Data Structures
Stack – Array Implementation

Team Emertxe

Stack – Array Implementation

Data Structure Stack-Array Implementation

Operations

- Create Stack
- Insert an Element
- Delete an Element

Stack – pop(stack,element)

pop(stack,element)

Input Specification:

stack : Pointer that contains address of structure variable (stack_t)

element : Pointer that contains address of an integer variable

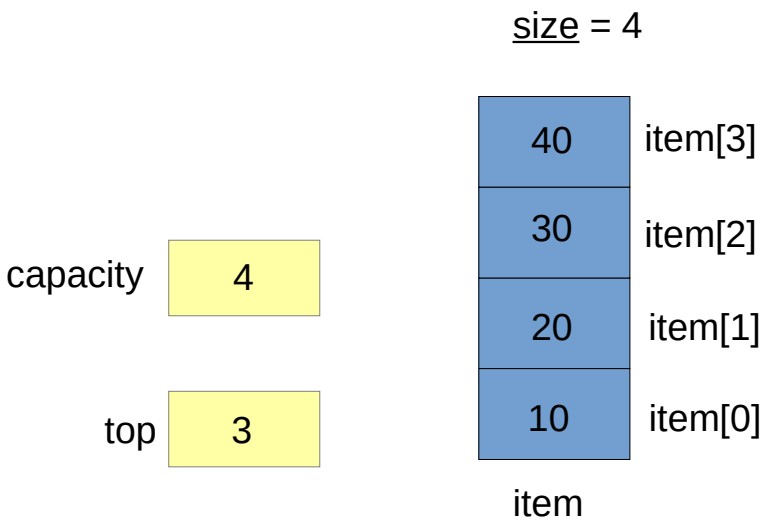
Output Specification:

Status : e_true / e_false

Data Structure Stack-Array Implementation

pop(stack,element)

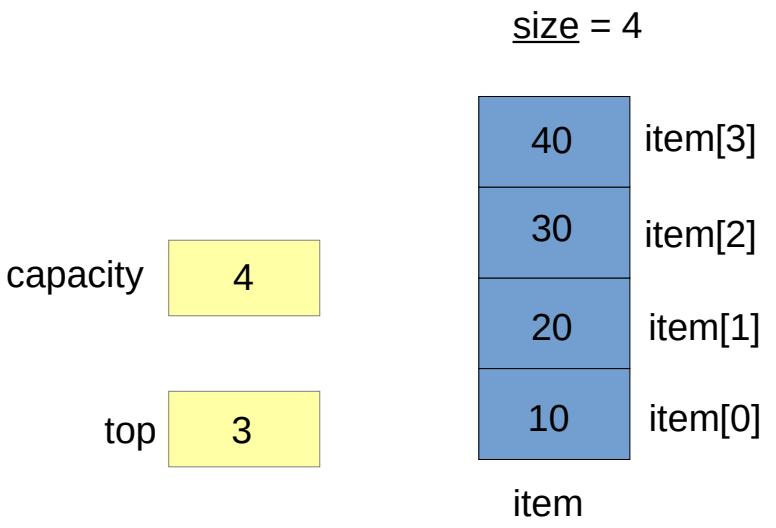
```
if(is_stack_empty(stack))
    return e_false
element = stack→item[stack→top]
(stack→top)--
return e_true
```



Data Structure Stack-Array Implementation

pop(stack,element)

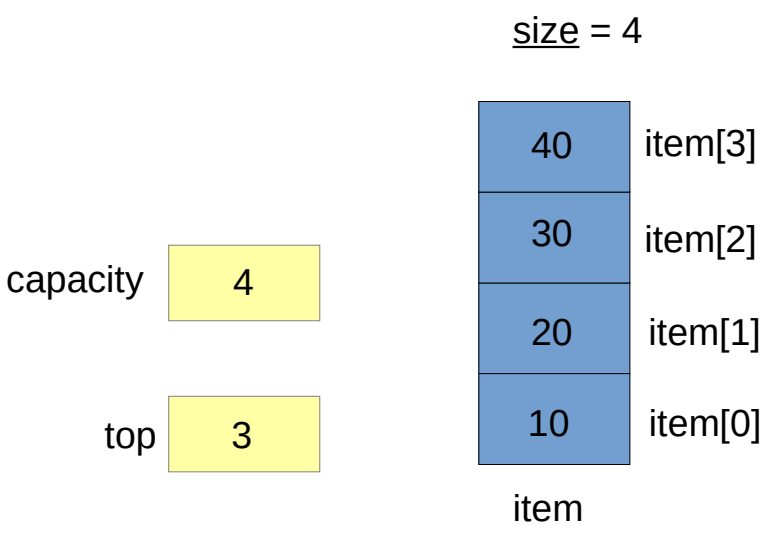
```
if(is_stack_empty(stack))
    return e_false
element = stack→item[stack→top]
(stack→top)--
return e_true
```



Data Structure Stack-Array Implementation

pop(stack,element)

```
if(is_stack_empty(stack))
    return e_false
element = stack→item[stack→top]
(stack→top)--
return e_true
```



is_stack_empty(stack)

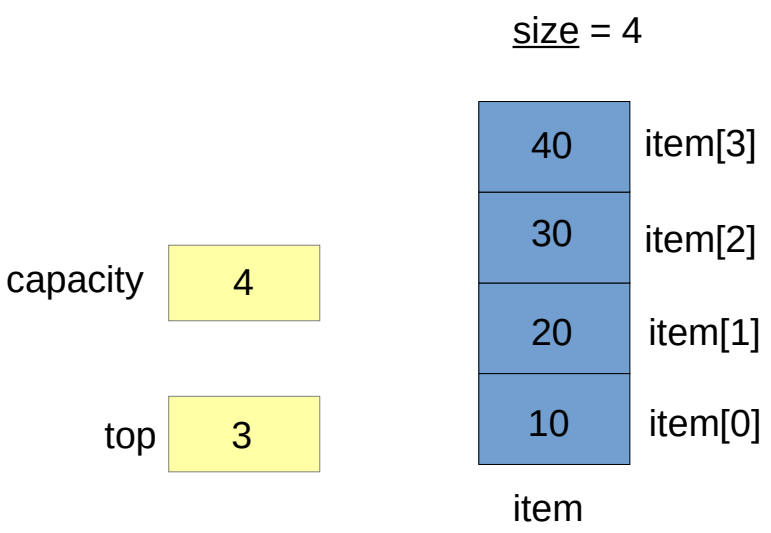
```
If (stack→top = -1)
    return e_true
else
    return e_false
```

EMERTXE

Data Structure Stack-Array Implementation

pop(stack,element)

```
if(is_stack_empty(stack))
    return e_false
element = stack→item[stack→top]
(stack→top)--
return e_true
```



is_stack_empty(stack)

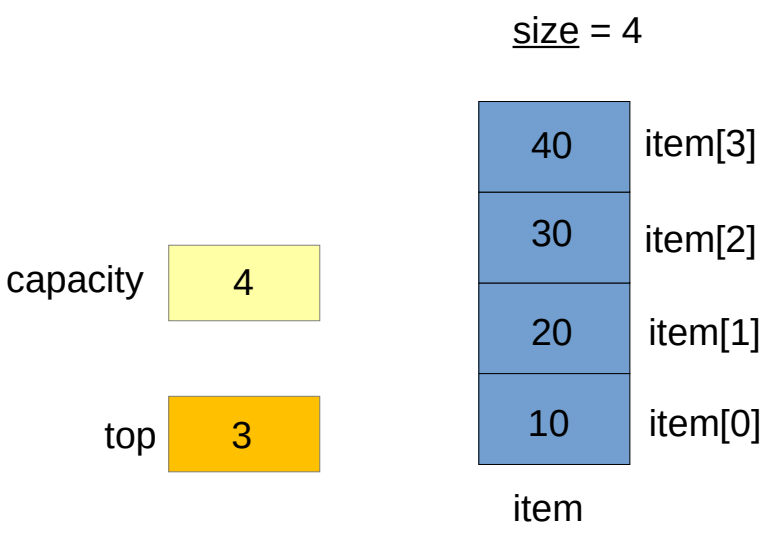
```
If (stack→top = -1)
    return e_true
else
    return e_false
```

EMERTXE

Data Structure Stack-Array Implementation

pop(stack,element)

```
if(is_stack_empty(stack))
    return e_false
element = stack→item[stack→top]
(stack→top)--
return e_true
```



is_stack_empty(stack)

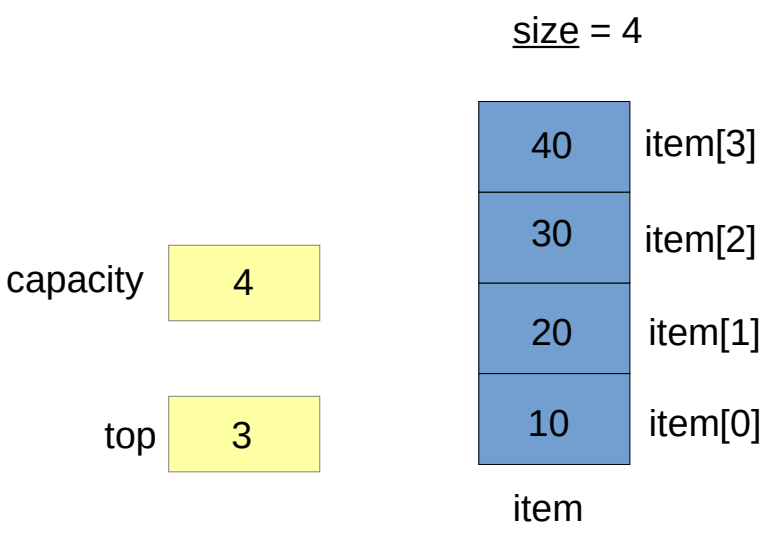
```
If (stack→top = -1)
    return e_true
else
    return e_false
```

EMERTXE

Data Structure Stack-Array Implementation

pop(stack,element)

```
if(is_stack_empty(stack))
    return e_false
element = stack→item[stack→top]
(stack→top)--
return e_true
```



is_stack_empty(stack)

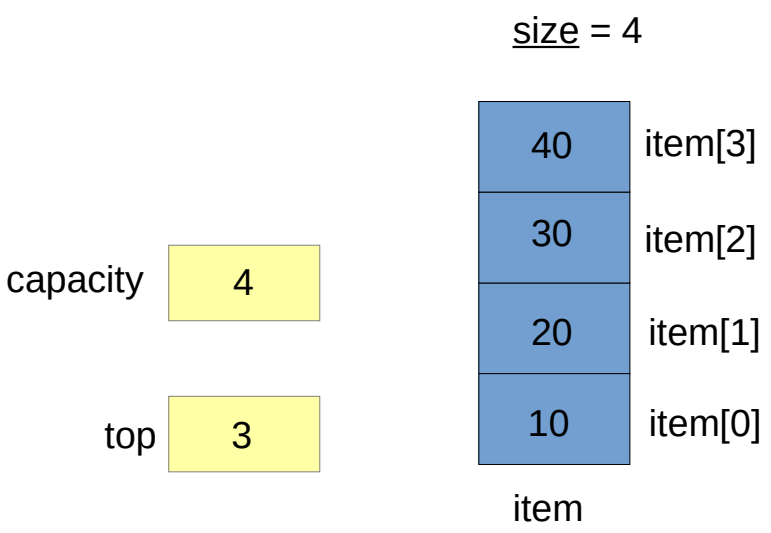
```
If (stack→top = -1)
    return e_true
else
    return e_false
```

EMERTXE

Data Structure Stack-Array Implementation

pop(stack,element)

```
if(is_stack_empty(stack))
    return e_false
element = stack→item[stack→top]
(stack→top)--
return e_true
```



is_stack_empty(stack)

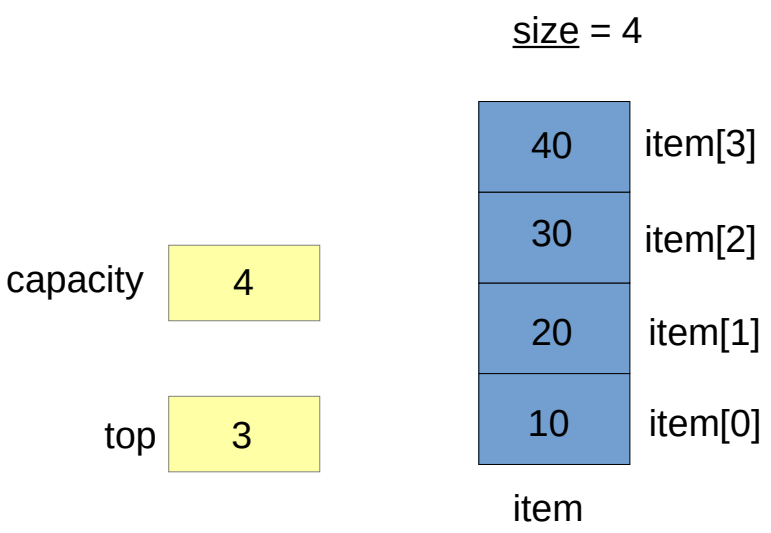
```
If (stack→top = -1)
    return e_true
else
    return e_false
```

EMERTXE

Data Structure Stack-Array Implementation

pop(stack,element)

```
if(is_stack_empty(stack))
    return e_false
element = stack→item[stack→top]
(stack→top)--
return e_true
```



is_stack_empty(stack)

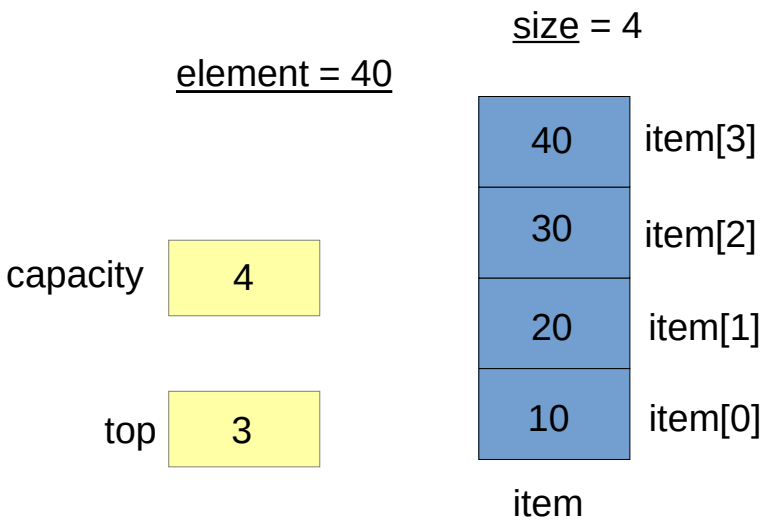
```
If (stack→top = -1)
    return e_true
else
    return e_false
```

EMERTXE

Data Structure Stack-Array Implementation

pop(stack,element)

```
if(is_stack_empty(stack))
    return e_false
element = stack→item[stack→top]
(stack→top)--
return e_true
```



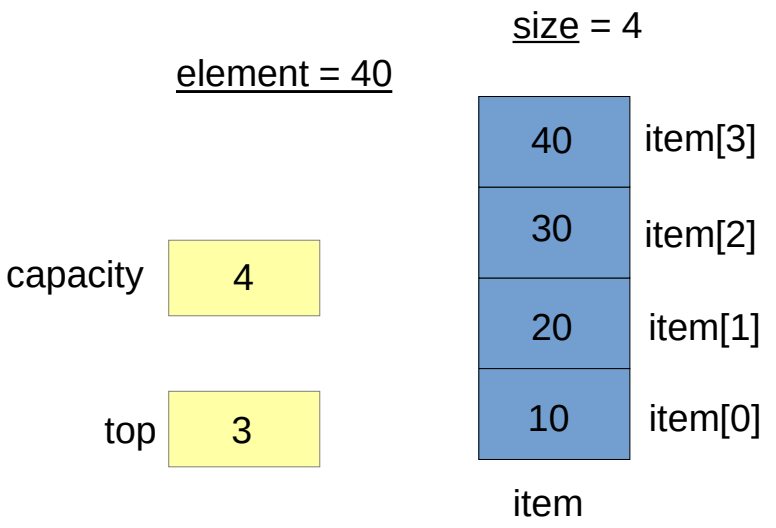
is_stack_empty(stack)

```
If (stack→top = -1)
    return e_true
else
    return e_false
```

Data Structure Stack-Array Implementation

pop(stack,element)

```
if(is_stack_empty(stack))
    return e_false
element = stack→item[stack→top]
(stack→top)--
return e_true
```



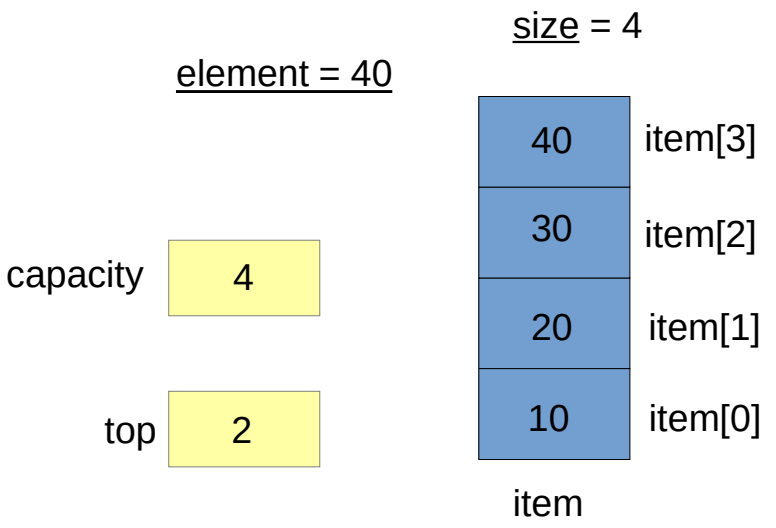
is_stack_empty(stack)

```
If (stack→top = -1)
    return e_true
else
    return e_false
```

Data Structure Stack-Array Implementation

pop(stack,element)

```
if(is_stack_empty(stack))
    return e_false
element = stack→item[stack→top]
(stack→top)--
return e_true
```



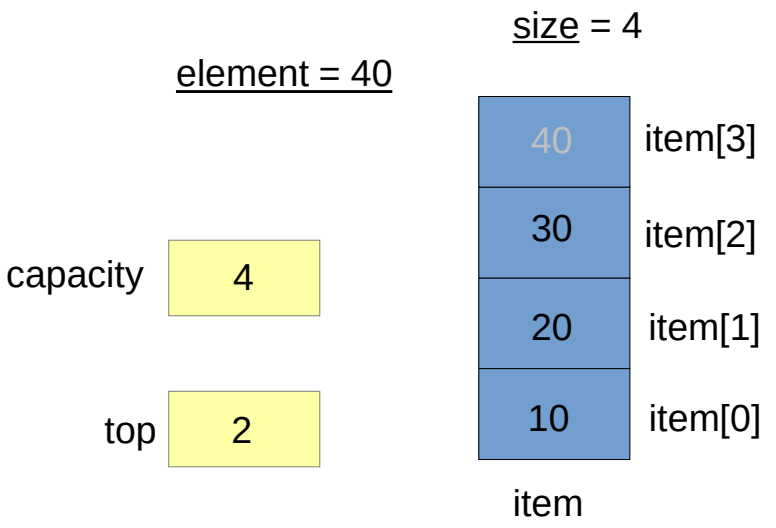
is_stack_empty(stack)

```
If (stack→top = -1)
    return e_true
else
    return e_false
```

Data Structure Stack-Array Implementation

pop(stack,element)

```
if(is_stack_empty(stack))
    return e_false
element = stack→item[stack→top]
(stack→top)--
return e_true
```



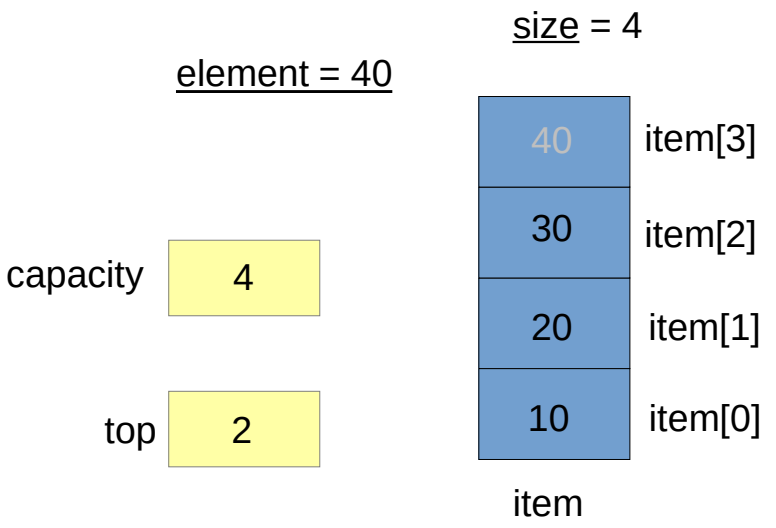
is_stack_empty(stack)

```
If (stack→top = -1)
    return e_true
else
    return e_false
```

Data Structure Stack-Array Implementation

pop(stack,element)

```
if(is_stack_empty(stack))
    return e_false
element = stack→item[stack→top]
(stack→top)--
return e_true
```



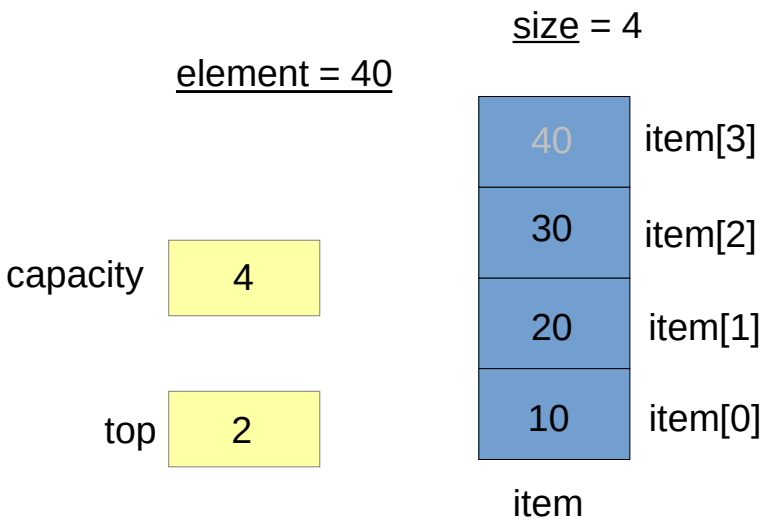
is_stack_empty(stack)

```
If (stack→top = -1)
    return e_true
else
    return e_false
```

Data Structure Stack-Array Implementation

pop(stack,element)

```
if(is_stack_empty(stack))
    return e_false
element = stack→item[stack→top]
(stack→top)--
return e_true
```



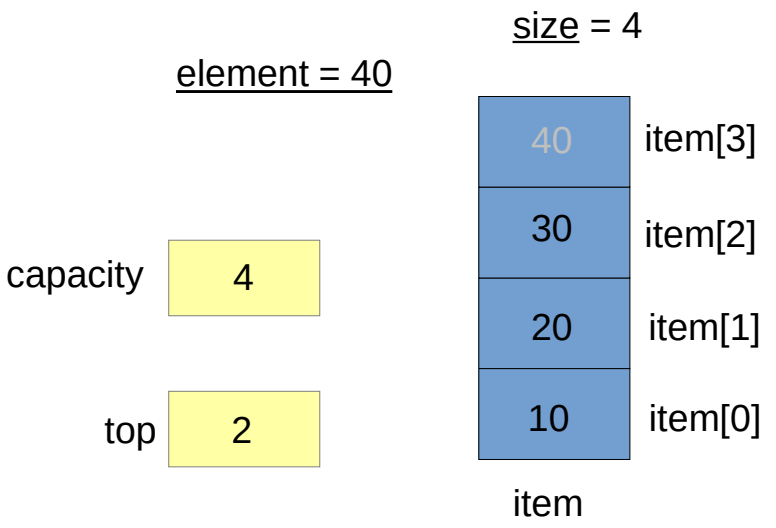
is_stack_empty(stack)

```
If (stack→top = -1)
    return e_true
else
    return e_false
```

Data Structure Stack-Array Implementation

pop(stack,element)

```
if(is_stack_empty(stack))
    return e_false
element = stack→item[stack→top]
(stack→top)--
return e_true
```



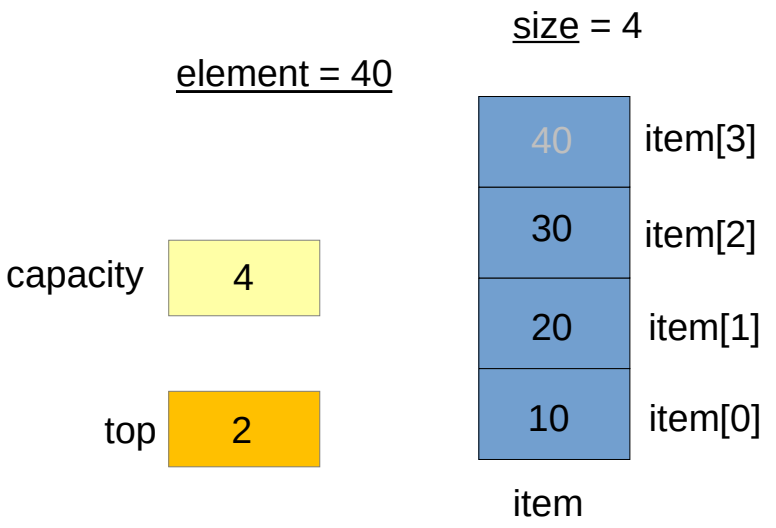
is_stack_empty(stack)

```
If (stack→top = -1)
    return e_true
else
    return e_false
```

Data Structure Stack-Array Implementation

pop(stack,element)

```
if(is_stack_empty(stack))
    return e_false
element = stack→item[stack→top]
(stack→top)--
return e_true
```



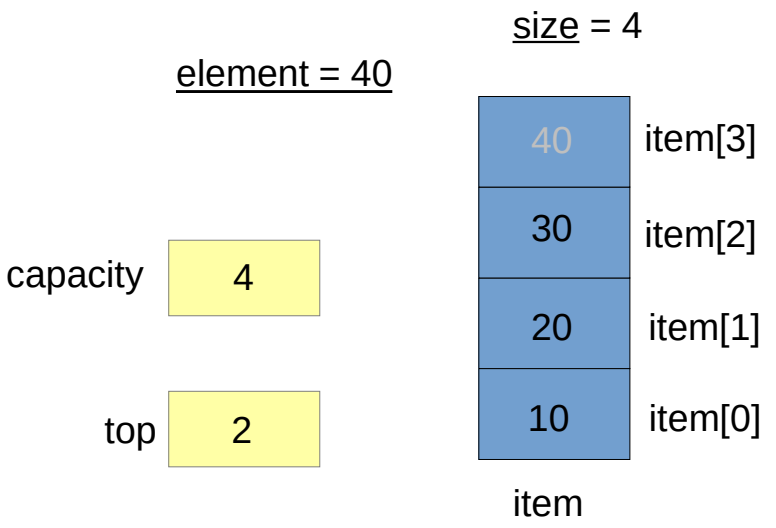
is_stack_empty(stack)

```
If (stack→top = -1)
    return e_true
else
    return e_false
```

Data Structure Stack-Array Implementation

pop(stack,element)

```
if(is_stack_empty(stack))
    return e_false
element = stack→item[stack→top]
(stack→top)--
return e_true
```



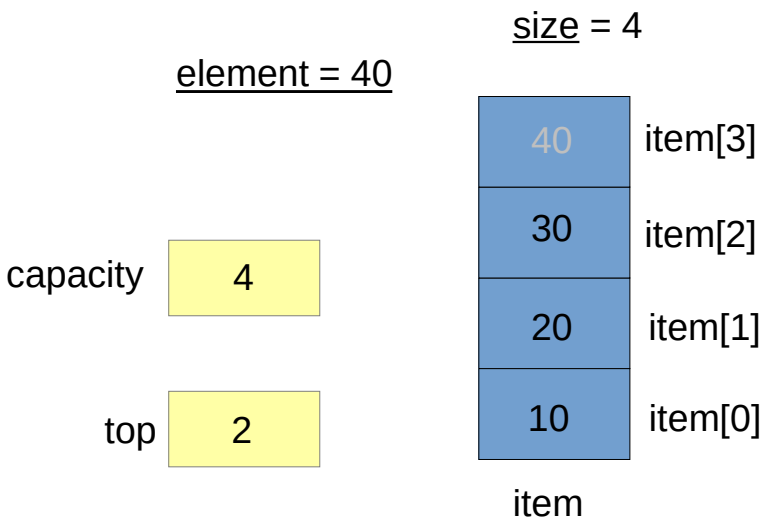
is_stack_empty(stack)

```
If (stack→top = -1)
    return e_true
else
    return e_false
```

Data Structure Stack-Array Implementation

pop(stack,element)

```
if(is_stack_empty(stack))
    return e_false
element = stack→item[stack→top]
(stack→top)--
return e_true
```



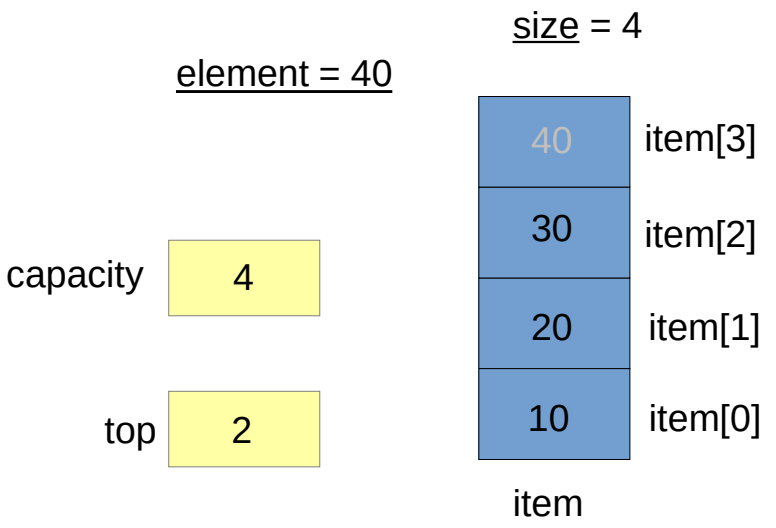
is_stack_empty(stack)

```
If (stack→top = -1)
    return e_true
else
    return e_false
```

Data Structure Stack-Array Implementation

pop(stack,element)

```
if(is_stack_empty(stack))
    return e_false
element = stack→item[stack→top]
(stack→top)--
return e_true
```



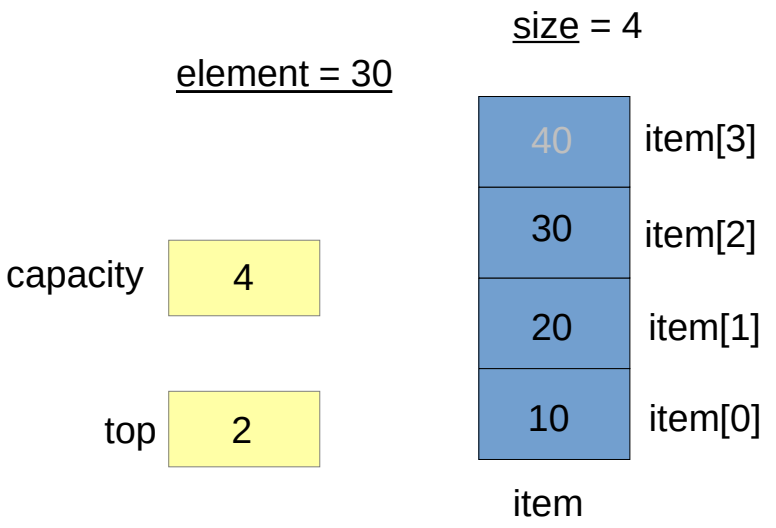
is_stack_empty(stack)

```
If (stack→top = -1)
    return e_true
else
    return e_false
```

Data Structure Stack-Array Implementation

pop(stack,element)

```
if(is_stack_empty(stack))
    return e_false
element = stack→item[stack→top]
(stack→top)--
return e_true
```



is_stack_empty(stack)

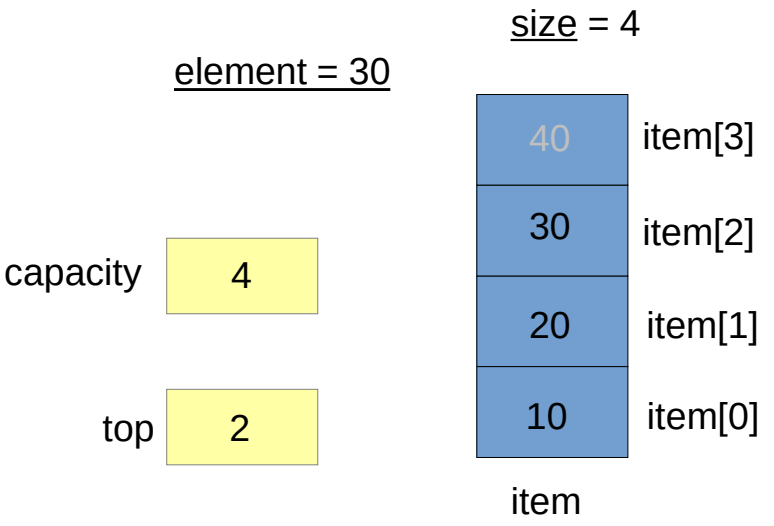
```
If (stack→top = -1)
    return e_true
else
    return e_false
```

EMERTXE

Data Structure Stack-Array Implementation

pop(stack,element)

```
if(is_stack_empty(stack))
    return e_false
element = stack→item[stack→top]
(stack→top)--
return e_true
```



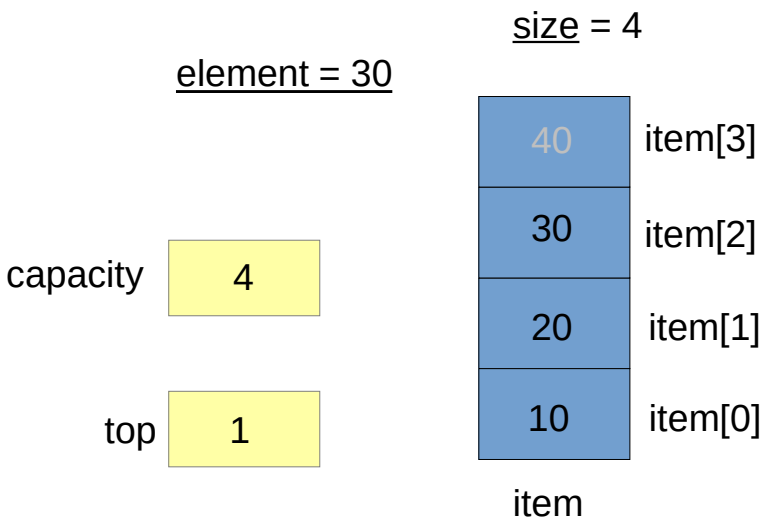
is_stack_empty(stack)

```
If (stack→top = -1)
    return e_true
else
    return e_false
```

Data Structure Stack-Array Implementation

pop(stack,element)

```
if(is_stack_empty(stack))
    return e_false
element = stack→item[stack→top]
(stack→top)--
return e_true
```



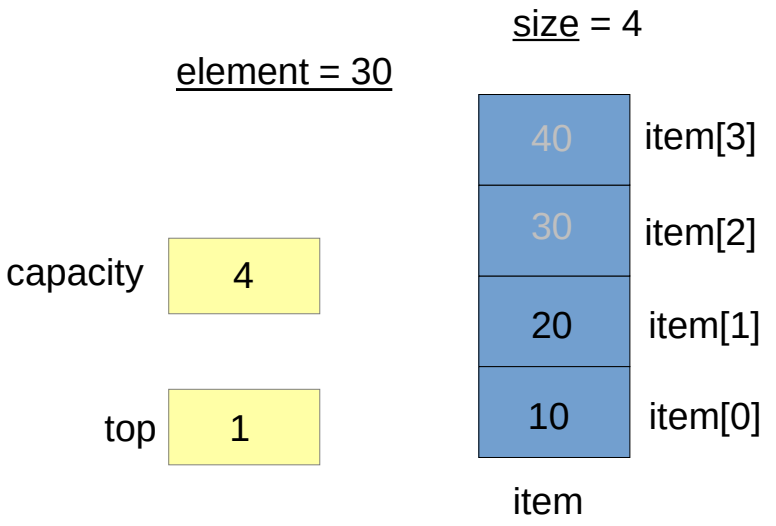
is_stack_empty(stack)

```
If (stack→top = -1)
    return e_true
else
    return e_false
```

Data Structure Stack-Array Implementation

pop(stack,element)

```
if(is_stack_empty(stack))
    return e_false
element = stack→item[stack→top]
(stack→top)--
return e_true
```



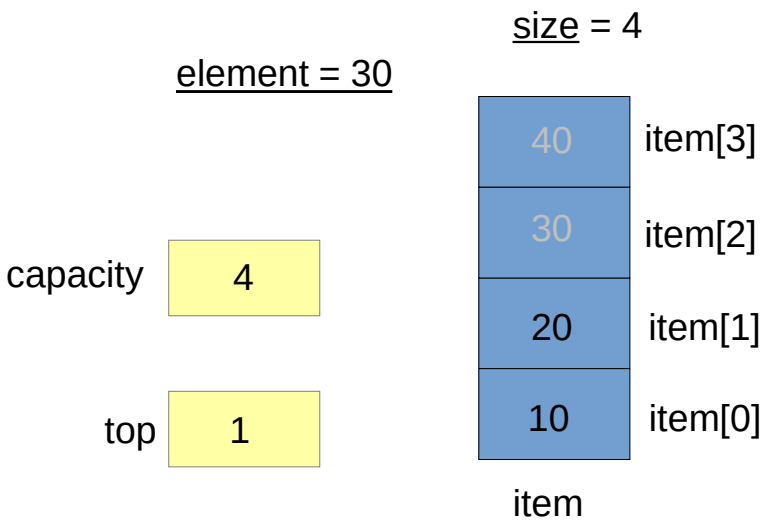
is_stack_empty(stack)

```
If (stack→top = -1)
    return e_true
else
    return e_false
```

Data Structure Stack-Array Implementation

pop(stack,element)

```
if(is_stack_empty(stack))
    return e_false
element = stack→item[stack→top]
(stack→top)--
return e_true
```



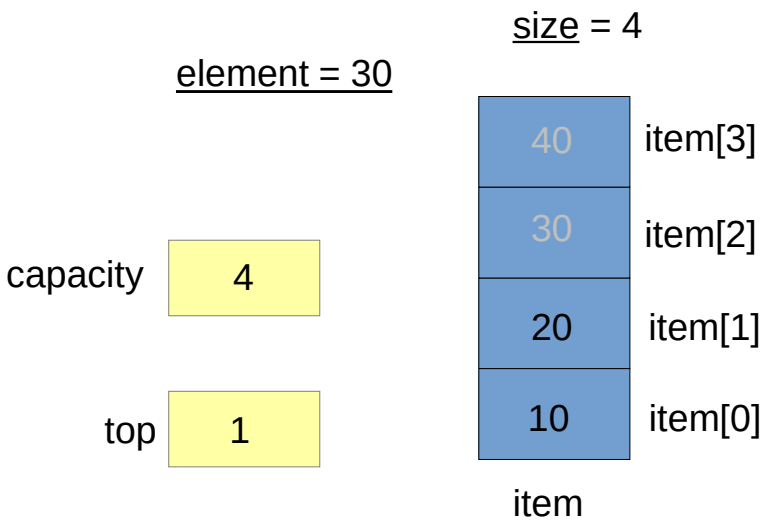
is_stack_empty(stack)

```
If (stack→top = -1)
    return e_true
else
    return e_false
```

Data Structure Stack-Array Implementation

pop(stack,element)

```
if(is_stack_empty(stack))
    return e_false
element = stack→item[stack→top]
(stack→top)--
return e_true
```



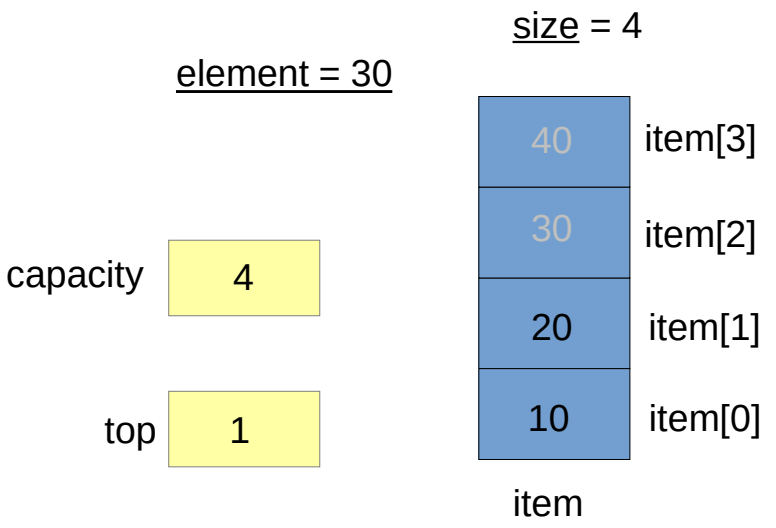
is_stack_empty(stack)

```
If (stack→top = -1)
    return e_true
else
    return e_false
```

Data Structure Stack-Array Implementation

pop(stack,element)

```
if(is_stack_empty(stack))
    return e_false
element = stack→item[stack→top]
(stack→top)--
return e_true
```



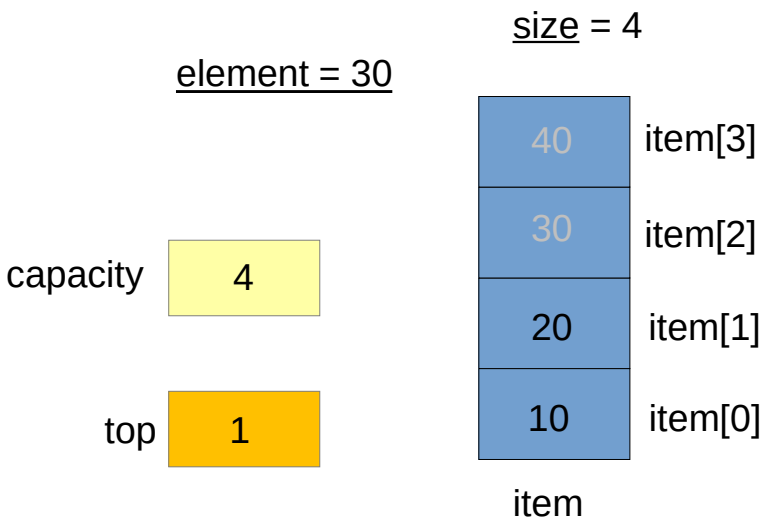
is_stack_empty(stack)

```
If (stack→top = -1)
    return e_true
else
    return e_false
```

Data Structure Stack-Array Implementation

pop(stack,element)

```
if(is_stack_empty(stack))
    return e_false
element = stack→item[stack→top]
(stack→top)--
return e_true
```



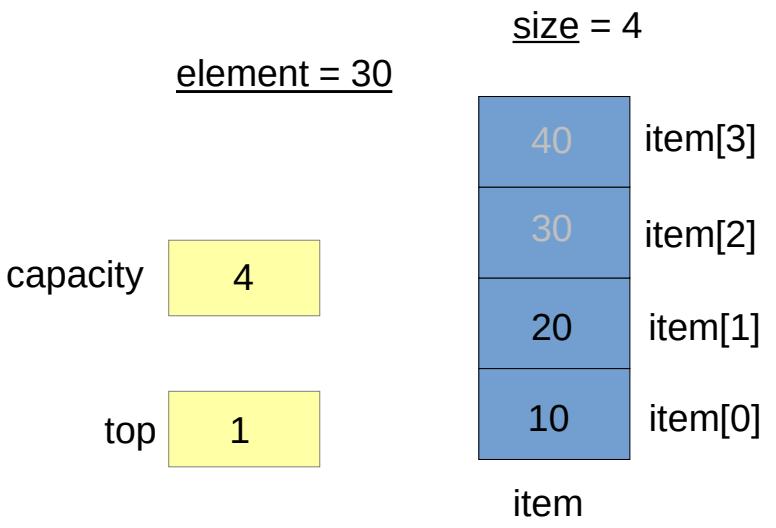
is_stack_empty(stack)

```
If (stack→top = -1)
    return e_true
else
    return e_false
```

Data Structure Stack-Array Implementation

pop(stack,element)

```
if(is_stack_empty(stack))
    return e_false
element = stack→item[stack→top]
(stack→top)--
return e_true
```



is_stack_empty(stack)

```
If (stack→top = -1)
    return e_true
else
    return e_false
```

EMERTXE

Stack – peek(stack,element)