

Array Memory Trace

Understanding Java Arrays and References

```
public class ArrayTest {  
    static int s_var = 10;  
  
    public static int[] foo(int[] a) {  
        s_var--;  
        int[] b = new int[a.length];  
  
        for (int i = 0; i < b.length; i++)  
            b[i] = a[i]++;  
  
        return b;  
    }  
  
    public static void main(String[] args) {  
        int i = ++s_var;  
        i+=5;  
        int[] a = {2, 4, 6, 8};  
        int[] b = new int[a.length];  
        int[] c = a;  
  
        c = foo(a);  
    }  
}
```

Static

Stack

Heap

Understanding Java Arrays and References

```
public class ArrayTest {  
    static int s_var = 10;  
  
    public static int[] foo(int[] a) {  
        s_var--;  
        int[] b = new int[a.length];  
  
        for (int i = 0; i < b.length; i++)  
            b[i] = a[i]++;  
  
        return b;  
    }  
  
    public static void main(String[] args) {  
        int i = ++s_var;  
        i+=5;  
        int[] a = {2, 4, 6, 8};  
        int[] b = new int[a.length];  
        int[] c = a;  
  
        c = foo(a);  
    }  
}
```

Static	
s_var	10



Understanding Java Arrays and References

```
public class ArrayTest {  
    static int s_var = 10;  
  
    public static int[] foo(int[] a) {  
        s_var--;  
        int[] b = new int[a.length];  
  
        for (int i = 0; i < b.length; i++)  
            b[i] = a[i]++;  
  
        return b;  
    }  
  
    public static void main(String[] args) {  
        int i = ++s_var;  
        i+=5;  
        int[] a = {2, 4, 6, 8};  
        int[] b = new int[a.length];  
        int[] c = a;  
  
        c = foo(a);  
    }  
}
```

Static
s_var 10

Stack

Heap

c	
b	
a	
i	
args	

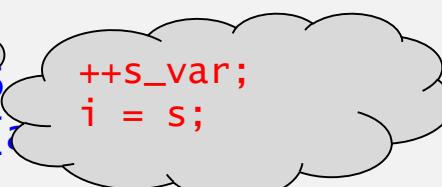
Understanding Java Arrays and References

```
public class ArrayTest {  
    static int s_var = 10;  
  
    public static int[] foo(int[] a) {  
        s_var--;  
        int[] b = new int[a.length];  
  
        for (int i = 0; i < b.length; i++)  
            b[i] = a[i]++;  
  
        return b;  
    }  
  
    public static void main(String[] args) {  
        int i = ++s_var;  
        i+=5;  
        int[] a = {2, 4, 6};  
        int[] b = new int[  
        int[] c = a;  
  
        c = foo(a);  
    }  
}
```

s_var	10
-------	----

Stack

Heap



c	
b	
a	
i	
args	

Understanding Java Arrays and References

```
public class ArrayTest {  
    static int s_var = 10;  
  
    public static int[] foo(int[] a) {  
        s_var--;  
        int[] b = new int[a.length];  
  
        for (int i = 0; i < b.length; i++)  
            b[i] = a[i]++;  
  
        return b;  
    }  
  
    public static void main(String[] args) {  
        int i = ++s_var;  
        i+=5;  
        int[] a = {2, 4, 6, 8};  
        int[] b = new int[a.length];  
        int[] c = a;  
  
        c = foo(a);  
    }  
}
```

Static	
s_var	11

Stack

Heap

c	
b	
a	
i	
args	

Understanding Java Arrays and References

```
public class ArrayTest {  
    static int s_var = 10;  
  
    public static int[] foo(int[] a) {  
        s_var--;  
        int[] b = new int[a.length];  
  
        for (int i = 0; i < b.length; i++)  
            b[i] = a[i]++;  
  
        return b;  
    }  
  
    public static void main(String[] args) {  
        int i = ++s_var;  
        i+=5;  
        int[] a = {2, 4, 6, 8};  
        int[] b = new int[a.length];  
        int[] c = a;  
  
        c = foo(a);  
    }  
}
```

Static	
s_var	11

Stack

Heap

c	
b	
a	
i	11
args	

Understanding Java Arrays and References

```
public class ArrayTest {  
    static int s_var = 10;  
  
    public static int[] foo(int[] a) {  
        s_var--;  
        int[] b = new int[a.length];  
  
        for (int i = 0; i < b.length; i++)  
            b[i] = a[i]++;  
  
        return b;  
    }  
  
    public static void main(String[] args) {  
        int i = ++s_var;  
        i+=5;  
        int[] a = {2, 4, 6, 8};  
        int[] b = new int[a.length];  
        int[] c = a;  
  
        c = foo(a);  
    }  
}
```

Static	
s_var	11

Stack

Heap

c	
b	
a	
i	16
args	

Understanding Java Arrays and References

```
public class ArrayTest {  
    static int s_var = 10;  
  
    public static int[] foo(int[] a) {  
        s_var--;  
        int[] b = new int[a.length];  
  
        for (int i = 0; i < b.length; i++)  
            b[i] = a[i]++;  
  
        return b;  
    }  
  
    public static void main(String[] args) {  
        int i = ++s_var;  
        i+=5;  
        int[] a = {2, 4, 6, 8};  
        int[] b = new int[a.length];  
        int[] c = a;  
  
        c = foo(a);  
    }  
}
```

Static	Stack
s_var	11

Stack

2
4
6
8

c	
b	
a	
i	16
args	

Understanding Java Arrays and References

```
public class ArrayTest {  
    static int s_var = 10;  
  
    public static int[] foo(int[] a) {  
        s_var--;  
        int[] b = new int[a.length];  
  
        for (int i = 0; i < b.length; i++)  
            b[i] = a[i]++;  
  
        return b;  
    }  
  
    public static void main(String[] args) {  
        int i = ++s_var;  
        i+=5;  
        int[] a = {2, 4, 6, 8};  
        int[] b = new int[a.length];  
        int[] c = a;  
  
        c = foo(a);  
    }  
}
```

Static	Stack
s_var	11

Stack

Heap

2
4
6
8

c	
b	
a	
i	16
args	

Understanding Java Arrays and References

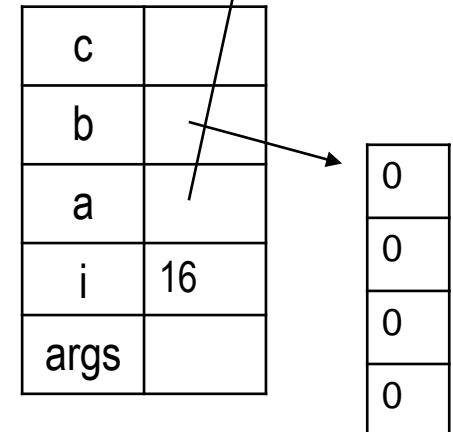
```
public class ArrayTest {  
    static int s_var = 10;  
  
    public static int[] foo(int[] a) {  
        s_var--;  
        int[] b = new int[a.length];  
  
        for (int i = 0; i < b.length; i++)  
            b[i] = a[i]++;  
  
        return b;  
    }  
  
    public static void main(String[] args) {  
        int i = ++s_var;  
        i+=5;  
        int[] a = {2, 4, 6, 8};  
        int[] b = new int[a.length];  
        int[] c = a;  
  
        c = foo(a);  
    }  
}
```

Static	s_var	11
--------	-------	----

Stack

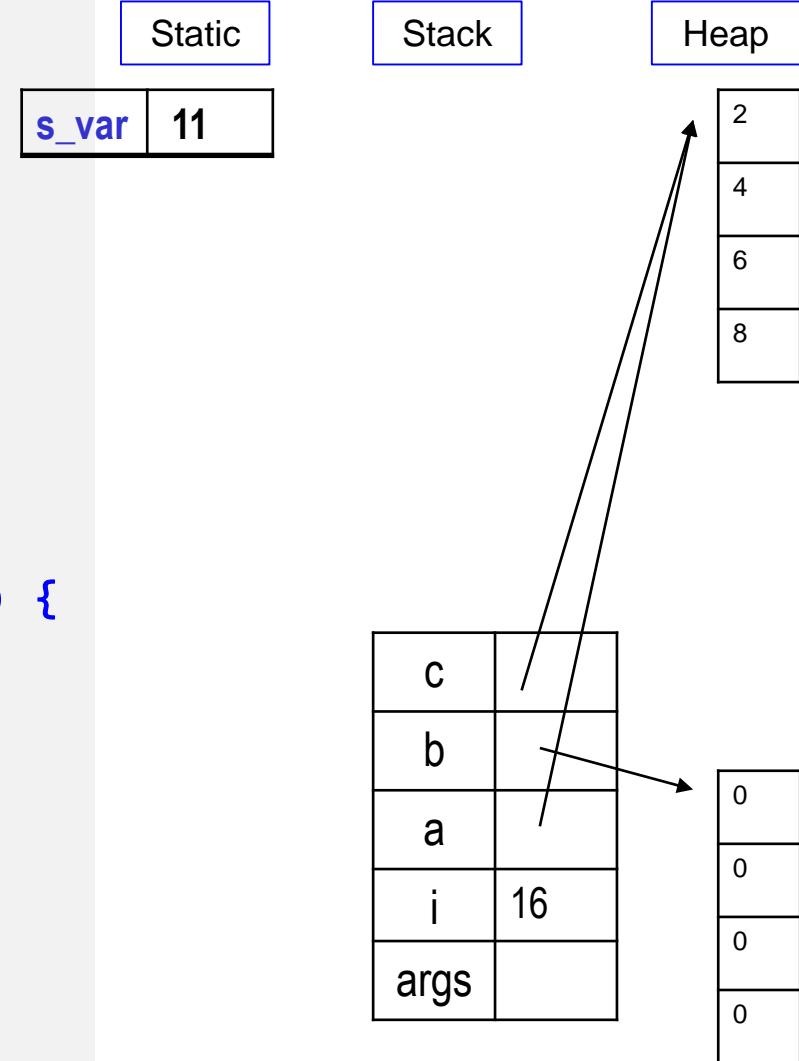
Heap

2
4
6
8



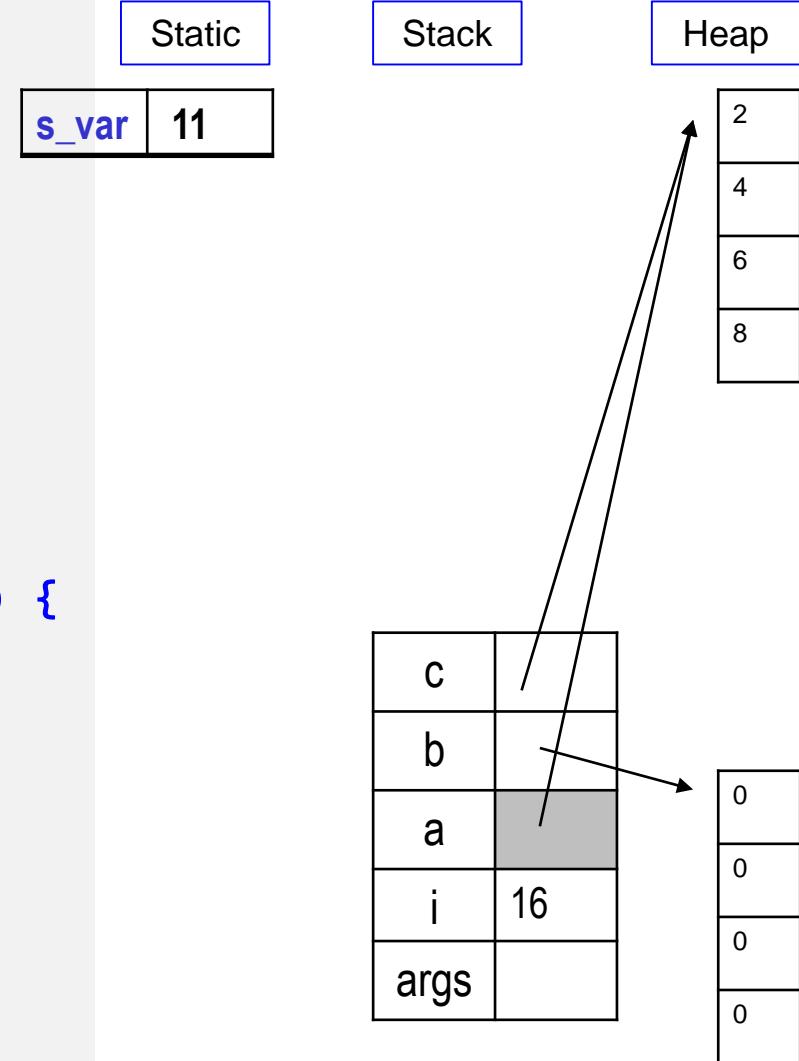
Understanding Java Arrays and References

```
public class ArrayTest {  
    static int s_var = 10;  
  
    public static int[] foo(int[] a) {  
        s_var--;  
        int[] b = new int[a.length];  
  
        for (int i = 0; i < b.length; i++)  
            b[i] = a[i]++;  
  
        return b;  
    }  
  
    public static void main(String[] args) {  
        int i = ++s_var;  
        i+=5;  
        int[] a = {2, 4, 6, 8};  
        int[] b = new int[a.length];  
        int[] c = a;  
  
        c = foo(a);  
    }  
}
```



Understanding Java Arrays and References

```
public class ArrayTest {  
    static int s_var = 10;  
  
    public static int[] foo(int[] a) {  
        s_var--;  
        int[] b = new int[a.length];  
  
        for (int i = 0; i < b.length; i++)  
            b[i] = a[i]++;  
  
        return b;  
    }  
  
    public static void main(String[] args) {  
        int i = ++s_var;  
        i+=5;  
        int[] a = {2, 4, 6, 8};  
        int[] b = new int[a.length];  
        int[] c = a;  
  
        c = foo(a);  
    }  
}
```



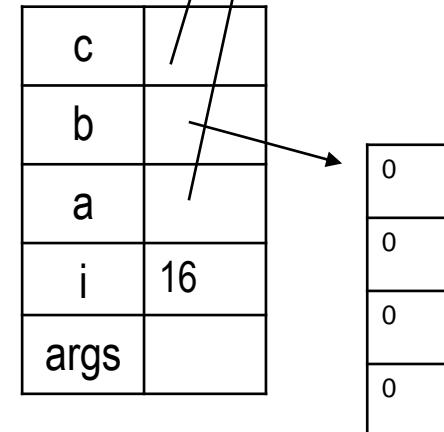
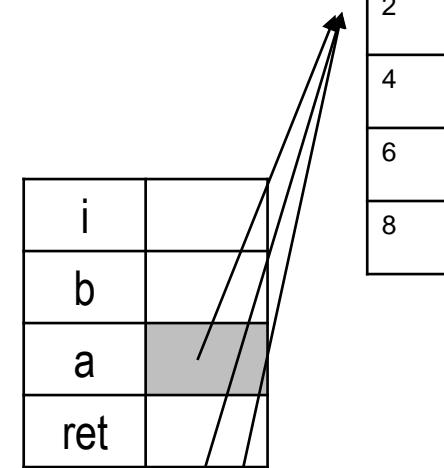
Understanding Java Arrays and References

```
public class ArrayTest {  
    static int s_var = 10;  
  
    public static int[] foo(int[] a) {  
        s_var--;  
        int[] b = new int[a.length];  
  
        for (int i = 0; i < b.length; i++)  
            b[i] = a[i]++;  
  
        return b;  
    }  
  
    public static void main(String[] args) {  
        int i = ++s_var++;  
        i+=5;  
        int[] a = {2, 4, 6, 8};  
        int[] b = new int[a.length];  
        int[] c = a;  
  
        c = foo(a);  
    }  
}
```

Static
s_var 11

Stack

Heap



Understanding Java Arrays and References

```
public class ArrayTest {  
    static int s_var = 10;  
  
    public static int[] foo(int[] a) {  
        s_var--;  
        int[] b = new int[a.length];  
  
        for (int i = 0; i < b.length; i++)  
            b[i] = a[i]++;  
  
        return b;  
    }  
  
    public static void main(String[] args) {  
        int i = ++s_var;  
        i+=5;  
        int[] a = {2, 4, 6, 8};  
        int[] b = new int[a.length];  
        int[] c = a;  
  
        c = foo(a);  
    }  
}
```

Static
s_var 10

Stack
i b a ret

Heap
2 4 6 8

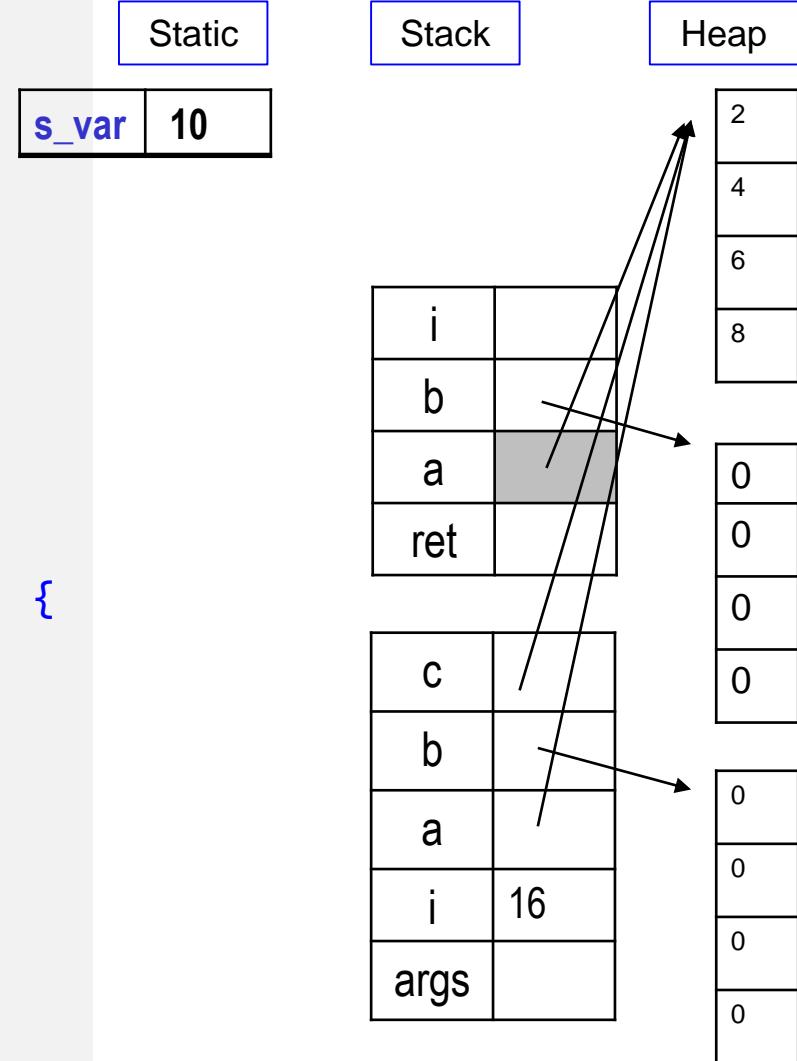
i	
b	
a	
ret	

c	
b	
a	
i	16
args	

2
4
6
8

Understanding Java Arrays and References

```
public class ArrayTest {  
    static int s_var = 10;  
  
    public static int[] foo(int[] a) {  
        s_var--;  
        int[] b = new int[a.length];  
  
        for (int i = 0; i < b.length; i++)  
            b[i] = a[i]++;  
  
        return b;  
    }  
  
    public static void main(String[] args) {  
        int i = ++s_var;  
        i+=5;  
        int[] a = {2, 4, 6, 8};  
        int[] b = new int[a.length];  
        int[] c = a;  
  
        c = foo(a);  
    }  
}
```



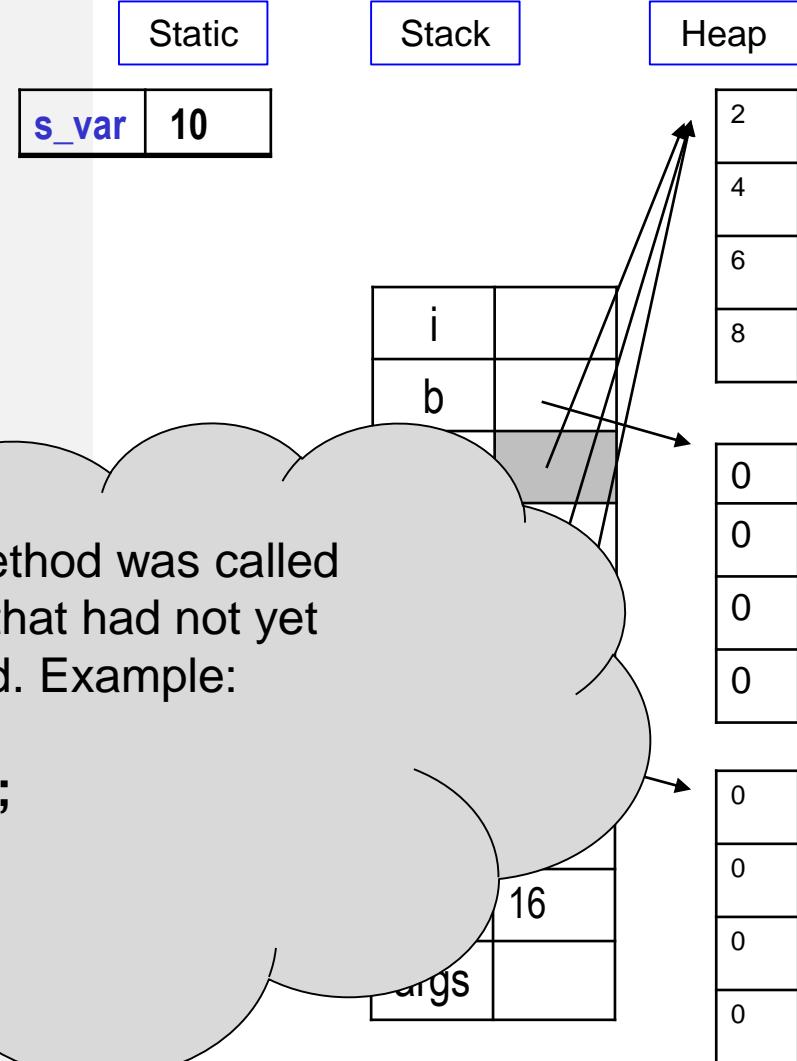
Understanding Java Arrays and References

```
public class ArrayTest {  
    static int s_var = 10;  
  
    public static int[] foo(int[] a) {  
        s_var--;  
        int[] b = new int[a.length];  
  
        for (int i = 0; i < b.length; i++)  
            b[i] = a[i]++;  
  
        return b;  
    }  
}
```

```
public static void main  
{  
    int i = ++s_var;  
    i+=5;  
    int[] a = {2, 4,  
    int[] b = new int[  
    int[] c = a;  
  
    c = foo(a);  
}
```

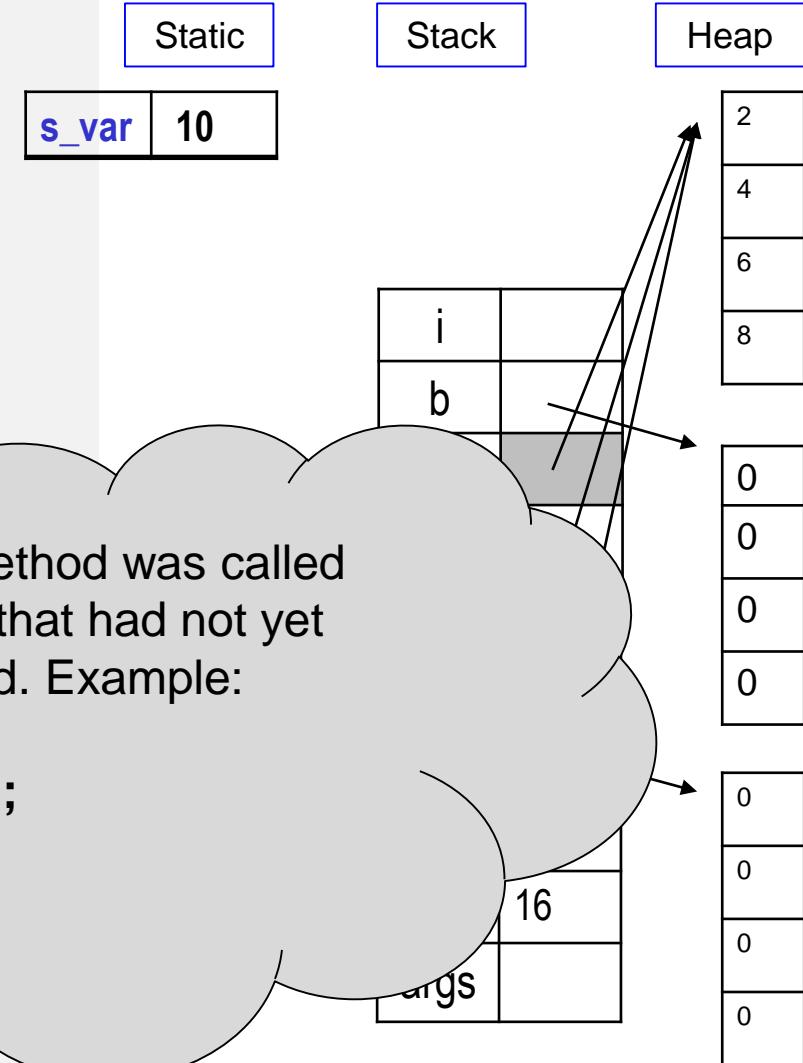
What if the method was called
with an array that had not yet
been allocated. Example:

```
{  
    int[] a;  
    foo(a);  
}
```



Understanding Java Arrays and References

```
public class ArrayTest {  
    static int s_var = 10;  
  
    public static int[] foo(int[] a) {  
        s_var--;  
        int[] b = new int[a.length]; //error  
        for (int i = 0; i < b.length; i++)  
            b[i] = a[i]++;  
  
        return b;  
    }  
  
    public static void main()  
    {  
        int i = ++s_var;  
        i+=5;  
        int[] a = {2, 4,  
        int[] b = new int[  
        int[] c = a;  
  
        c = foo(a);  
    }
```

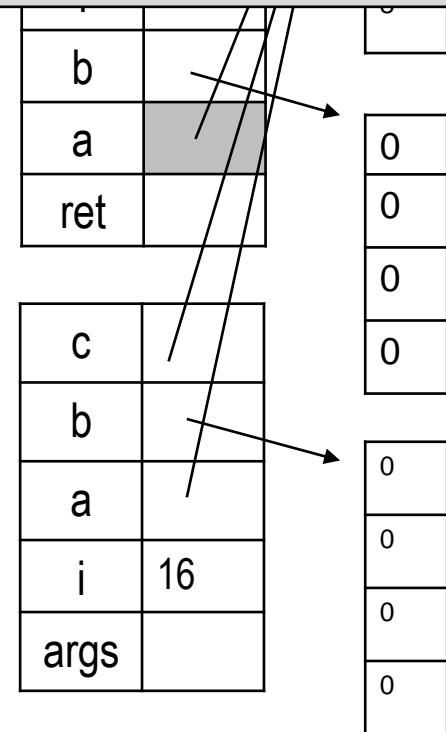


Understand

```
public class ArrayTest {  
    static int s_var = 10;  
  
    public static int[] foo()  
    {  
        s_var--;  
        int[] b = new int[a.length];  
  
        for (int i = 0; i < b.length; i++)  
            b[i] = a[i]++;  
  
        return b;  
    }  
}
```

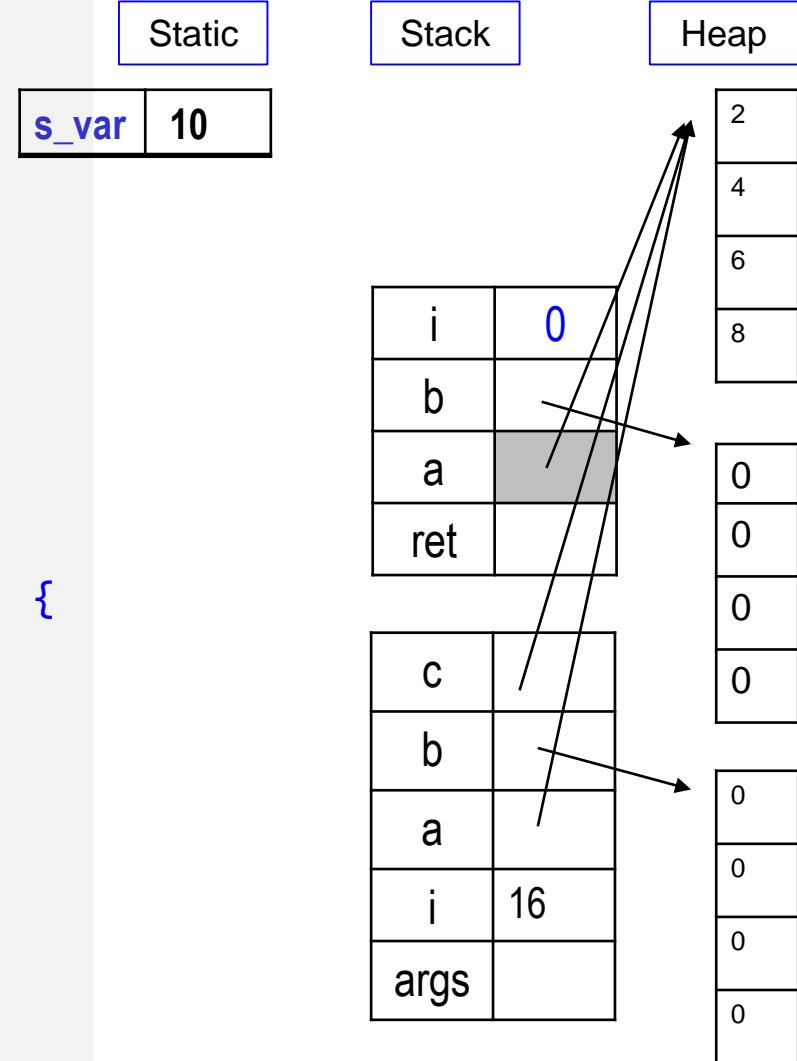
Add error checking to the beginning of the method:

```
{  
    if (a == null)  
        throw new IllegalArgumentException();  
  
    // method continues...  
    s_var--;  
  
    .  
    .
```



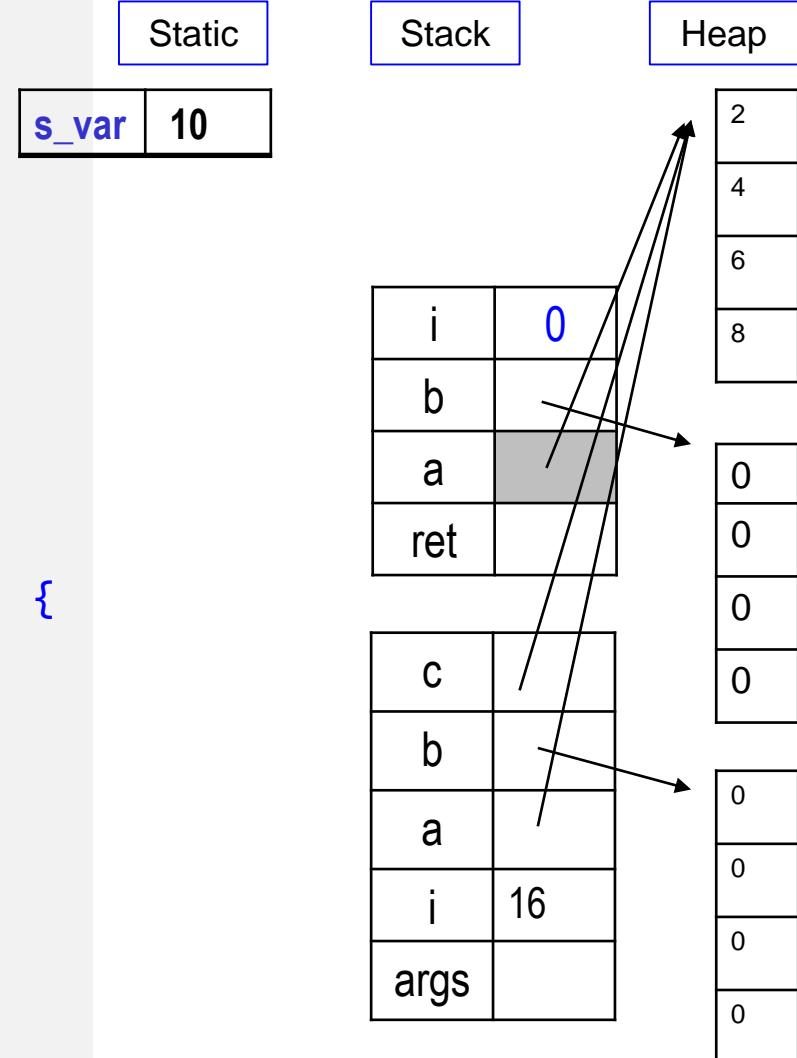
Understanding Java Arrays and References

```
public class ArrayTest {  
    static int s_var = 10;  
  
    public static int[] foo(int[] a) {  
        s_var--;  
        int[] b = new int[a.length];  
  
        for (int i = 0; i < b.length; i++)  
            b[i] = a[i]++;  
  
        return b;  
    }  
  
    public static void main(String[] args) {  
        int i = ++s_var;  
        i+=5;  
        int[] a = {2, 4, 6, 8};  
        int[] b = new int[a.length];  
        int[] c = a;  
  
        c = foo(a);  
    }  
}
```



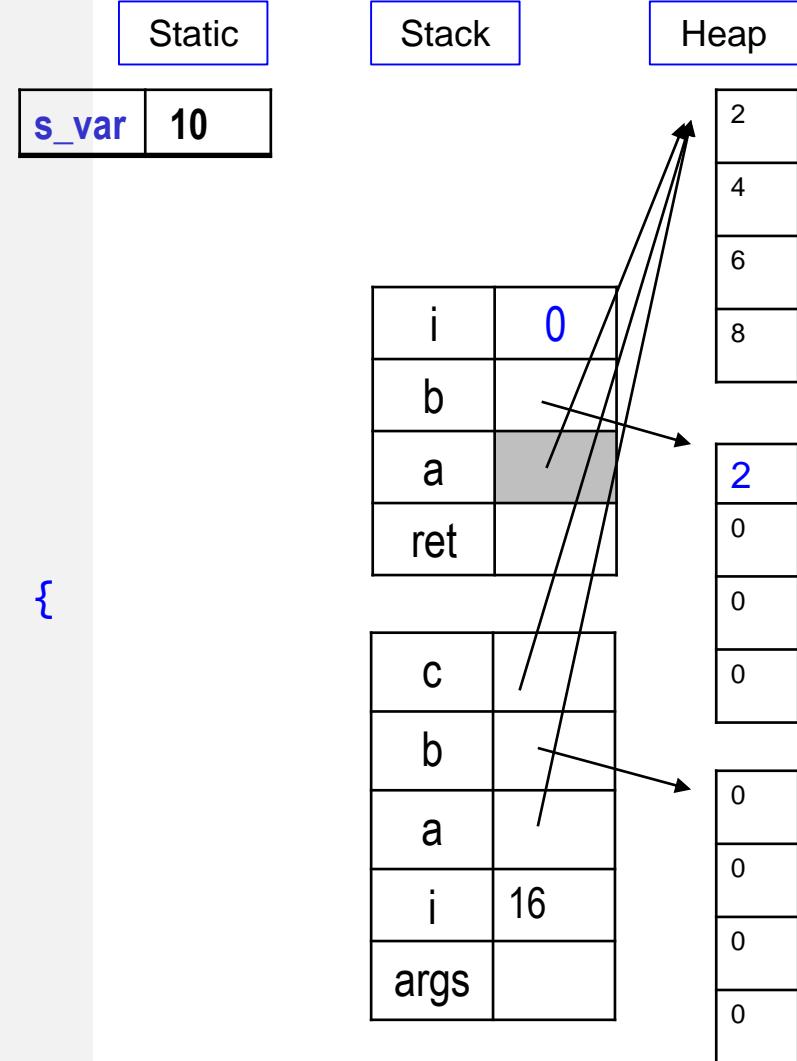
Understanding Java Arrays and References

```
public class ArrayTest {  
    static int s_var = 10;  
  
    public static int[] foo(int[] a) {  
        s_var--;  
        int[] b = new int[a.length];  
  
        for (int i = 0; i < b.length; i++)  
            b[i] = a[i]++;  
  
        return b;  
    }  
  
    public static void main(String[] args) {  
        int i = ++s_var;  
        i+=5;  
        int[] a = {2, 4, 6, 8};  
        int[] b = new int[a.length];  
        int[] c = a;  
  
        c = foo(a);  
    }  
}
```



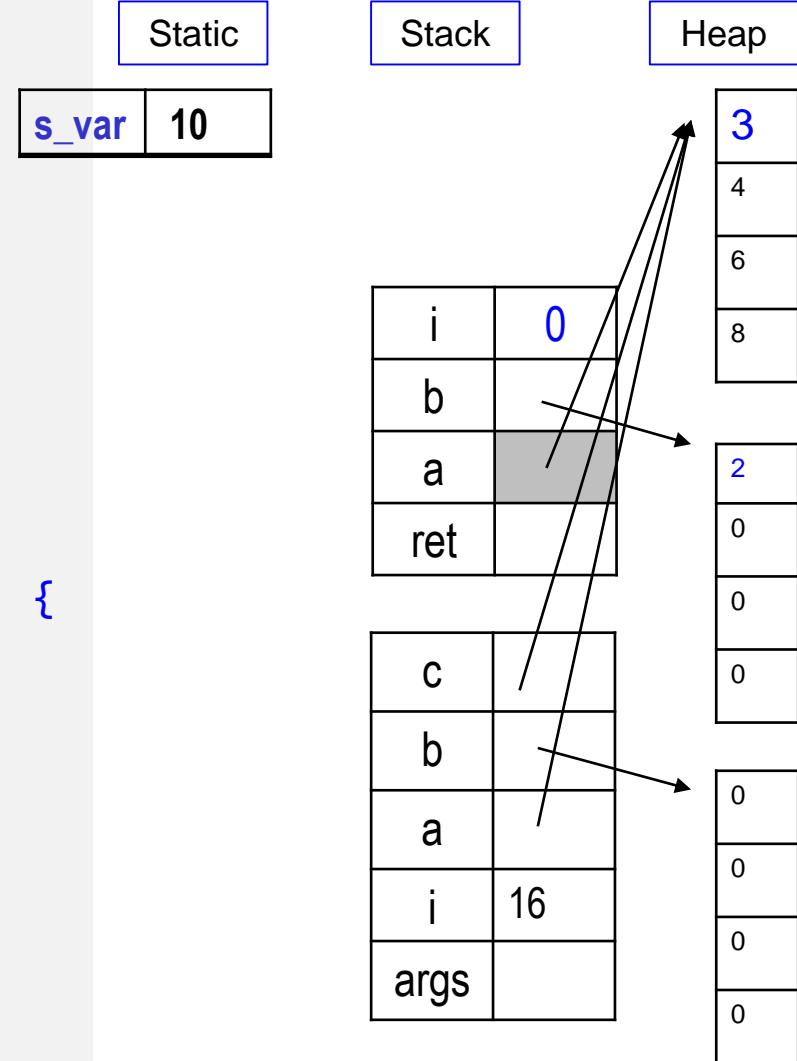
Understanding Java Arrays and References

```
public class ArrayTest {  
    static int s_var = 10;  
  
    public static int[] foo(int[] a) {  
        s_var--;  
        int[] b = new int[a.length];  
  
        for (int i = 0; i < b.length; i++)  
            b[i] = a[i]++;  
  
        return b;  
    }  
  
    public static void main(String[] args) {  
        int i = ++s_var;  
        i+=5;  
        int[] a = {2, 4, 6, 8};  
        int[] b = new int[a.length];  
        int[] c = a;  
  
        c = foo(a);  
    }  
}
```



Understanding Java Arrays and References

```
public class ArrayTest {  
    static int s_var = 10;  
  
    public static int[] foo(int[] a) {  
        s_var--;  
        int[] b = new int[a.length];  
  
        for (int i = 0; i < b.length; i++)  
            b[i] = a[i]++;  
  
        return b;  
    }  
  
    public static void main(String[] args) {  
        int i = ++s_var;  
        i+=5;  
        int[] a = {2, 4, 6, 8};  
        int[] b = new int[a.length];  
        int[] c = a;  
  
        c = foo(a);  
    }  
}
```



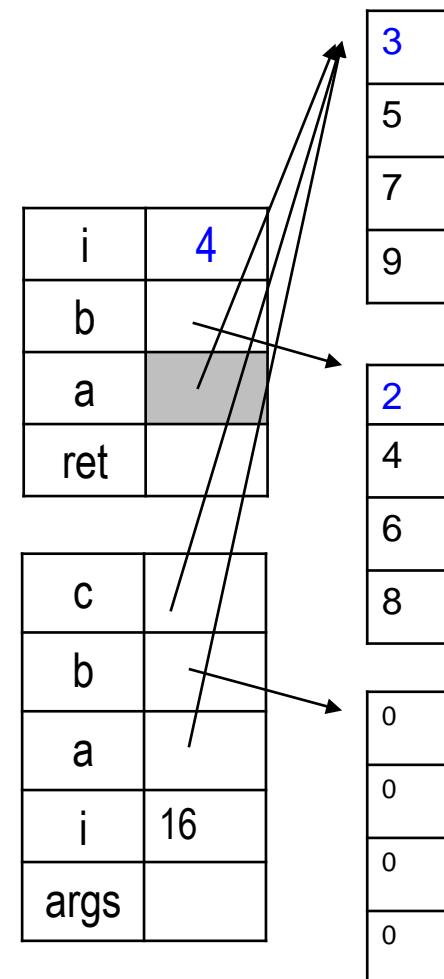
Understanding Java Arrays and References

```
public class ArrayTest {  
    static int s_var = 10;  
  
    public static int[] foo(int[] a) {  
        s_var--;  
        int[] b = new int[a.length];  
  
        for (int i = 0; i < b.length; i++)  
            b[i] = a[i]++;  
  
        return b;  
    }  
  
    public static void main(String[] args) {  
        int i = ++s_var;  
        i+=5;  
        int[] a = {2, 4, 6, 8};  
        int[] b = new int[a.length];  
        int[] c = a;  
  
        c = foo(a);  
    }  
}
```

Static	s_var	10
--------	-------	----

Stack

Heap



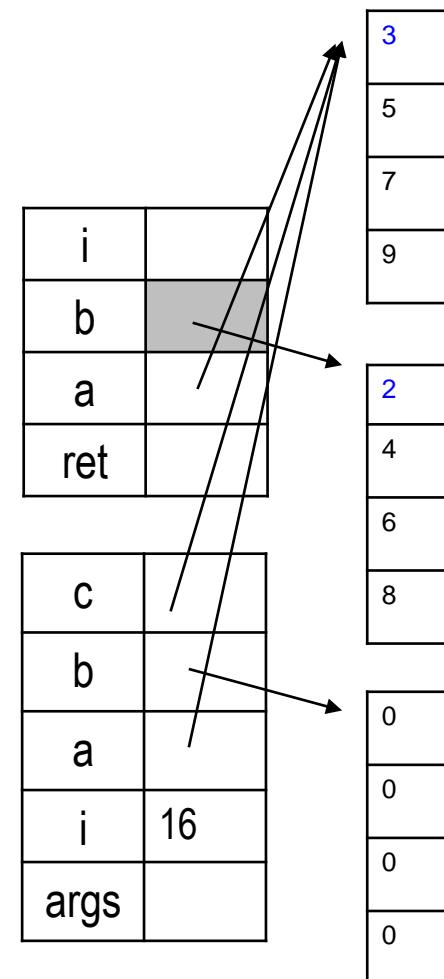
Understanding Java Arrays and References

```
public class ArrayTest {  
    static int s_var = 10;  
  
    public static int[] foo(int[] a) {  
        s_var--;  
        int[] b = new int[a.length];  
  
        for (int i = 0; i < b.length; i++)  
            b[i] = a[i]++;  
  
        return b;  
    }  
  
    public static void main(String[] args) {  
        int i = ++s_var;  
        i+=5;  
        int[] a = {2, 4, 6, 8};  
        int[] b = new int[a.length];  
        int[] c = a;  
  
        c = foo(a);  
    }  
}
```

Static	s_var	10
--------	-------	----

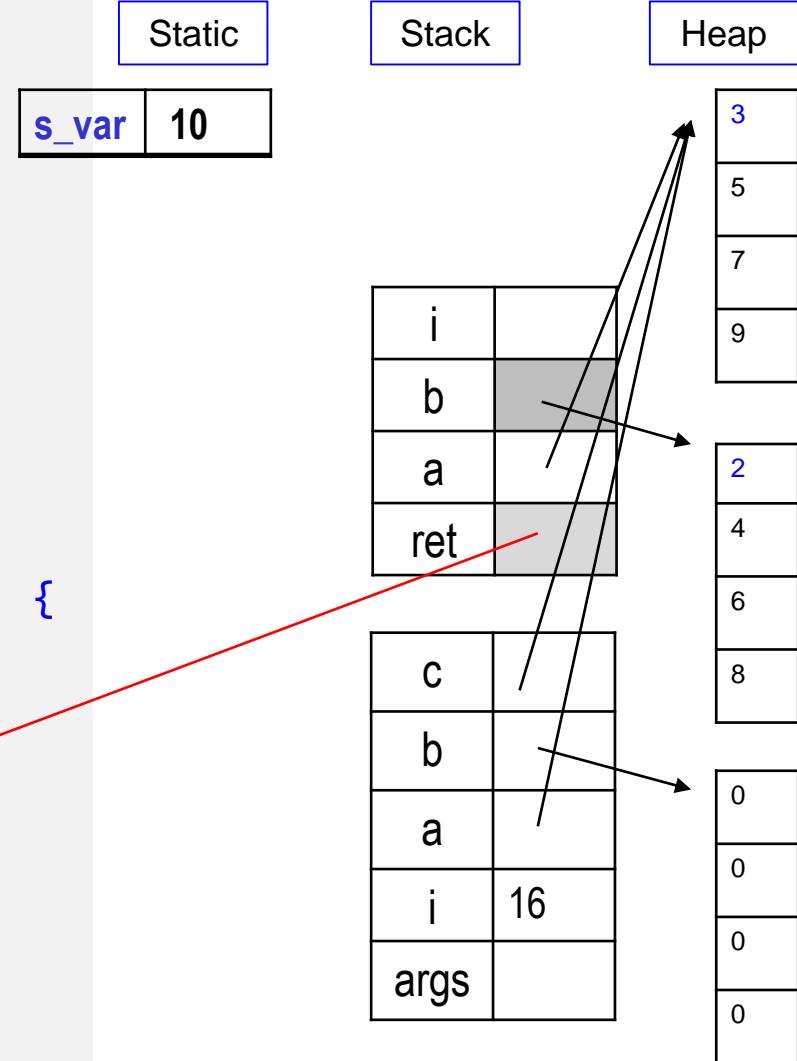
Stack

Heap



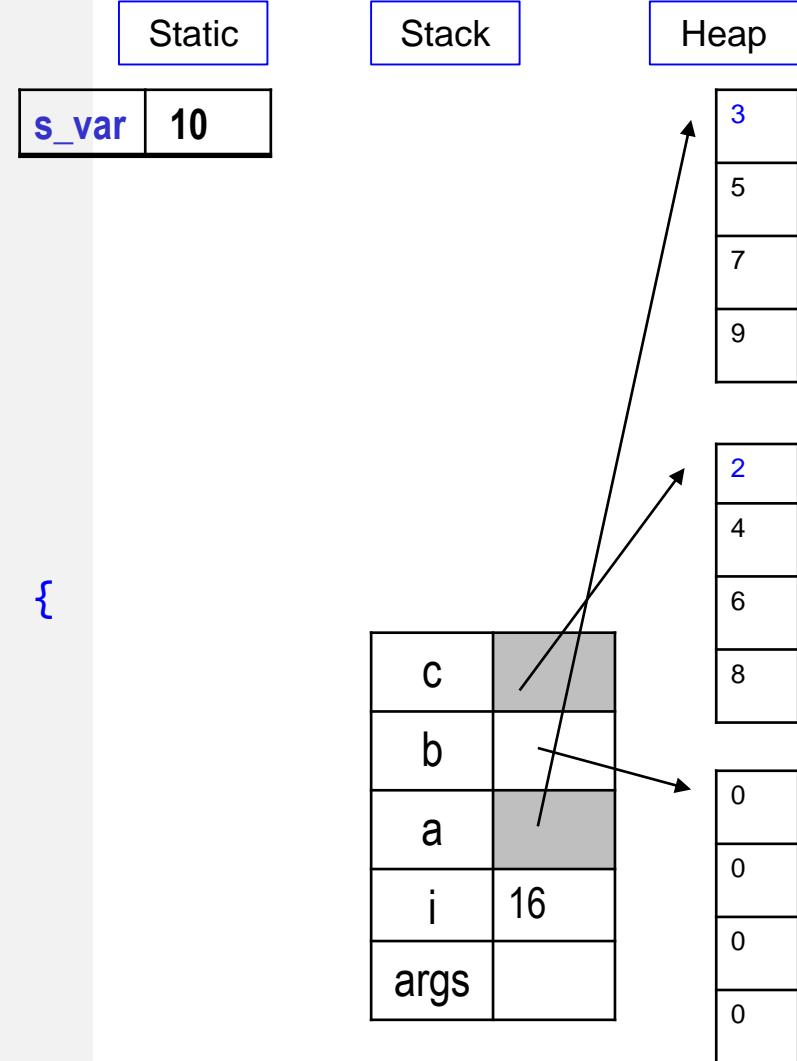
Understanding Java Arrays and References

```
public class ArrayTest {  
    static int s_var = 10;  
  
    public static int[] foo(int[] a) {  
        s_var--;  
        int[] b = new int[a.length];  
  
        for (int i = 0; i < b.length; i++)  
            b[i] = a[i]++;  
  
        return b;  
    }  
  
    public static void main(String[] args) {  
        int i = ++s_var;  
        i+=5;  
        int[] a = {2, 4, 6, 8};  
        int[] b = new int[a.length];  
        int[] c = a;  
  
        c = foo(a);  
    }  
}
```



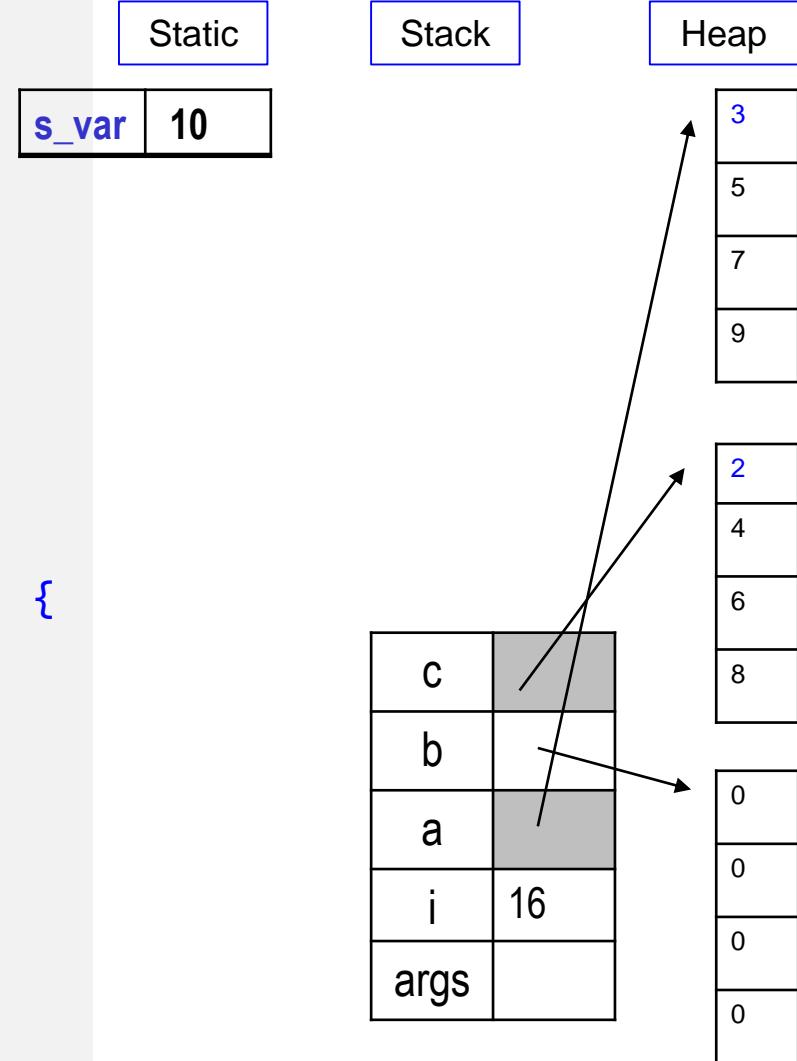
Understanding Java Arrays and References

```
public class ArrayTest {  
    static int s_var = 10;  
  
    public static int[] foo(int[] a) {  
        s_var--;  
        int[] b = new int[a.length];  
  
        for (int i = 0; i < b.length; i++)  
            b[i] = a[i]++;  
  
        return b;  
    }  
  
    public static void main(String[] args) {  
        int i = ++s_var;  
        i+=5;  
        int[] a = {2, 4, 6, 8};  
        int[] b = new int[a.length];  
        int[] c = a;  
  
        c = foo(a);  
    }  
}
```



Understanding Java Arrays and References

```
public class ArrayTest {  
    static int s_var = 10;  
  
    public static int[] foo(int[] a) {  
        s_var--;  
        int[] b = new int[a.length];  
  
        for (int i = 0; i < b.length; i++)  
            b[i] = a[i]++;  
  
        return b;  
    }  
  
    public static void main(String[] args) {  
        int i = ++s_var;  
        i+=5;  
        int[] a = {2, 4, 6, 8};  
        int[] b = new int[a.length];  
        int[] c = a;  
  
        c = foo(a);  
    }  
}
```



Understanding Java Arrays and References

```
public class ArrayTest {  
    static int s_var = 10;  
  
    public static int[] foo(int[] a) {  
        s_var--;  
        int[] b = new int[a.length];  
  
        for (int i = 0; i < b.length; i++)  
            b[i] = a[i]++;  
  
        return b;  
    }  
  
    public static void main(String[] args) {  
        int i = ++s_var;  
        i+=5;  
        int[] a = {2, 4, 6, 8};  
        int[] b = new int[a.length];  
        int[] c = a;  
  
        c = foo(a);  
    }  
}
```

Static	Stack
s_var	10

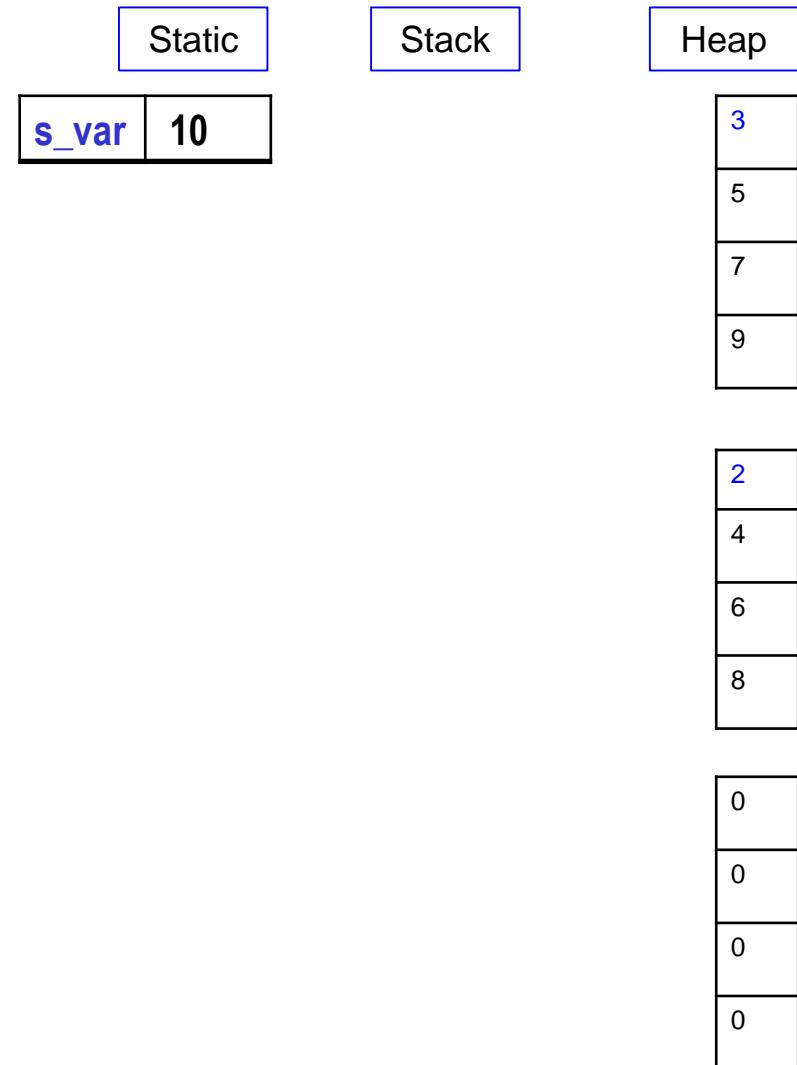


3
5
7
9

2
4
6
8

0
0
0
0

Understanding Java Arrays and References



Understanding Java Arrays and References

Static

Stack

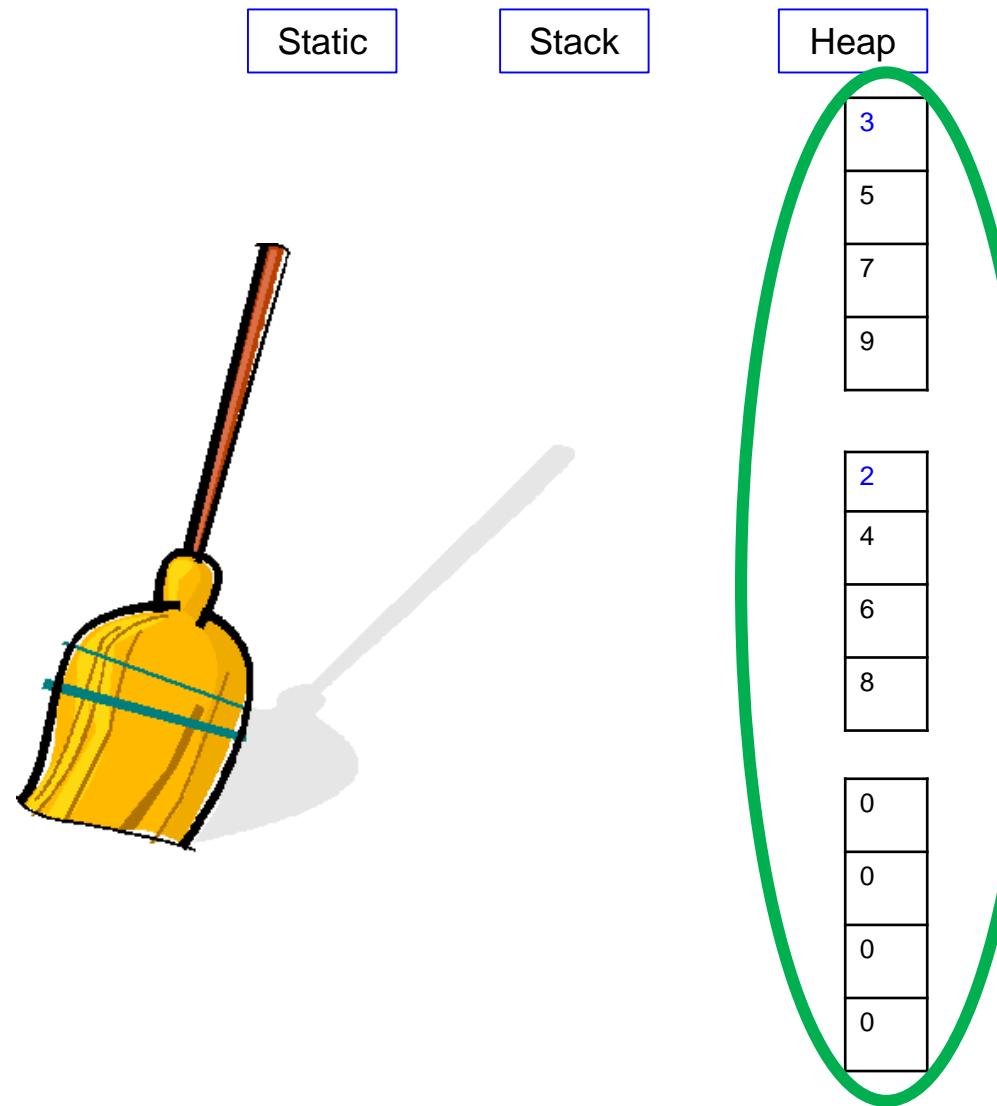
Heap

3
5
7
9

2
4
6
8

0
0
0
0

Understanding Java Arrays and References



Understanding Java Arrays and References

