c"); q.offer("d");

while(!q.isEmpty())
  s.push(q.poll());
System.out.println("s: " + s + " --- q: " + q);

while(!s.isEmpty())
  q.offer(s.pop());
System.out.println("s: " + s + " --- q: " + q);
```

Soru 9

```java
public static void main(String[] args) {

  Node head = new Node(1);
  head.next = new Node(2);
  head.next.next = new Node(3);
  head.next.next.next = new Node(4);
  head.next.next.next.next = head;
  System.out.println(fonksiyon(head, head.next));
}
```

Soru 9

```
public static int fonksiyon(Node head, Node curr) {
  if (head == null)
    return 1;

  if(curr == null)
    return 0;

  if(head == curr)
    return 1;

  return fonksiyon(head, curr.next);
}
```

SON