

Subject: CSE 220 Fall 14

Date: _____

Q1
a) public static void insertUnique(Object[] a, int size,
Object elem, int idx)
throws new
NoSpaceEx

```
{ if (size >= a.length)
```

```
{ throw new NoSpaceleftException;
```

```
} if (idx < 0 || idx > a.length)
```

```
{ return  
}
```

```
for (int i = 0; i < size; i++)
```

```
{ if ((Integer) a[i] == (Integer) elem)
```

```
{ return;
```

```
} }
```

```
a[idx] = elem;
```

```
size++;
```

b

```

public static void printReverse(Node head)
{
    Node x = head.prev;

```

```

    while (x != head)

```

```

    {

```

```

        System.out.println(x.elem);

```

```

    }

```

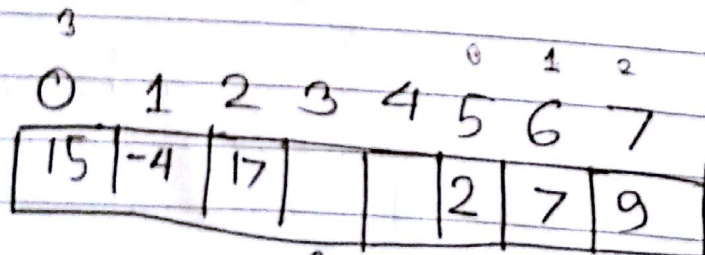
```

}

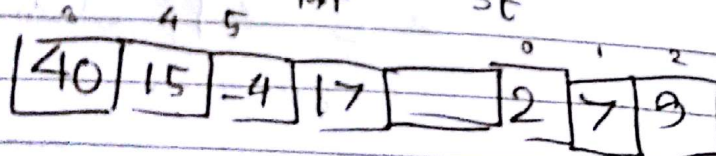
```

c

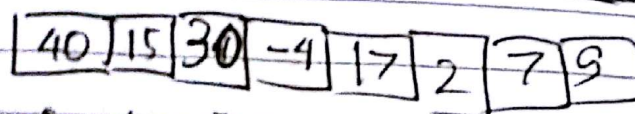
Quest



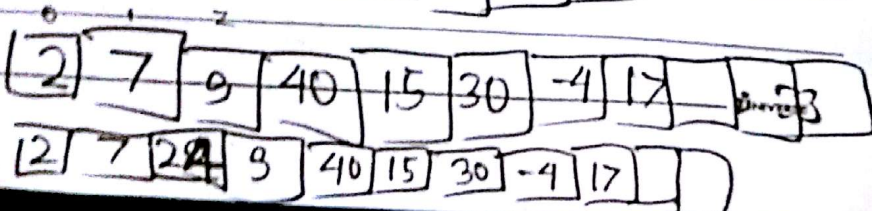
Step 1:



Step 2:

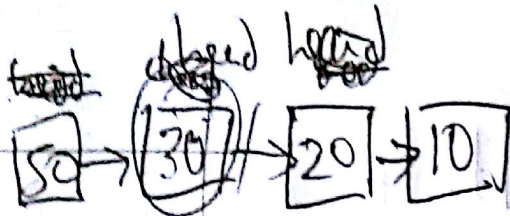


Step 3:

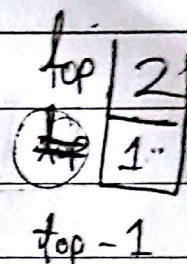
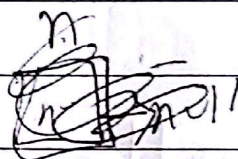
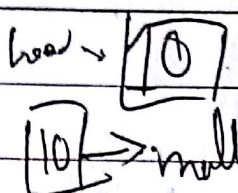


Subject :

Date :



Q2



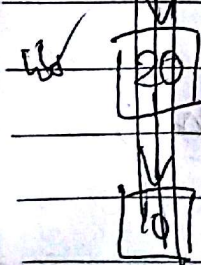
a) public void push (Object e)

```

1) {
    if (head == null)
    
```

```

        Node n = new Node(e, null);
    
```



```

        if (head == null)
        {
            head = n;
            size++;
        }
    
```

```

    else {
        n.next = head;
    }
    
```

```

    head = n;
    size++;
}
    
```

Subject: _____

Date: _____

11)

```
public Object pop() throws EmptyStackEx  
{  
    if (size == 0)  
    {  
        throw new EmptyStackEx();  
    }  
    Node x = head;  
    Object temp = head.elem;  
    head = head.next;  
  
    head (x.next = null;  
           x = null;  
           return temp;  
    )  
}
```


Q3 | dl main

val = A(5) ←

$n=5$
~~if (n == 0)~~

~~A(4)~~ + ~~A(3)~~ + ~~A(2)~~

~~BADIE~~

$\begin{matrix} 3 \\ A(3) + A(2) + A(1) \\ / \quad 2 \quad 1 \end{matrix}$

$\begin{matrix} A(2) + A(1) + A(0) \\ 2 \quad 1 \quad 0 \end{matrix}$

$\begin{matrix} A(2) + A(1) + A(0) \\ 2 \quad 1 \quad 0 \end{matrix}$

- Output
- Bad 5
 - ~~Good 2~~
 - Bad 4
 - Bad 3
 - Good 2
 - Good 1
 - Good 0
 - Good 2
 - Good 1
 - Bad 3
 - Good 2
 - Good 1
 - Good 0
 - Good 2

✓ very good !!