

Tail calls

```
function fact(n, acc) {  
    if (n == 0) {  
        return acc  
    }  
    return fact(n - 1, n * acc)  
}
```

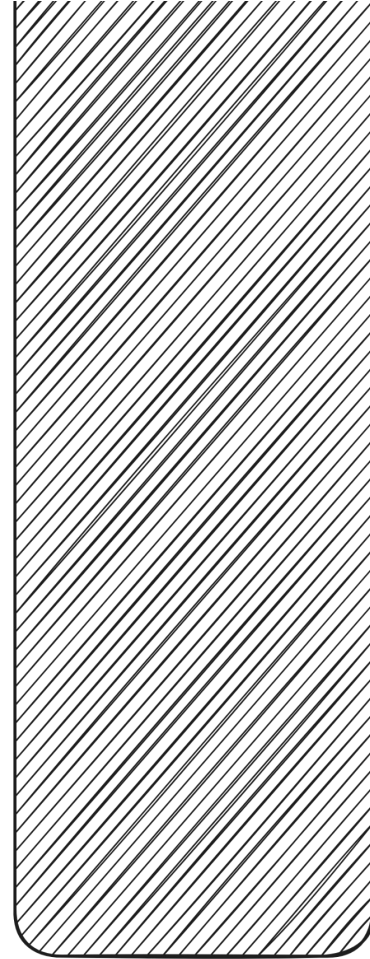
```
function fact(n, acc) {  
  if (n == 0) {  
    return acc  
  }  
  return fact(n - 1, n * acc)  
}
```



*Address 20*

```
function fact(n, acc) {  
  if (n == 0) {  
    return acc  
  }  
20:  return fact(n - 1, n * acc)  
}
```

Execution:



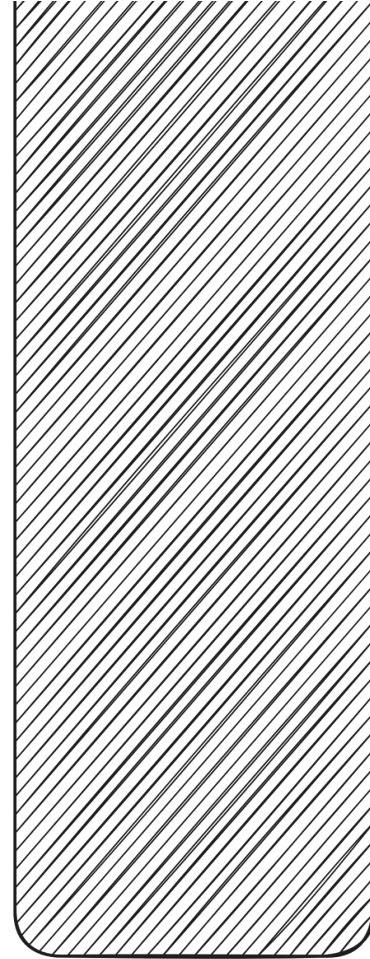

```
function fact(n, acc) {  
  if (n == 0) {  
    return acc  
  }  
  return fact(n - 1, n * acc)  
}
```

20:

Execution:

*Address 10*

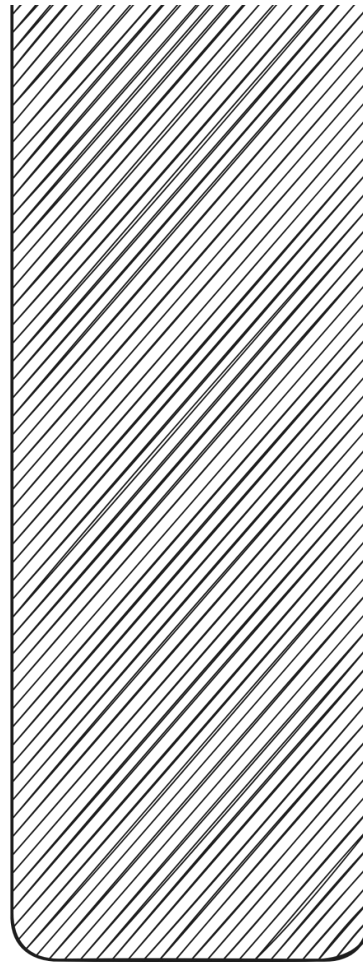
fact(4, 1)



```
function fact(n, acc) {  
  if (n == 0) {  
    return acc  
  }  
20: return fact(n - 1, n * acc)  
}
```

Execution:

10: fact(4, 1)

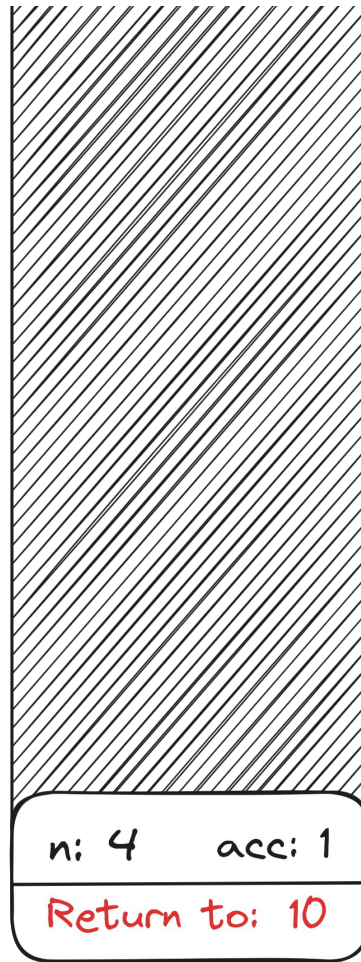


```
function fact(n, acc) {  
    if (n == 0) {  
        return acc  
    }  
    return fact(n - 1, n * acc)  
}
```



Execution:

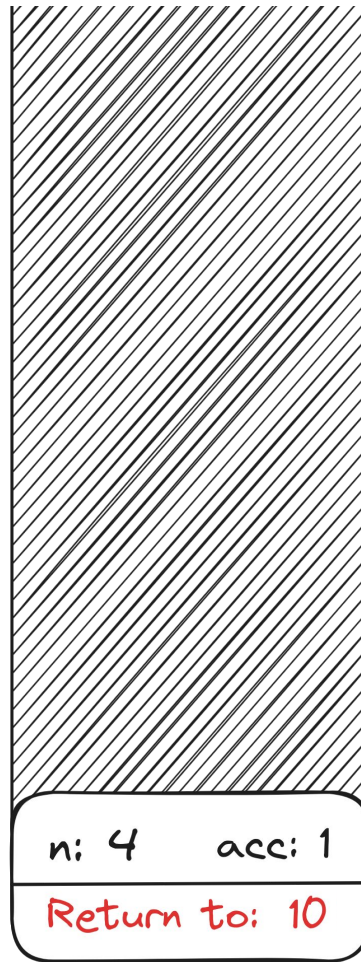
10: fact(4, 1)



```
function fact(n, acc) {  
  if (n == 0) {  
    return acc  
  }  
  20: return fact(n - 1, n * acc) ←  
}
```

Execution:

10: fact(4, 1)



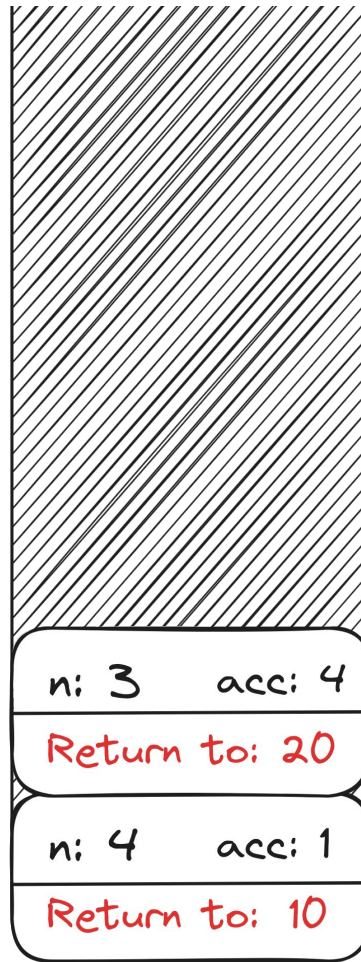


```
function fact(n, acc) {  
  if (n == 0) {  
    return acc  
  }  
20: return fact(n - 1, n * acc)  
}
```



Execution:

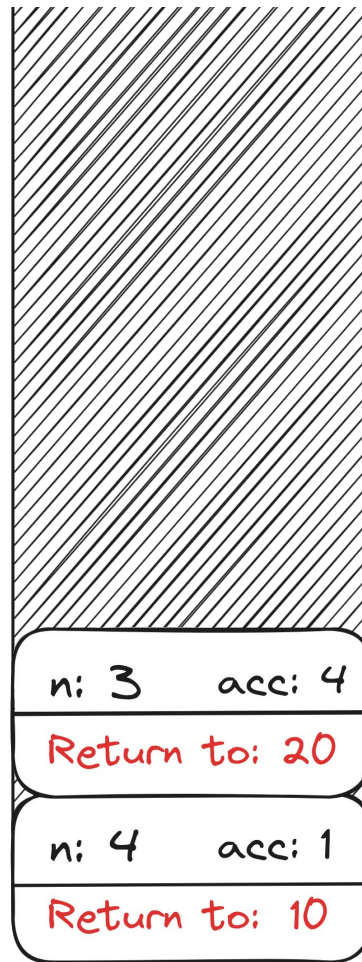
```
10: fact(4, 1)  
  ↳ fact(3, 4)
```



```
function fact(n, acc) {  
  if (n == 0) {  
    return acc  
  }  
20:  return fact(n - 1, n * acc) ←  
}
```

Execution:

10: fact(4, 1)  
    ↪ fact(3, 4)

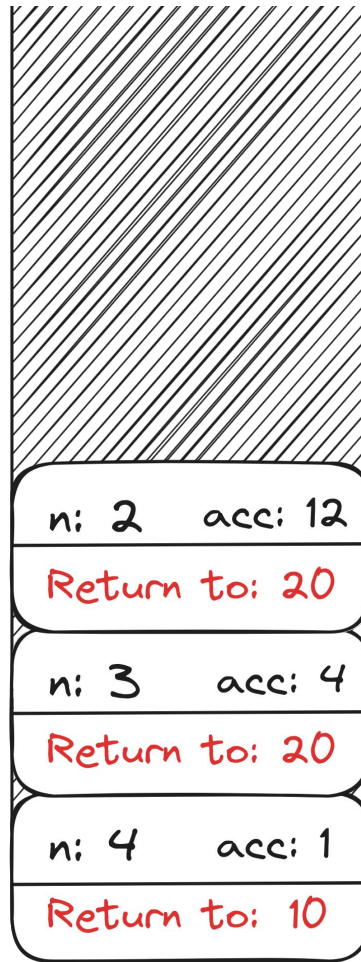


```
function fact(n, acc) {  
    if (n == 0) {  
        return acc  
    }  
    20: return fact(n - 1, n * acc)  
}
```



## Execution:

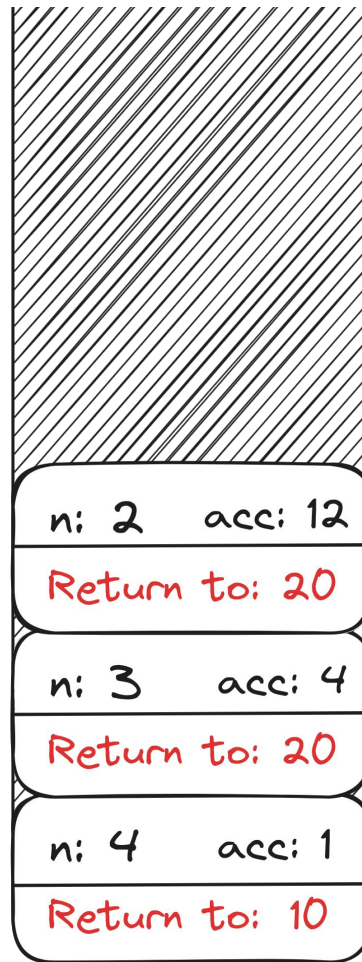
```
10: fact(4, 1)  
    ↪ fact(3, 4)  
        ↪ fact(2, 12)
```



```
function fact(n, acc) {  
    if (n == 0) {  
        return acc  
    }  
    20: return fact(n - 1, n * acc) ←  
}
```

## Execution:

10: fact(4, 1)  
    ↪ fact(3, 4)  
        ↪ fact(2, 12)

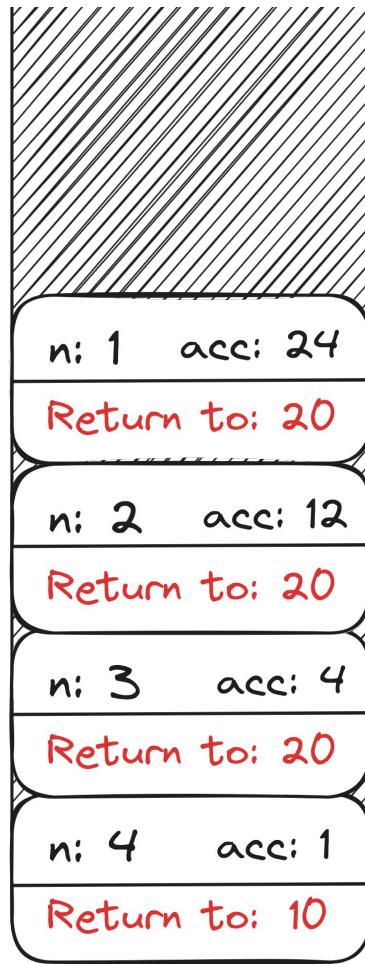


```
function fact(n, acc) {  
    if (n == 0) {  
        return acc  
    }  
    20: return fact(n - 1, n * acc)  
}
```



## Execution:

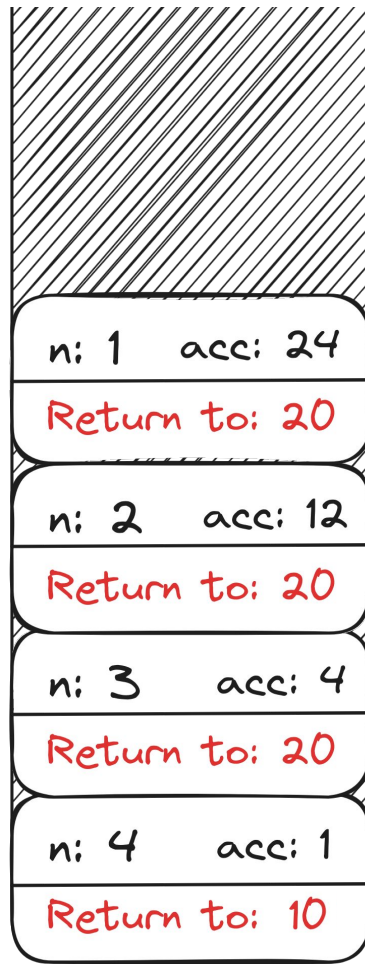
```
10: fact(4, 1)  
    ↪ fact(3, 4)  
        ↪ fact(2, 12)  
            ↪ fact(1, 24)
```



```
function fact(n, acc) {  
    if (n == 0) {  
        return acc  
    }  
    20: return fact(n - 1, n * acc) ←  
}
```

## Execution:

```
10: fact(4, 1)  
    ↪ fact(3, 4)  
        ↪ fact(2, 12)  
            ↪ fact(1, 24)
```



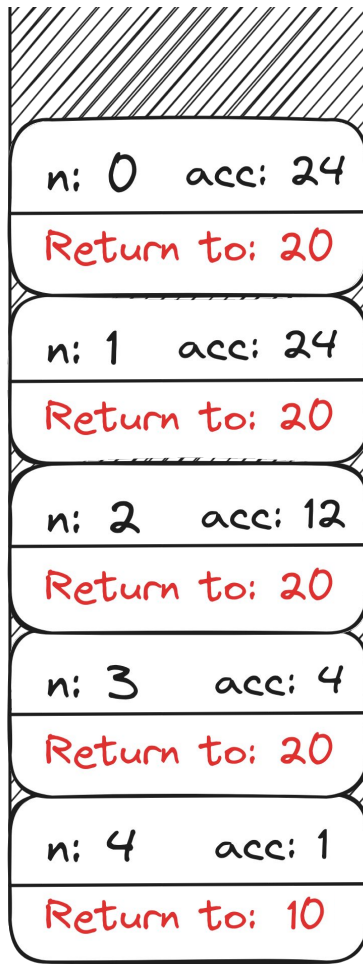
```
function fact(n, acc) {  
    if (n == 0) {  
        return acc  
    }
```



```
20: return fact(n - 1, n * acc)  
}
```

## Execution:

```
10: fact(4, 1)  
    ↪ fact(3, 4)  
        ↪ fact(2, 12)  
            ↪ fact(1, 24)  
                ↪ fact(0, 24)
```

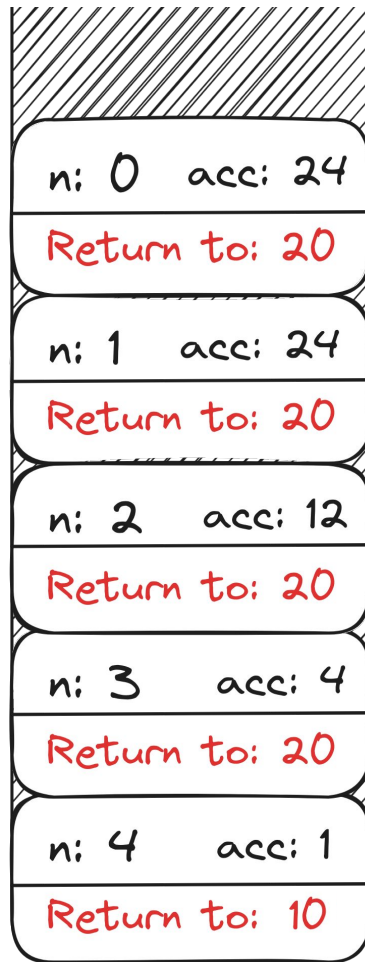


```
function fact(n, acc) {  
    if (n == 0) {  
        return acc  
    }  
    return fact(n - 1, n * acc)  
}
```



## Execution:

10: fact(4, 1)  
    ↪ fact(3, 4)  
        ↪ fact(2, 12)  
            ↪ fact(1, 24)  
                ↪ fact(0, 24) = 24

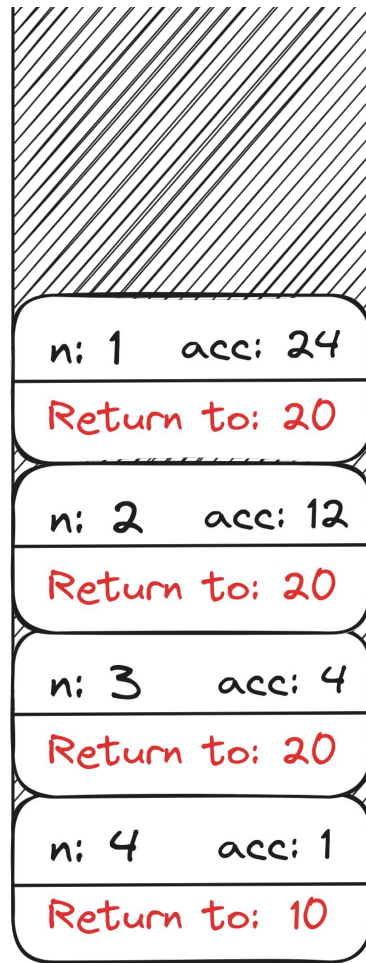




```
function fact(n, acc) {  
  if (n == 0) {  
    return acc  
  }  
  20: return fact(n - 1, n * acc) ←  
}
```

## Execution:

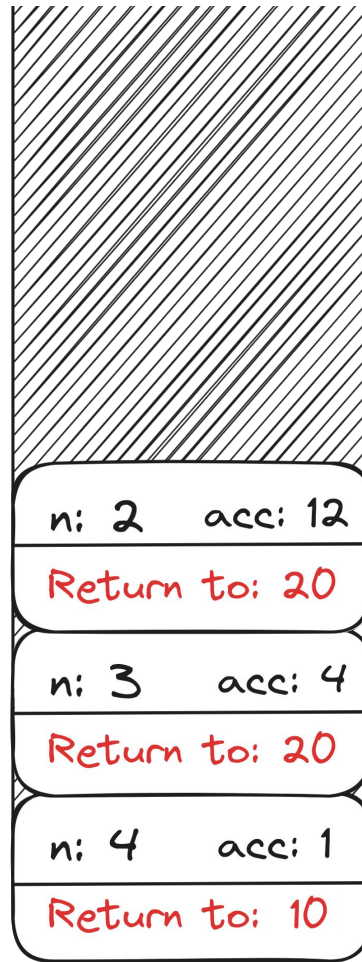
```
10: fact(4, 1)  
  ↳ fact(3, 4)  
    ↳ fact(2, 12)  
      ↳ fact(1, 24) = 24
```



```
function fact(n, acc) {  
    if (n == 0) {  
        return acc  
    }  
    20: return fact(n - 1, n * acc) ←  
}
```

## Execution:

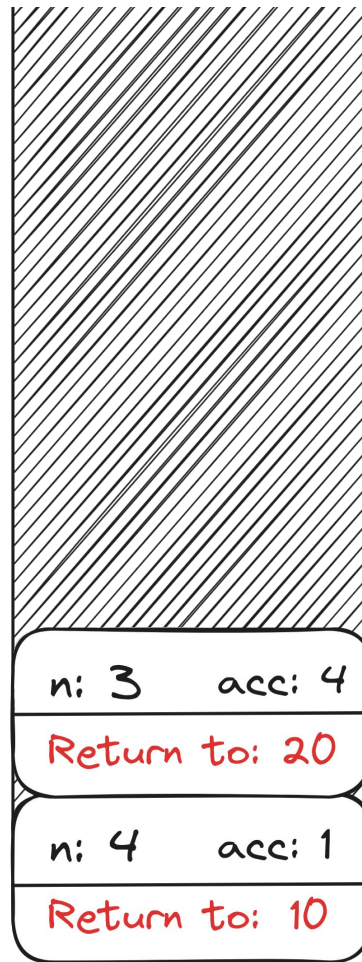
10: fact(4, 1)  
    ↪ fact(3, 4)  
        ↪ fact(2, 12) = 24



```
function fact(n, acc) {  
  if (n == 0) {  
    return acc  
  }  
  20: return fact(n - 1, n * acc) ←  
}
```

Execution:

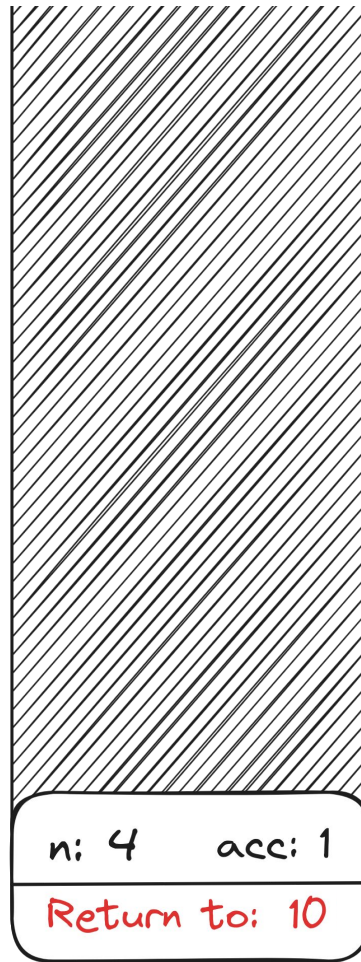
10: fact(4, 1)  
    ↪ fact(3, 4) = 24



```
function fact(n, acc) {  
  if (n == 0) {  
    return acc  
  }  
20:  return fact(n - 1, n * acc) ←  
}
```

Execution:

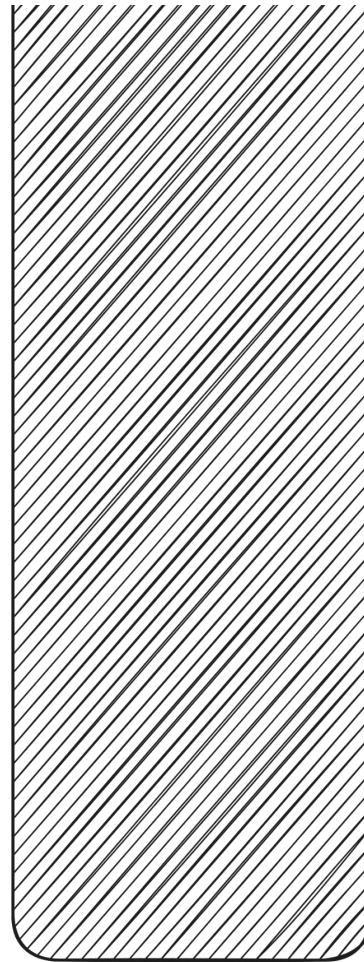
10: fact(4, 1) = 24



```
function fact(n, acc) {  
    if (n == 0) {  
        return acc  
    }  
    return fact(n - 1, n * acc)  
}
```

Execution:

10: 24



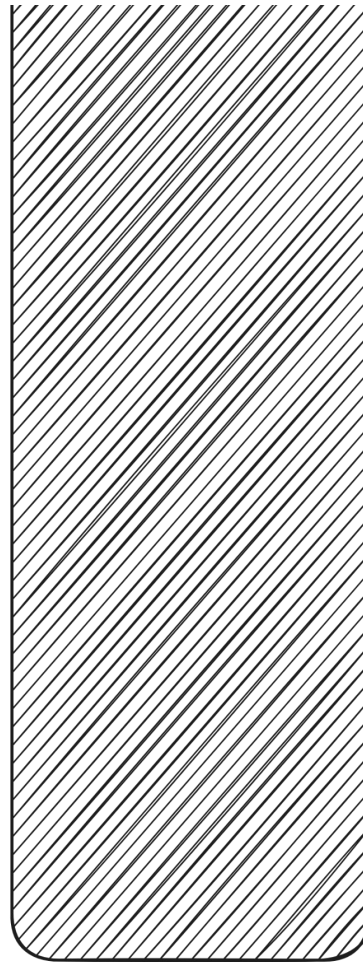
# Questions

- Time?
- Space?
- Can we do better?

```
function fact(n, acc) {  
    if (n == 0) {  
        return acc  
    }  
20:   return fact(n - 1, n * acc)  
}
```

Execution:

10: fact(4, 1)

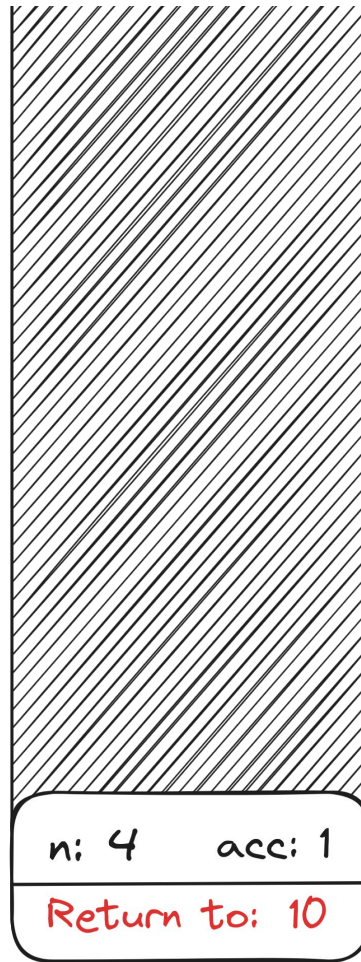


```
function fact(n, acc) {  
  if (n == 0) {  
    return acc  
  }  
20:  return fact(n - 1, n * acc)  
}
```



Execution:

10: fact(4, 1)

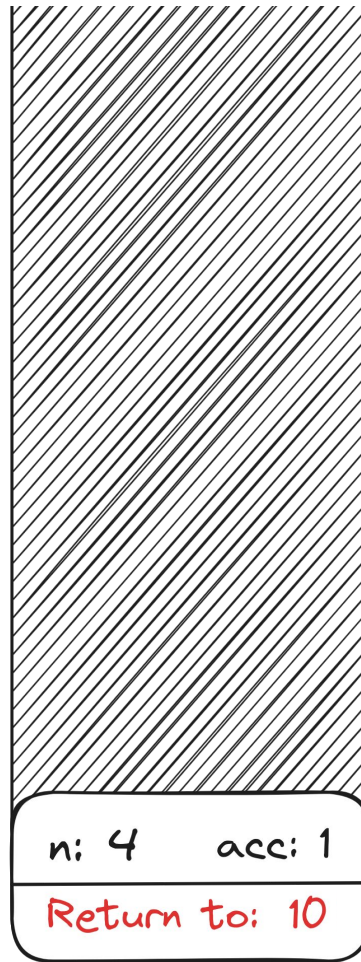




```
function fact(n, acc) {  
  if (n == 0) {  
    return acc  
  }  
20:  return fact(n - 1, n * acc) ←  
}
```

Execution:

10: fact(4, 1)

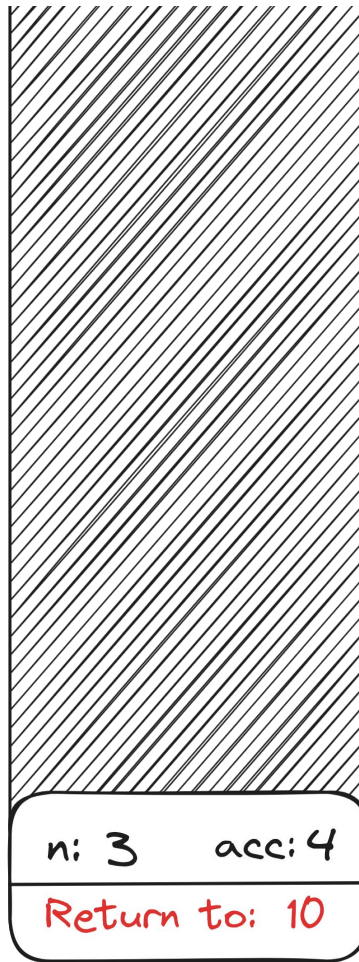


```
function fact(n, acc) {  
    if (n == 0) {  
        return acc  
    }  
20: return fact(n - 1, n * acc)  
}
```



Execution:

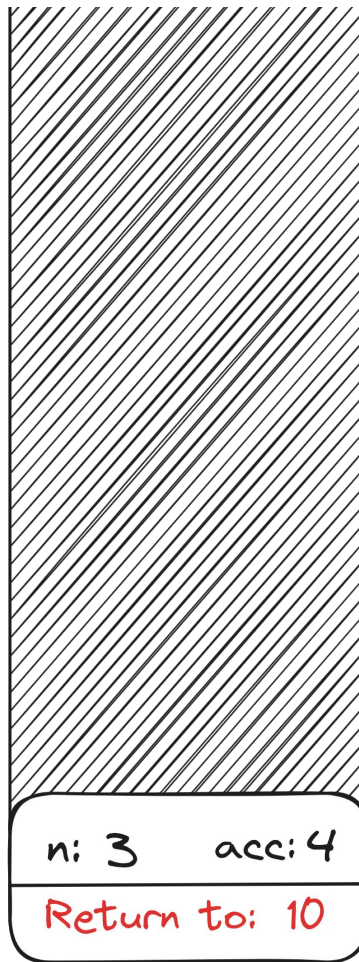
10: fact(4, 1)  
    ↪ fact(3, 4)



```
function fact(n, acc) {  
  if (n == 0) {  
    return acc  
  }  
20:  return fact(n - 1, n * acc) ←  
}
```

Execution:

10: fact(4, 1)  
    ↪ fact(3, 4)



```
function fact(n, acc) {  
    if (n == 0) {  
        return acc  
    }  
    return fact(n - 1, n * acc)  
}
```



## Execution:

```
10: fact(4, 1)  
    ↪ fact(3, 4)  
        ↪ fact(2, 12)
```

n: 2    acc: 12

Return to: 10

```
function fact(n, acc) {  
  if (n == 0) {  
    return acc  
  }  
20:  return fact(n - 1, n * acc) ←  
}
```

Execution:

```
10: fact(4, 1)  
  ↪ fact(3, 4)  
    ↪ fact(2, 12)
```

n: 2    acc: 12

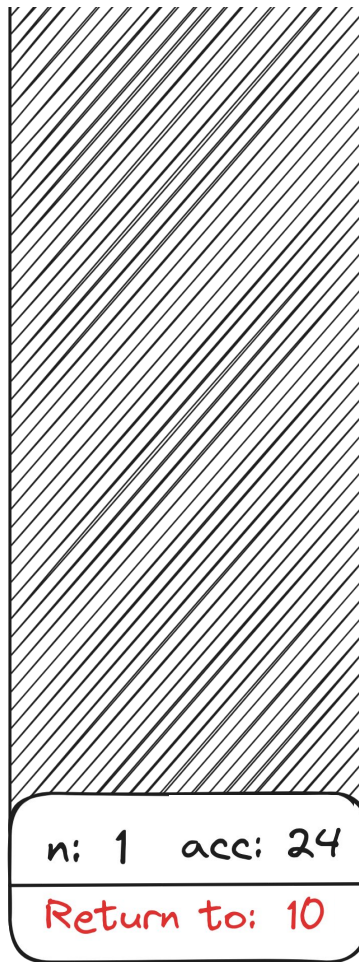
Return to: 10

```
function fact(n, acc) {  
    if (n == 0) {  
        return acc  
    }  
20: return fact(n - 1, n * acc)  
}
```



## Execution:

```
10: fact(4, 1)  
    ↪ fact(3, 4)  
        ↪ fact(2, 12)  
            ↪ fact(1, 24)
```



```
function fact(n, acc) {  
    if (n == 0) {  
        return acc  
    }  
    20: return fact(n - 1, n * acc) ←  
}
```

## Execution:

```
10: fact(4, 1)  
    ↪ fact(3, 4)  
        ↪ fact(2, 12)  
            ↪ fact(1, 24)
```

n: 1 acc: 24

Return to: 10

```
function fact(n, acc) {  
    if (n == 0) {  
        return acc  
    }  
    return fact(n - 1, n * acc)  
}
```



## Execution:

```
10: fact(4, 1)  
    ↪ fact(3, 4)  
        ↪ fact(2, 12)  
            ↪ fact(1, 24)  
                ↪ fact(0, 24)
```

n: 0 acc: 24

Return to: 10

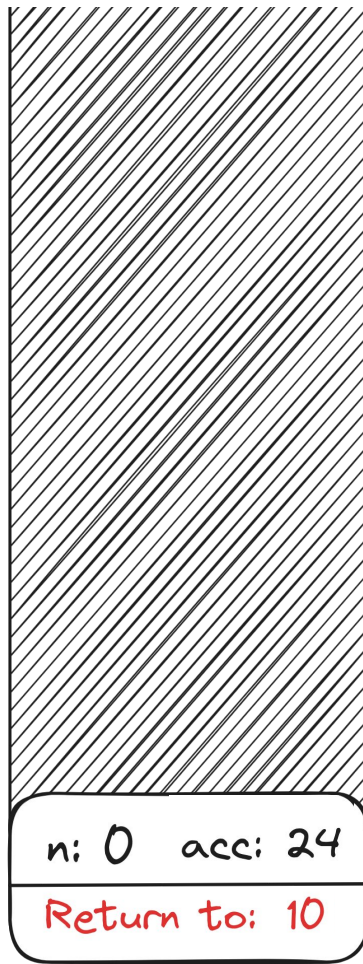


```
function fact(n, acc) {  
  if (n == 0) {  
    return acc  
  }  
20: return fact(n - 1, n * acc)  
}
```



## Execution:

```
10: fact(4, 1)  
  ↳ fact(3, 4)  
    ↳ fact(2, 12)  
      ↳ fact(1, 24)  
        ↳ fact(0, 24) = 24
```



```
function fact(n, acc) {  
    if (n == 0) {  
        return acc  
    }  
    return fact(n - 1, n * acc)  
}
```

Execution:

10: 24

