

SC101

Week5

```
def main():
```

```
    def factorial(n):
```

```
        return n * factorial(n-1)
```


```
    factorial(5)
```

```
    print(factorial(5))
```

```
    return
```

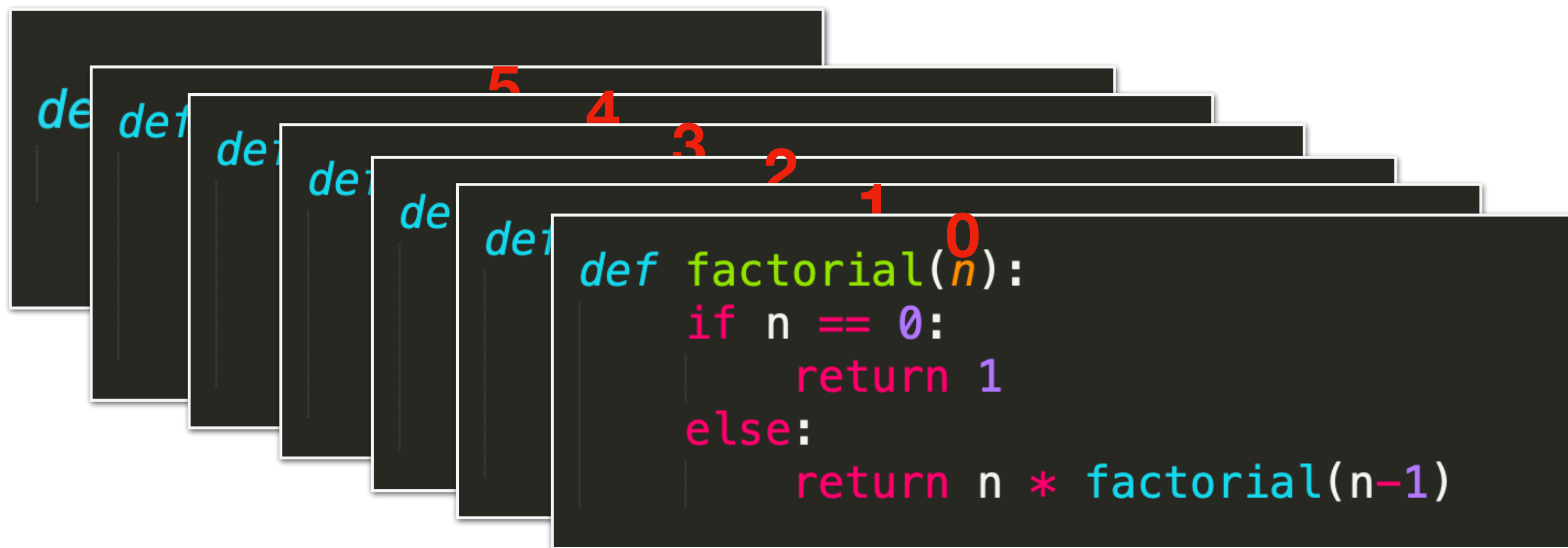
```
def factorial(n):  
    return n * factorial(n-1)
```

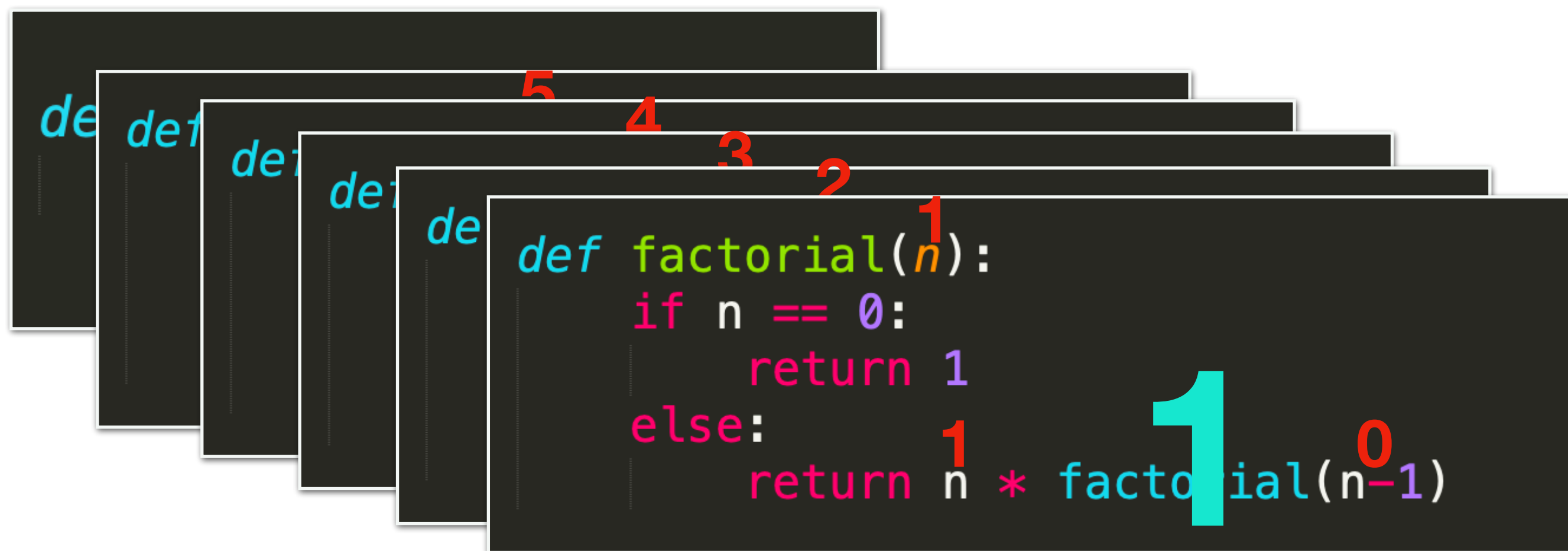
Stack Overflow

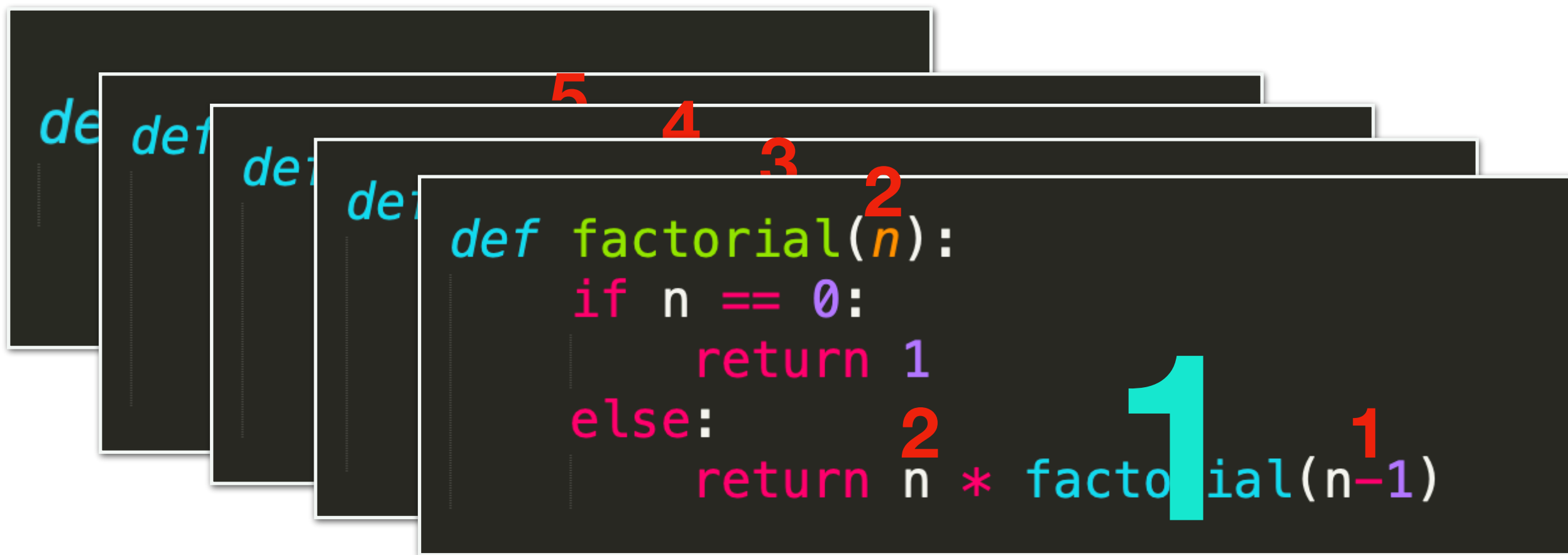


```
def main():
    ...
    d
    d
    d
    d
    d
```

```
def factorial(n):  
    return n * factorial(n-1)
```





```
def factorial(n):  
    if n == 0:  
        return 1  
    else:  
        return n * factorial(n-1)
```


def

def

def factorial(⁵⁴*n*):

if n == 0:

return 1

else:

return ⁴n * factorial(³*n*-1)

6

```
def factorial(5n):  
    if n == 0:  
        return 1  
    else:  
        return 5n * factorial(4n-1)
```

24

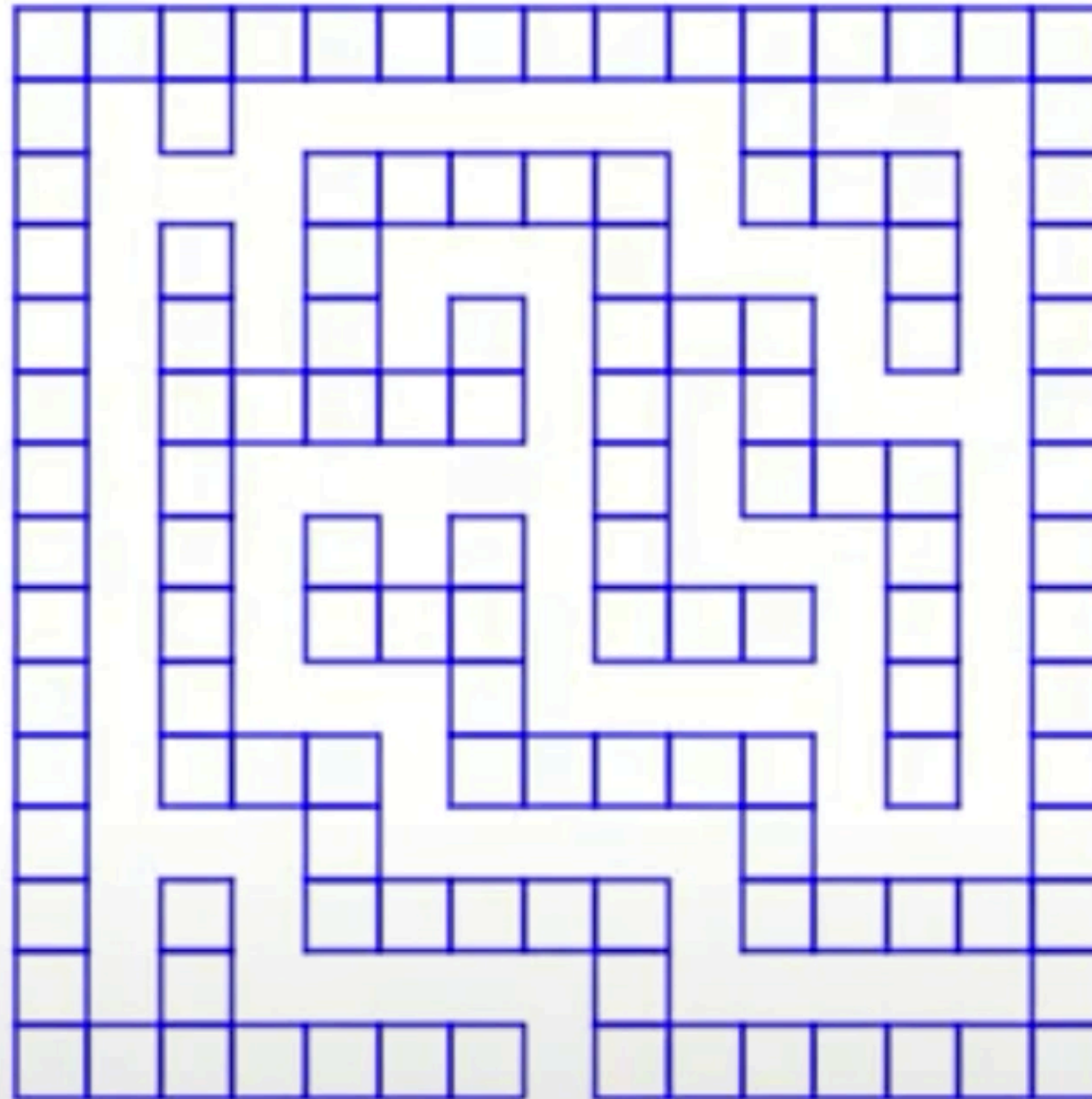
```
def main():  
    print(factorial(5))
```

120

Self Similarity

Backtracking

Load ... ☒ Animation?



r04 c11

```
def main():  
    permutation([1, 2, 3])
```

```
def main():  
    permutation([1, 2, 3])
```



```
def main():
```

```
    def permutation(lst):
```

```
        permutation_helper(lst, [], len(lst))
```

```
def main():
```

```
    def permutation(lst):
```

```
        permutation_helper(lst, [], len(lst))
```

```
def main():
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
    if len(current_lst) == ans_len:
```

```
        print(current_lst)
```

```
    else:
```

```
        for num in lst:
```

```
            if num in current_lst:
```

```
                pass
```

```
            else:
```

```
                current_lst.append(num)
```

```
                permutation_helper(lst, current_lst, ans_len)
```

```
                current_lst.pop()
```

```
def main():
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
    if len(current_lst) == ans_len:
```

```
        print(current_lst)
```

```
    else:
```

```
        for num in lst:
```

```
            if num in current_lst:
```

```
                pass
```

```
            else:
```

```
                current_lst.append(num)
```

```
                permutation_helper(lst, current_lst, ans_len)
```

```
                current_lst.pop()
```

lst = [1, 2, 3]

current_lst = []

ans_len = 3

```
def main():
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
    if len(current_lst) == ans_len:
```

```
        print(current_lst)
```

```
    else:
```

```
        for num in lst:
```

```
            if num in current_lst:
```

```
                pass
```

```
            else:
```

```
                current_lst.append(num)
```

```
                permutation_helper(lst, current_lst, ans_len)
```

```
                current_lst.pop()
```

lst = [1, 2, 3]

current_lst = []

ans_len = 3

```
def main():
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
    if len(current_lst) == ans_len:
```

```
        print(current_lst)
```

```
    else:
```

```
        for num in lst:
```

```
            if num in current_lst:
```

```
                pass
```

```
            else:
```

```
                current_lst.append(num)
```

```
                permutation_helper(lst, current_lst, ans_len)
```

```
                current_lst.pop()
```

lst = [1, 2, 3]

current_lst = []

ans_len = 3

```
def main():
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
    if len(current_lst) == ans_len:
```

```
        print(current_lst)
```

```
    else:
```

```
        for num in lst:
```

```
            if num in current_lst:
```

```
                pass
```

```
            else:
```

```
                current_lst.append(num)
```

```
                permutation_helper(lst, current_lst, ans_len)
```

```
                current_lst.pop()
```

num = 1

lst = [1, 2, 3]

current_lst = []

ans_len = 3

```
def main():
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
    if len(current_lst) == ans_len:
```

```
        print(current_lst)
```

```
    else:
```

```
        for num in lst:
```

```
            if num in current_lst:
```

```
                pass
```

```
            else:
```

```
                current_lst.append(num)
```

```
                permutation_helper(lst, current_lst, ans_len)
```

```
                current_lst.pop()
```

num = 1

lst = [1, 2, 3]

current_lst = []

ans_len = 3


```
def main():
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
    if len(current_lst) == ans_len:
```

```
        print(current_lst)
```

```
    else:
```

```
        for num in lst:
```

```
            if num in current_lst:
```

```
                pass
```

```
            else:
```

```
                current_lst.append(num)
```

```
                permutation_helper(lst, current_lst, ans_len)
```

```
                current_lst.pop()
```

num = 1

lst = [1, 2, 3]

current_lst = []

ans_len = 3

```
def main():
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
    if len(current_lst) == ans_len:
```

```
        print(current_lst)
```

```
    else:
```

```
        for num in lst:
```

```
            if num in current_lst:
```

```
                pass
```

```
            else:
```

```
                current_lst.append(num)
```

```
                permutation_helper(lst, current_lst, ans_len)
```

```
                current_lst.pop()
```

num = 1

lst = [1, 2, 3]

current_lst = []

ans_len = 3

```
def main():
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
    if len(current_lst) == ans_len:
```

```
        print(current_lst)
```

```
    else:
```

```
        for num in lst:
```

```
            if num in current_lst:
```

```
                pass
```

```
            else:
```

```
                current_lst.append(num)
```

```
                permutation_helper(lst, current_lst, ans_len)
```

```
                current_lst.pop()
```

num = 1

lst = [1, 2, 3]

current_lst = [1]

ans_len = 3

```
def main():
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
    if len(current_lst) == ans_len:
```

```
        print(current_lst)
```

```
    else:
```

```
        for num in lst:
```

```
            if num in current_lst:
```

```
                pass
```

```
            else:
```

```
                current_lst.append(num)
```

```
                permutation_helper(lst, current_lst, ans_len)
```

```
                current_lst.pop()
```

num = 1

lst = [1, 2, 3]

current_lst = [1]

ans_len = 3

```
def main():
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
    if len(current_lst) == ans_len:
```

```
        print(current_lst)
```

```
    else:
```

```
        for num in lst:
```

```
            if num in current_lst:
```

```
                pass
```

```
            else:
```

```
                current_lst.append(num)
```

```
                permutation_helper(lst, current_lst, ans_len)
```

```
                current_lst.pop()
```

lst = [1, 2, 3]

current_lst = [1]

ans_len = 3

```
def main():
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
    if len(current_lst) == ans_len:
```

```
        print(current_lst)
```

```
    else:
```

```
        for num in lst:
```

```
            if num in current_lst:
```

```
                pass
```

```
            else:
```

```
                current_lst.append(num)
```

```
                permutation_helper(lst, current_lst, ans_len)
```

```
                current_lst.pop()
```

lst = [1, 2, 3]

current_lst = [1]

ans_len = 3

```
def main():
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
    if len(current_lst) == ans_len:
```

```
        print(current_lst)
```

```
    else:
```

```
        for num in lst:
```

```
            if num in current_lst:
```

```
                pass
```

```
            else:
```

```
                current_lst.append(num)
```

```
                permutation_helper(lst, current_lst, ans_len)
```

```
                current_lst.pop()
```

lst = [1, 2, 3]

current_lst = [1]

ans_len = 3

```
def main():
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
    if len(current_lst) == ans_len:
```

```
        print(current_lst)
```

```
    else:
```

```
        for num in lst:
```

num = 1

```
            if num in current_lst:
```

```
                pass
```

```
            else:
```

```
                current_lst.append(num)
```

```
                permutation_helper(lst, current_lst, ans_len)
```

```
                current_lst.pop()
```

lst = [1, 2, 3]

current_lst = [1]

ans_len = 3


```
def main():
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
    if len(current_lst) == ans_len:
```

```
        print(current_lst)
```

```
    else:
```

```
        for num in lst:
```

```
            if num in current_lst:
```

```
                pass
```

```
            else:
```

```
                current_lst.append(num)
```

```
                permutation_helper(lst, current_lst, ans_len)
```

```
                current_lst.pop()
```

lst = [1, 2, 3]

current_lst = [1]

ans_len = 3

num = 1

```
def main():
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
    if len(current_lst) == ans_len:
```

```
        print(current_lst)
```

```
    else:
```

```
        for num in lst:
```

```
            if num in current_lst:
```

```
                pass
```

```
            else:
```

```
                current_lst.append(num)
```

```
                permutation_helper(lst, current_lst, ans_len)
```

```
                current_lst.pop()
```

lst = [1, 2, 3]

current_lst = [1]

ans_len = 3

num = 1

```
def main():
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
    if len(current_lst) == ans_len:
```

```
        print(current_lst)
```

```
    else:
```

```
        for num in lst:
```

num = 2

```
            if num in current_lst:
```

```
                pass
```

```
            else:
```

```
                current_lst.append(num)
```

```
                permutation_helper(lst, current_lst, ans_len)
```

```
                current_lst.pop()
```

lst = [1, 2, 3]

current_lst = [1]

ans_len = 3

```
def main():
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
    if len(current_lst) == ans_len:
```

```
        print(current_lst)
```

```
    else:
```

```
        for num in lst:
```

```
            if num in current_lst:
```

```
                pass
```

```
            else:
```

```
                current_lst.append(num)
```

```
                permutation_helper(lst, current_lst, ans_len)
```

```
                current_lst.pop()
```

lst = [1, 2, 3]

current_lst = [1]

ans_len = 3

num = 2

```
def main():
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
    if len(current_lst) == ans_len:
```

```
        print(current_lst)
```

```
    else:
```

```
        for num in lst:
```

num = 2

```
            if num in current_lst:
```

```
                pass
```

```
            else:
```

```
                current_lst.append(num)
```

```
                permutation_helper(lst, current_lst, ans_len)
```

```
                current_lst.pop()
```

lst = [1, 2, 3]

current_lst = [1]

ans_len = 3

```
def main():
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
    if len(current_lst) == ans_len:
```

```
        print(current_lst)
```

```
    else:
```

```
        for num in lst:
```

num = 2

```
            if num in current_lst:
```

```
                pass
```

```
            else:
```

```
                current_lst.append(num)
```

```
                permutation_helper(lst, current_lst, ans_len)
```

```
                current_lst.pop()
```

lst = [1, 2, 3]

current_lst = [1, 2]

ans_len = 3

```
def main():
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
    if len(current_lst) == ans_len:
```

```
        print(current_lst)
```

```
    else:
```

```
        for num in lst:
```

num = 2

```
            if num in current_lst:
```

```
                pass
```

```
            else:
```

```
                current_lst.append(num)
```

```
                permutation_helper(lst, current_lst, ans_len)
```

```
                current_lst.pop()
```

lst = [1, 2, 3]

current_lst = [1, 2]

ans_len = 3


```
def main():
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
    if len(current_lst) == ans_len:
```

```
        print(current_lst)
```

```
    else:
```

```
        for num in lst:
```

```
            if num in current_lst:
```

```
                pass
```

```
            else:
```

```
                current_lst.append(num)
```

```
                permutation_helper(lst, current_lst, ans_len)
```

```
                current_lst.pop()
```

lst = [1, 2, 3]

current_lst = [1, 2]

ans_len = 3


```
def main():
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
if len(current_lst) == ans_len:
```

```
    print(current_lst)
```

```
else:
```

```
    for num in lst:
```

```
        if num in current_lst:
```

```
            pass
```

```
        else:
```

```
            current_lst.append(num)
```

```
            permutation_helper(lst, current_lst, ans_len)
```

```
            current_lst.pop()
```

lst = [1, 2, 3]

current_lst = [1, 2]

ans_len = 3

```
def main():
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
    if len(current_lst) == ans_len:
```

```
        print(current_lst)
```

```
    else:
```

```
        for num in lst:
```

```
            if num in current_lst:
```

```
                pass
```

```
            else:
```

```
                current_lst.append(num)
```

```
                permutation_helper(lst, current_lst, ans_len)
```

```
                current_lst.pop()
```

lst = [1, 2, 3]

current_lst = [1, 2]

ans_len = 3

```
def main():
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
    if len(current_lst) == ans_len:
```

```
        print(current_lst)
```

```
    else:
```

```
        for num in lst:
```

```
            if num in current_lst:
```

```
                pass
```

```
            else:
```

```
                current_lst.append(num)
```

```
                permutation_helper(lst, current_lst, ans_len)
```

```
                current_lst.pop()
```

lst = [1, 2, 3]

current_lst = [1, 2]

ans_len = 3

num = 1

```
def main():
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
    if len(current_lst) == ans_len:
```

```
        print(current_lst)
```

```
    else:
```

```
        for num in lst:
```

```
            if num in current_lst:
```

```
                pass
```

```
            else:
```

```
                current_lst.append(num)
```

```
                permutation_helper(lst, current_lst, ans_len)
```

```
                current_lst.pop()
```

lst = [1, 2, 3]

current_lst = [1, 2]

ans_len = 3

num = 1

```
def main():
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
    if len(current_lst) == ans_len:
```

```
        print(current_lst)
```

```
    else:
```

```
        for num in lst:
```

```
            if num in current_lst:
```

```
                pass
```

```
            else:
```

```
                current_lst.append(num)
```

```
                permutation_helper(lst, current_lst, ans_len)
```

```
                current_lst.pop()
```

lst = [1, 2, 3]

current_lst = [1, 2]

ans_len = 3

num = 1

```
def main():
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
    if len(current_lst) == ans_len:
```

```
        print(current_lst)
```

```
    else:
```

```
        for num in lst:
```

```
            if num in current_lst:
```

```
                pass
```

```
            else:
```

```
                current_lst.append(num)
```

```
                permutation_helper(lst, current_lst, ans_len)
```

```
                current_lst.pop()
```

lst = [1, 2, 3]

current_lst = [1, 2]

ans_len = 3

num = 2

```
def main():
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
    if len(current_lst) == ans_len:
```

```
        print(current_lst)
```

```
    else:
```

```
        for num in lst:
```

```
            if num in current_lst:
```

```
                pass
```

```
            else:
```

```
                current_lst.append(num)
```

```
                permutation_helper(lst, current_lst, ans_len)
```

```
                current_lst.pop()
```

lst = [1, 2, 3]

current_lst = [1, 2]

ans_len = 3

num = 2


```
def main():
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
    if len(current_lst) == ans_len:
```

```
        print(current_lst)
```

```
    else:
```

```
        for num in lst:
```

```
            if num in current_lst:
```

```
                pass
```

```
            else:
```

```
                current_lst.append(num)
```

```
                permutation_helper(lst, current_lst, ans_len)
```

```
                current_lst.pop()
```

lst = [1, 2, 3]

current_lst = [1, 2]

ans_len = 3

num = 2


```
def main():
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
    if len(current_lst) == ans_len:
```

```
        print(current_lst)
```

```
    else:
```

```
        for num in lst:
```

```
            if num in current_lst:
```

```
                pass
```

```
            else:
```

```
                current_lst.append(num)
```

```
                permutation_helper(lst, current_lst, ans_len)
```

```
                current_lst.pop()
```

lst = [1, 2, 3]

current_lst = [1, 2]

ans_len = 3

num = 3

```
def main():
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
    if len(current_lst) == ans_len:
```

```
        print(current_lst)
```

```
    else:
```

```
        for num in lst:
```

```
            if num in current_lst:
```

```
                pass
```

```
            else:
```

```
                current_lst.append(num)
```

```
                permutation_helper(lst, current_lst, ans_len)
```

```
                current_lst.pop()
```

lst = [1, 2, 3]

current_lst = [1, 2]

ans_len = 3

num = 3

```
def main():
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
    if len(current_lst) == ans_len:
```

```
        print(current_lst)
```

```
    else:
```

```
        for num in lst:
```

```
            if num in current_lst:
```

```
                pass
```

```
            else:
```

```
                current_lst.append(num)
```

```
                permutation_helper(lst, current_lst, ans_len)
```

```
                current_lst.pop()
```

lst = [1, 2, 3]

current_lst = [1, 2]

ans_len = 3

num = 3

```
def main():
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
    if len(current_lst) == ans_len:
```

```
        print(current_lst)
```

```
    else:
```

```
        for num in lst:
```

```
            if num in current_lst:
```

```
                pass
```

```
            else:
```

```
                current_lst.append(num)
```

```
                permutation_helper(lst, current_lst, ans_len)
```

```
                current_lst.pop()
```

lst = [1, 2, 3]

current_lst = [1, 2, 3]

ans_len = 3

num = 3

```
def main():
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
    if len(current_lst) == ans_len:
```

```
        print(current_lst)
```

```
    else:
```

```
        for num in lst:
```

```
            if num in current_lst:
```

```
                pass
```

```
            else:
```

```
                current_lst.append(num)
```

```
                permutation_helper(lst, current_lst, ans_len)
```

```
                current_lst.pop()
```

lst = [1, 2, 3]

current_lst = [1, 2, 3]

ans_len = 3

num = 3

```
def main():
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
    if len(current_lst) == ans_len:
```

```
        print(current_lst)
```

```
    else:
```

```
        for num in lst:
```

```
            if num in current_lst:
```

```
                pass
```

```
            else:
```

```
                current_lst.append(num)
```

```
                permutation_helper(lst, current_lst, ans_len)
```

```
                current_lst.pop()
```

lst = [1, 2, 3]

current_lst = [1, 2, 3]

ans_len = 3

```
def main():
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
    if len(current_lst) == ans_len:
```

```
        print(current_lst)
```

```
    else:
```

```
        for num in lst:
```

```
            if num in current_lst:
```

```
                pass
```

```
            else:
```

```
                current_lst.append(num)
```

```
                permutation_helper(lst, current_lst, ans_len)
```

```
                current_lst.pop()
```

lst = [1, 2, 3]

current_lst = [1, 2, 3]

ans_len = 3


```
def main():
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
    if len(current_lst) == ans_len:
```

```
        print(current_lst)
```

```
    else:
```

```
        for num in lst:
```

```
            if num in current_lst:
```

```
                pass
```

```
            else:
```

```
                current_lst.append(num)
```

```
                permutation_helper(lst, current_lst, ans_len)
```

```
                current_lst.pop()
```

lst = [1, 2, 3]

current_lst = [1, 2, 3]

ans_len = 3

```
permutation x
/usr/local/bin/python
[1, 2, 3]
```



```
def main():
```

```
def permutation(lst):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
    if len(current_lst) == ans_len:
```

```
        print(current_lst)
```

```
    else:
```

```
        for num in lst:
```

```
            if num in current_lst:
```

```
                pass
```

```
            else:
```

```
                current_lst.append(num)
```

```
                permutation_helper(lst, current_lst, ans_len)
```

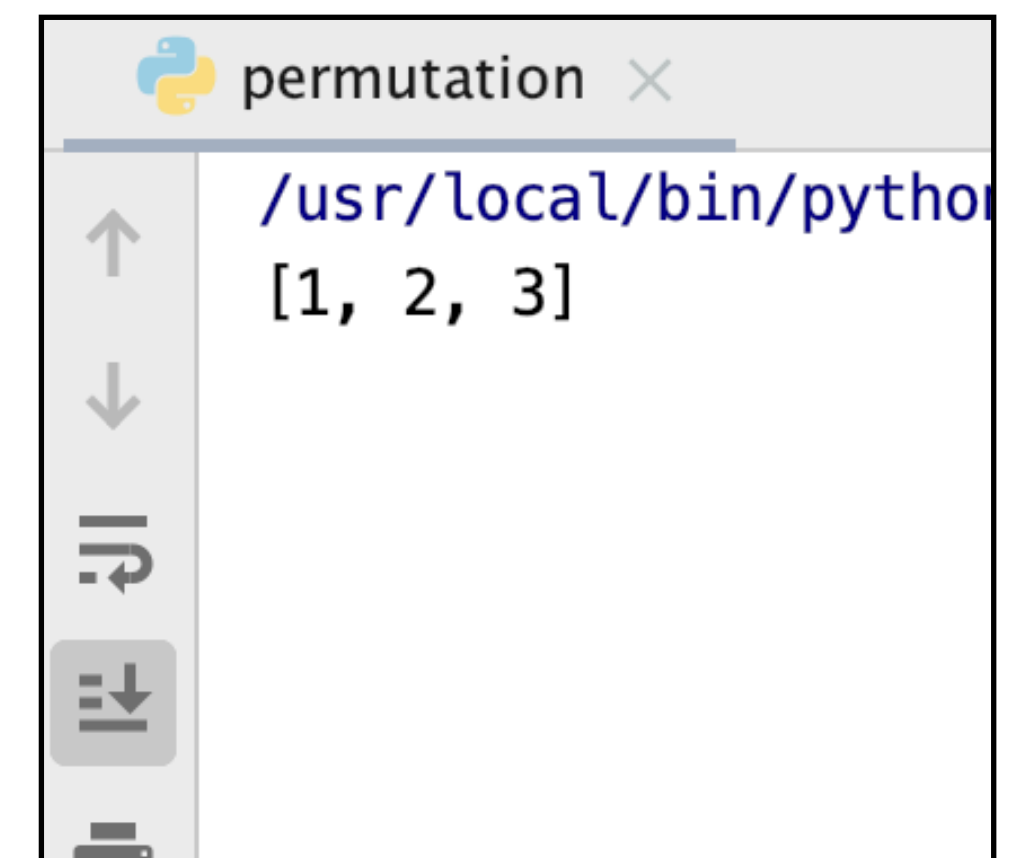
```
                current_lst.pop()
```

lst = [1, 2, 3]

current_lst = [1, 2, 3]

ans_len = 3

num = 3



A terminal window titled "permutation" with a close button. It shows the command `/usr/local/bin/python` and the output `[1, 2, 3]`. The terminal has a vertical toolbar on the left with icons for back, forward, search, and other navigation functions.

```
def main():
```

```
def permutation(lst):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
    if len(current_lst) == ans_len:
```

```
        print(current_lst)
```

```
    else:
```

```
        for num in lst:
```

```
            if num in current_lst:
```

```
                pass
```

```
            else:
```

```
                current_lst.append(num)
```

```
                permutation_helper(lst, current_lst, ans_len)
```

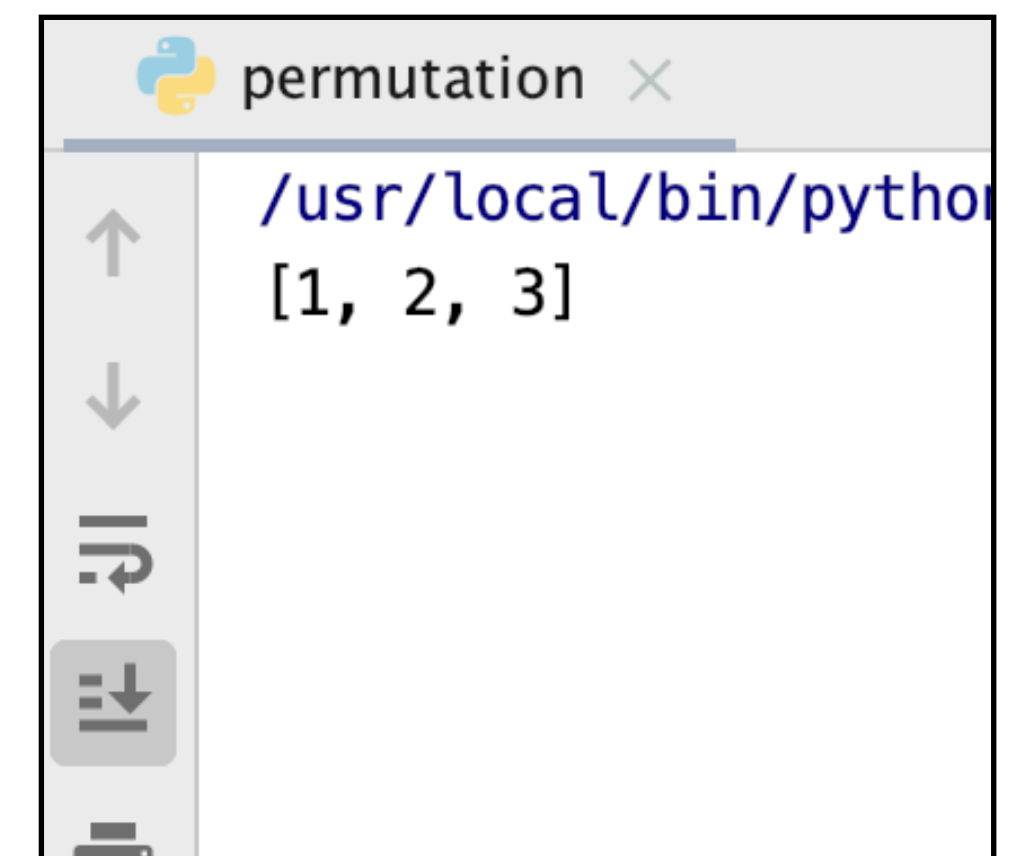
```
                current_lst.pop()
```

lst = [1, 2, 3]

current_lst = [1, 2, 3]

ans_len = 3

num = 3



A terminal window titled "permutation" with a close button. It shows the command `/usr/local/bin/python` and the output `[1, 2, 3]`. The terminal has a vertical toolbar on the left with icons for back, forward, search, and other navigation functions.

```
def main():
```

```
def permutation(lst):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
    if len(current_lst) == ans_len:
```

```
        print(current_lst)
```

```
    else:
```

```
        for num in lst:
```

```
            if num in current_lst:
```

```
                pass
```

```
            else:
```

```
                current_lst.append(num)
```

```
                permutation_helper(lst, current_lst, ans_len)
```

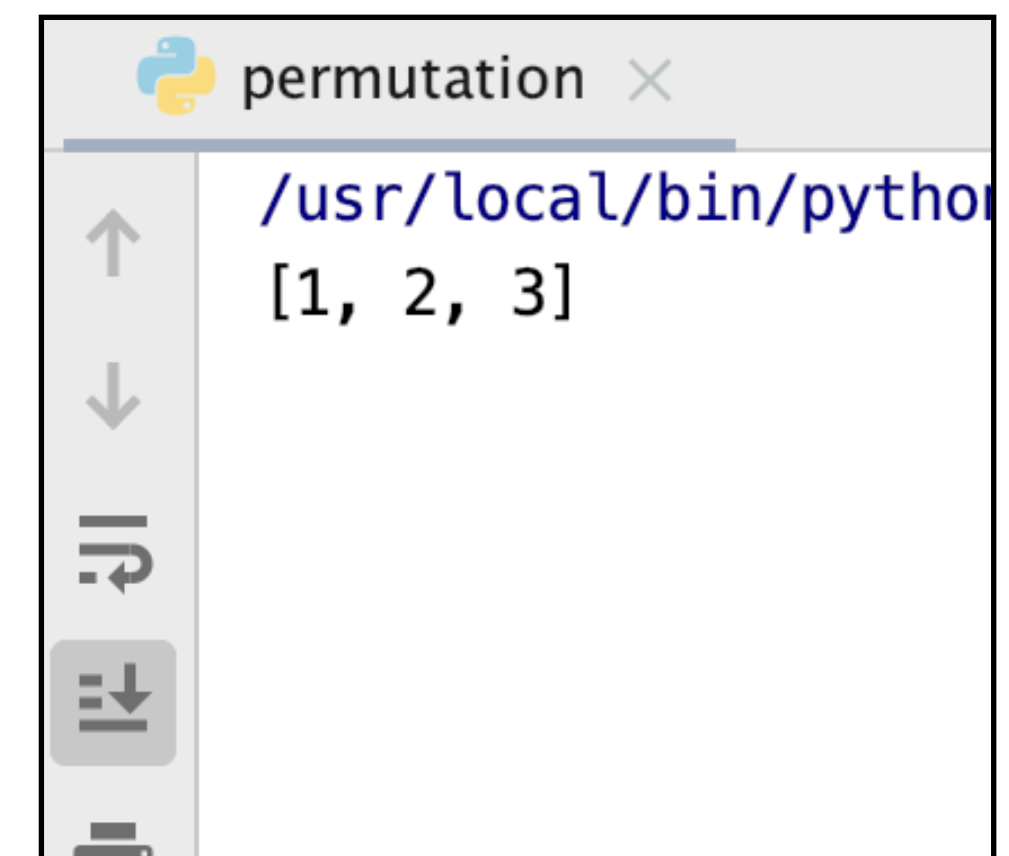
```
                current_lst.pop()
```

lst = [1, 2, 3]

current_lst = [1, 2]

ans_len = 3

num = 3



A terminal window titled "permutation" with a close button. It shows the command `/usr/local/bin/python` and the output `[1, 2, 3]`. The terminal has a vertical toolbar on the left with icons for back, forward, search, and other navigation functions.

```
def main():
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
    if len(current_lst) == ans_len:
```

```
        print(current_lst)
```

```
    else:
```

```
        for num in lst:
```

num = 2

```
            if num in current_lst:
```

```
                pass
```

```
            else:
```

```
                current_lst.append(num)
```

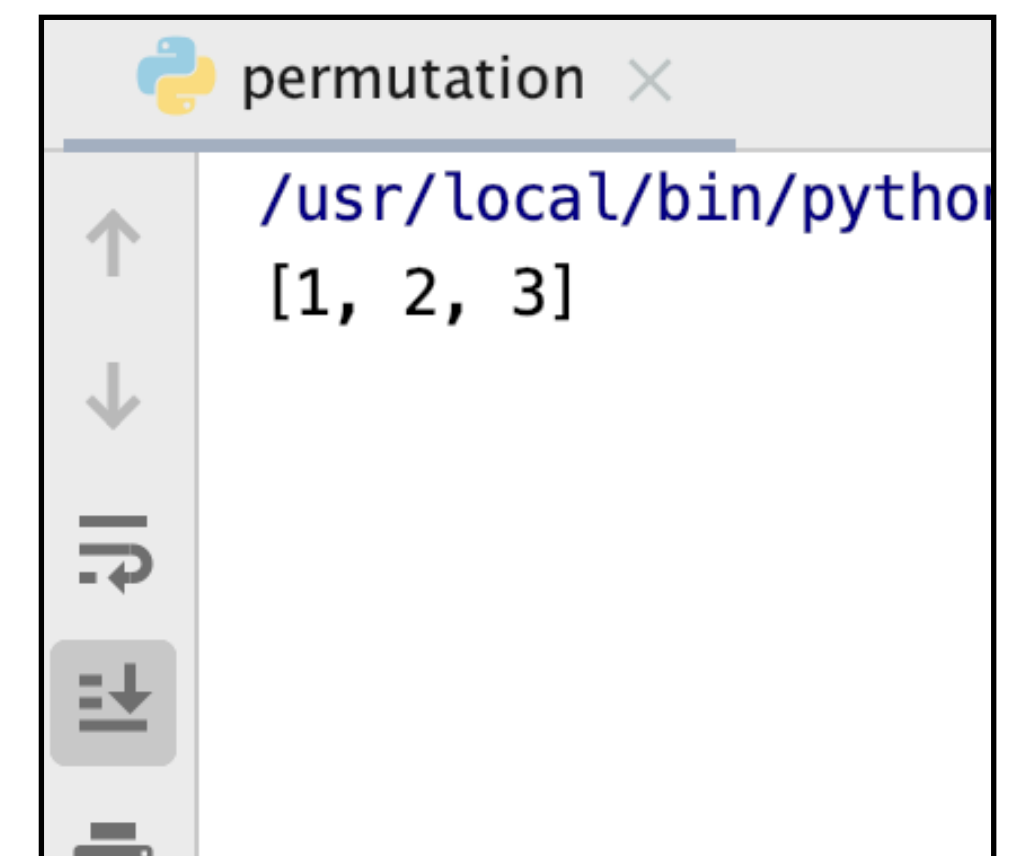
```
                permutation_helper(lst, current_lst, ans_len)
```

```
                current_lst.pop()
```

lst = [1, 2, 3]

current_lst = [1, 2]

ans_len = 3



```
permutation x
/usr/local/bin/python
[1, 2, 3]
```

```
def main():
```

```
def permutation(lst):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
    if len(current_lst) == ans_len:
```

```
        print(current_lst)
```

```
    else:
```

```
        for num in lst:
```

num = 2

```
            if num in current_lst:
```

```
                pass
```

```
            else:
```

```
                current_lst.append(num)
```

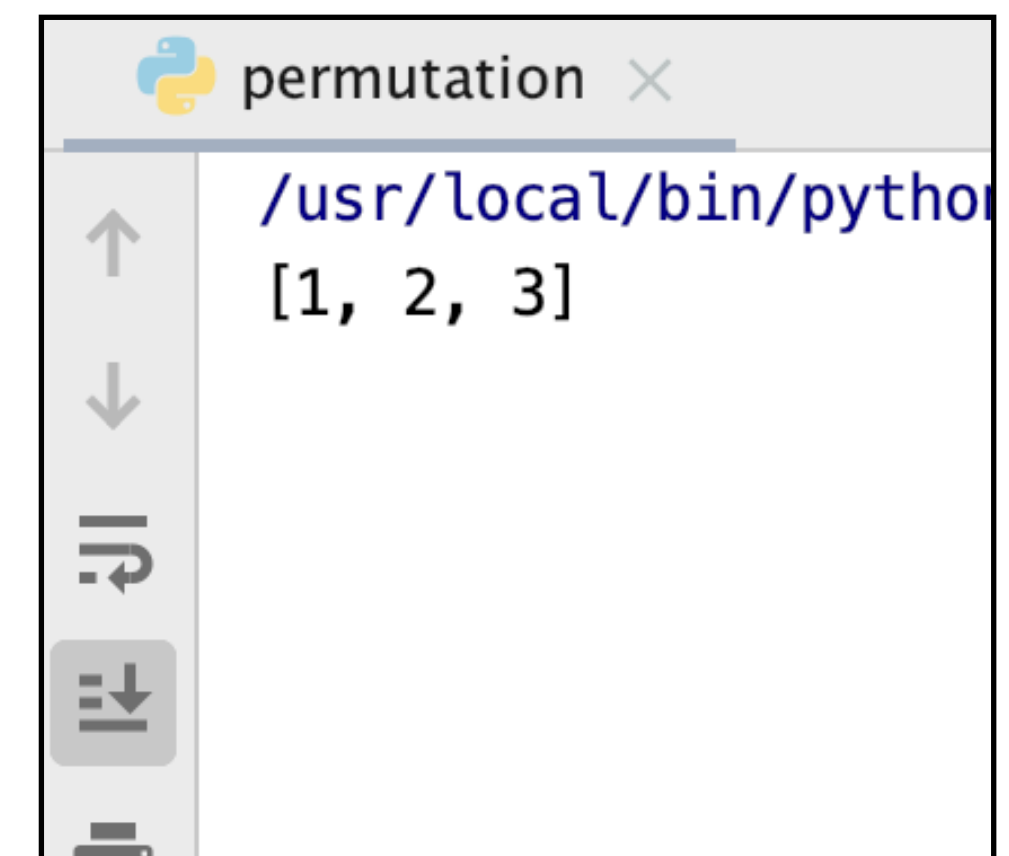
```
                permutation_helper(lst, current_lst, ans_len)
```

```
                current_lst.pop()
```

lst = [1, 2, 3]

current_lst = [1, 2]

ans_len = 3



A terminal window titled "permutation" with a close button. It shows the command `/usr/local/bin/python` and the output `[1, 2, 3]`. The terminal has a vertical toolbar on the left with icons for back, forward, search, and other navigation functions.

```
def main():
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
    if len(current_lst) == ans_len:
```

```
        print(current_lst)
```

```
    else:
```

```
        for num in lst:
```

num = 2

```
            if num in current_lst:
```

```
                pass
```

```
            else:
```

```
                current_lst.append(num)
```

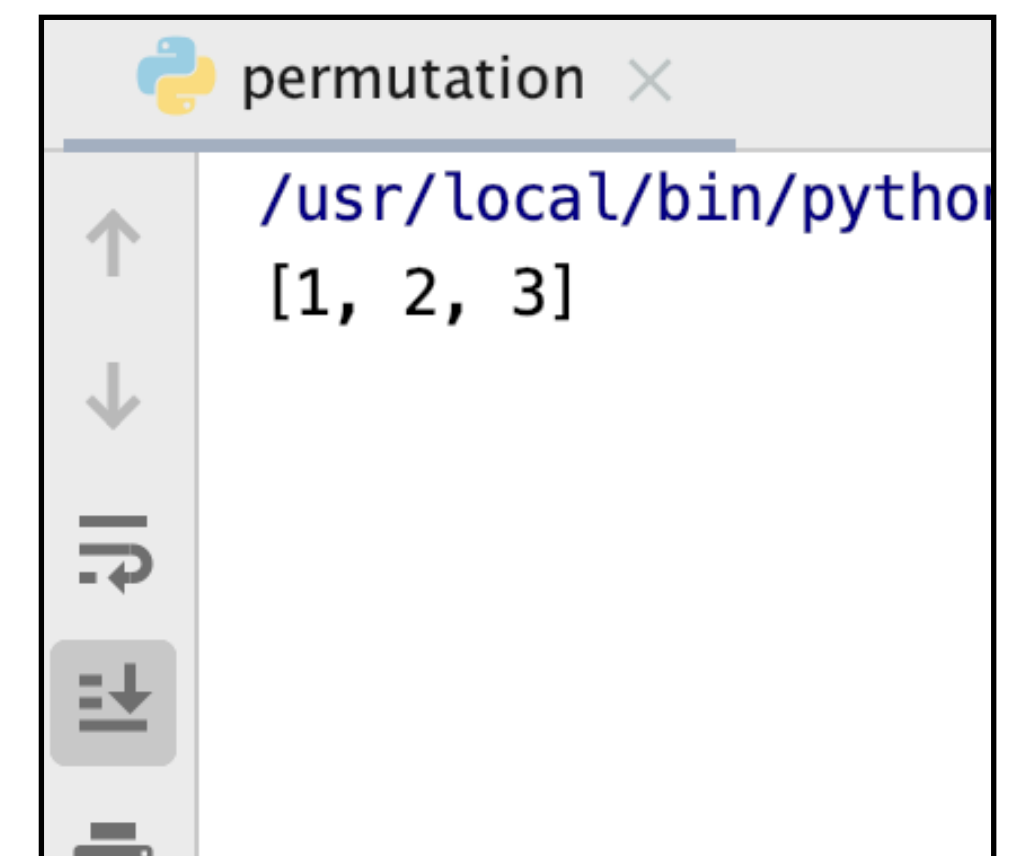
```
                permutation_helper(lst, current_lst, ans_len)
```

```
                current_lst.pop()
```

lst = [1, 2, 3]

current_lst = [1]

ans_len = 3



A terminal window titled "permutation" with a close button. The prompt is `/usr/local/bin/python`. The output is `[1, 2, 3]`. The terminal has a vertical toolbar on the left with icons for back, forward, search, and other navigation functions.

```
permutation ×  
/usr/local/bin/python  
[1, 2, 3]
```



```
def main():
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
    if len(current_lst) == ans_len:
```

```
        print(current_lst)
```

```
    else:
```

```
        for num in lst:
```

num = 3

```
            if num in current_lst:
```

```
                pass
```

```
            else:
```

```
                current_lst.append(num)
```

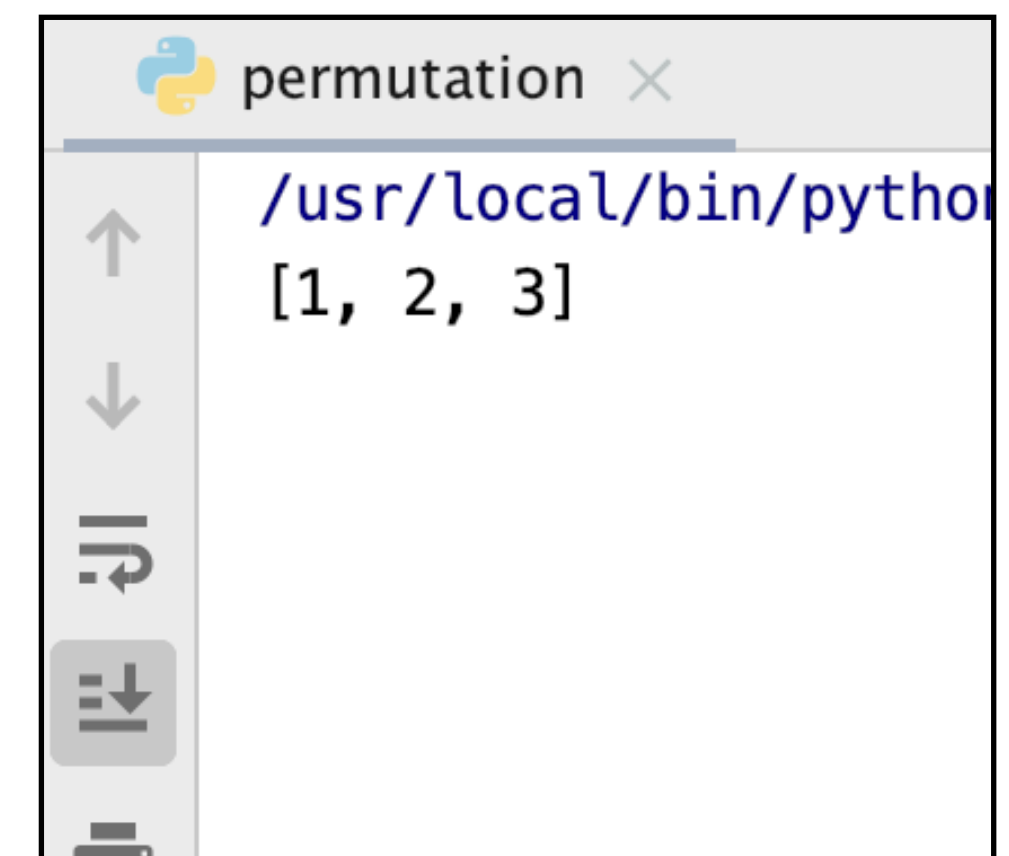
```
                permutation_helper(lst, current_lst, ans_len)
```

```
                current_lst.pop()
```

lst = [1, 2, 3]

current_lst = [1]

ans_len = 3



A terminal window titled "permutation" with a close button. It shows the command `/usr/local/bin/python` and the output `[1, 2, 3]`. The terminal has a vertical toolbar on the left with icons for back, forward, search, and other navigation functions.

```
permutation ×  
/usr/local/bin/python  
[1, 2, 3]
```

```
def main():
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
    if len(current_lst) == ans_len:
```

```
        print(current_lst)
```

```
    else:
```

```
        for num in lst:
```

```
            if num in current_lst:
```

```
                pass
```

```
            else:
```

```
                current_lst.append(num)
```

```
                permutation_helper(lst, current_lst, ans_len)
```

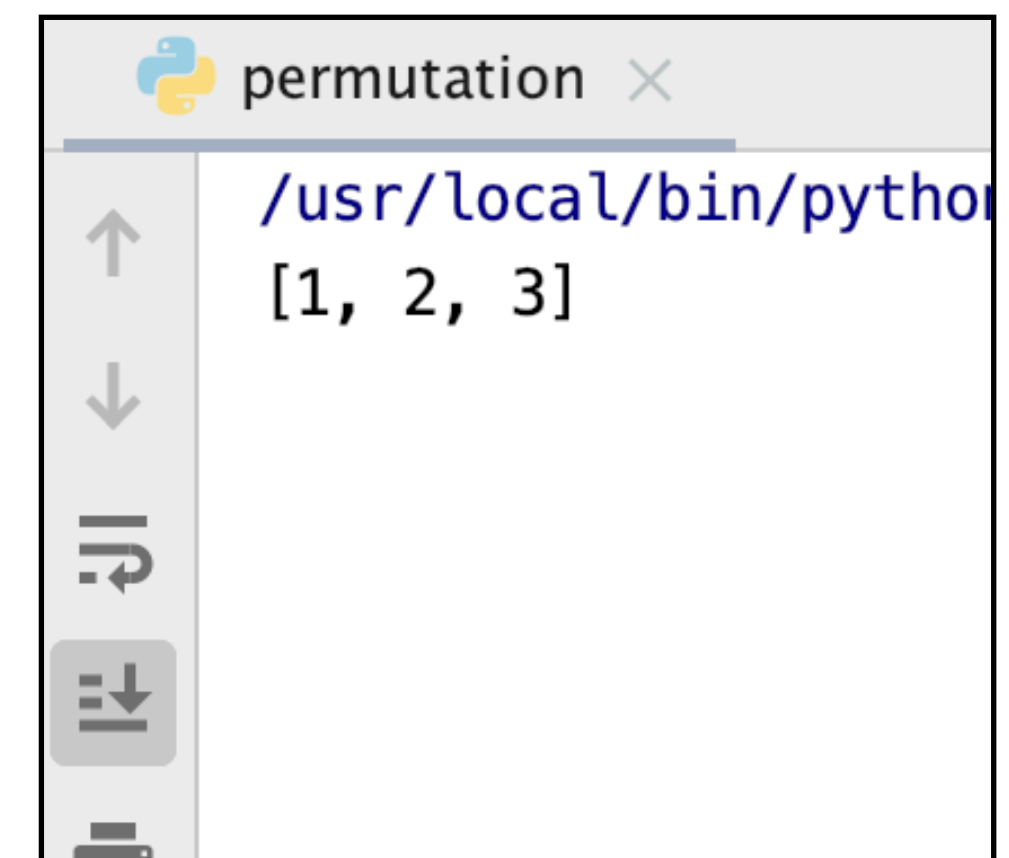
```
                current_lst.pop()
```

num = 3

lst = [1, 2, 3]

current_lst = [1]

ans_len = 3



A terminal window titled "permutation" with a close button. It shows the command `/usr/local/bin/python` and the output `[1, 2, 3]`. The terminal has a vertical toolbar on the left with icons for back, forward, search, and other navigation functions.


```
def main():
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
    if len(current_lst) == ans_len:
```

```
        print(current_lst)
```

```
    else:
```

```
        for num in lst:
```

```
            if num in current_lst:
```

```
                pass
```

```
            else:
```

```
                current_lst.append(num)
```

```
                permutation_helper(lst, current_lst, ans_len)
```

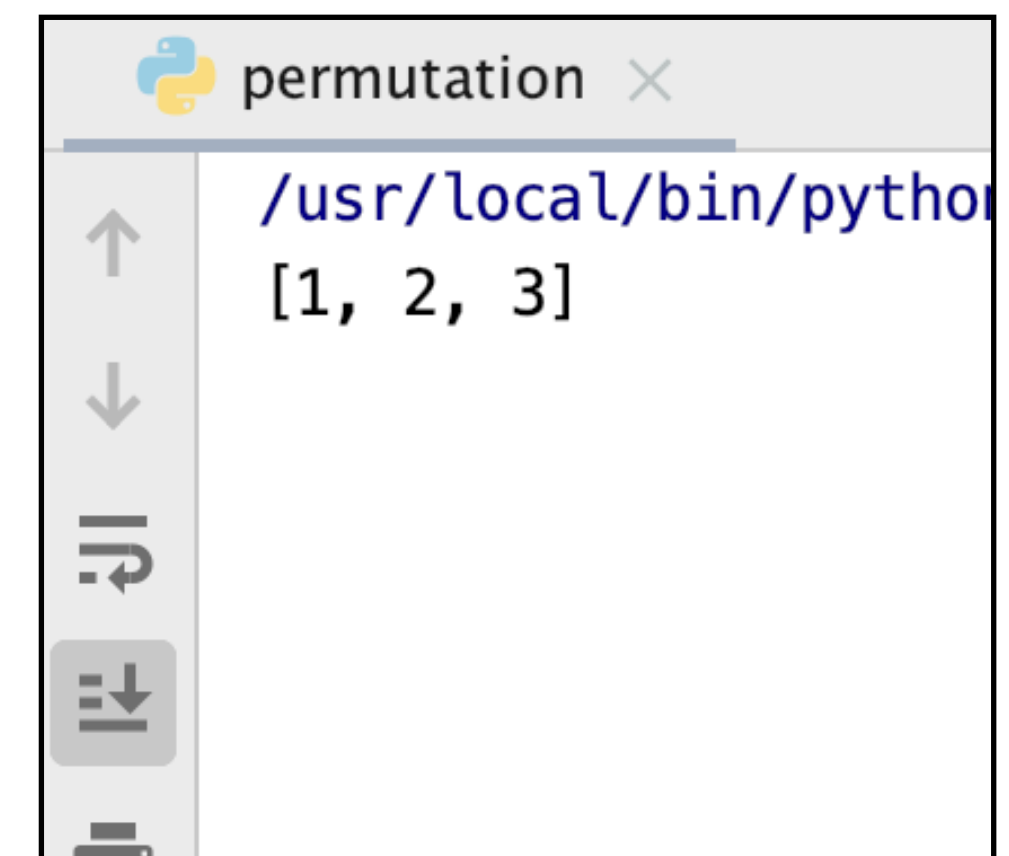
```
                current_lst.pop()
```

num = 3

lst = [1, 2, 3]

current_lst = [1, 3]

ans_len = 3



A terminal window titled "permutation" with a close button. It shows the command `/usr/local/bin/python` and the output `[1, 2, 3]`. The terminal has a vertical toolbar on the left with icons for back, forward, search, and other navigation functions.

```
def main():
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
    if len(current_lst) == ans_len:
```

```
        print(current_lst)
```

```
    else:
```

```
        for num in lst:
```

num = 3

```
            if num in current_lst:
```

```
                pass
```

```
            else:
```

```
                current_lst.append(num)
```

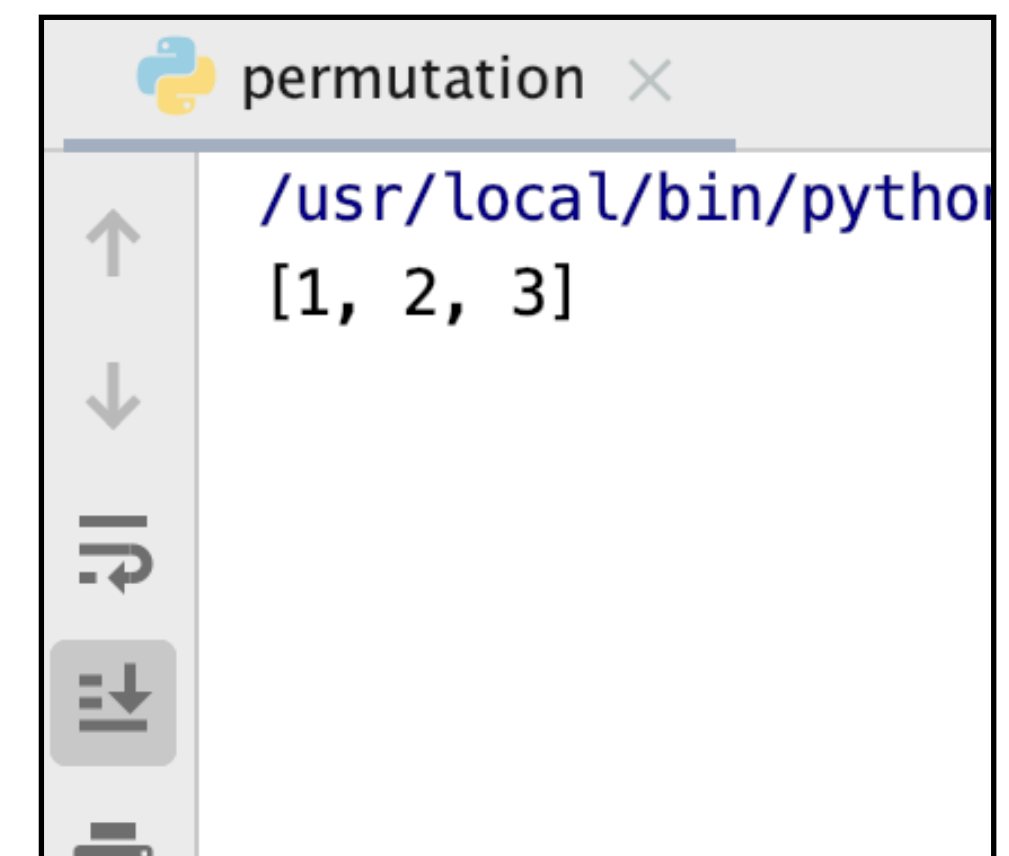
```
                permutation_helper(lst, current_lst, ans_len)
```

```
                current_lst.pop()
```

lst = [1, 2, 3]

current_lst = [1, 3]

ans_len = 3



```
permutation x
/usr/local/bin/python
[1, 2, 3]
```

```
def main():
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
    if len(current_lst) == ans_len:
```

```
        print(current_lst)
```

```
    else:
```

```
        for num in lst:
```

```
            if num in current_lst:
```

```
                pass
```

```
            else:
```

```
                current_lst.append(num)
```

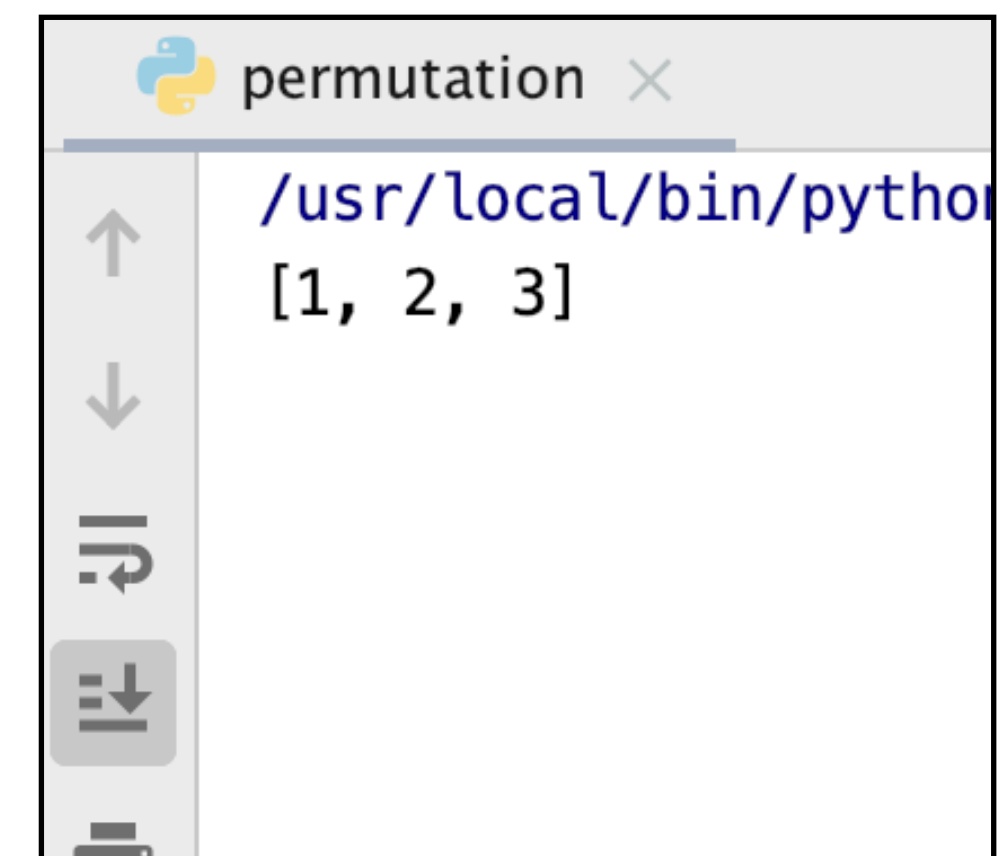
```
                permutation_helper(lst, current_lst, ans_len)
```

```
                current_lst.pop()
```

lst = [1, 2, 3]

current_lst = [1, 3]

ans_len = 3



A terminal window titled 'permutation' with a close button. The prompt is `/usr/local/bin/python`. The input is `[1, 2, 3]`. The terminal has a vertical toolbar on the left with icons for back, forward, search, and other navigation functions.

```
def main():
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
    if len(current_lst) == ans_len:
```

```
        print(current_lst)
```

```
    else:
```

```
        for num in lst:
```

```
            if num in current_lst:
```

```
                pass
```

```
            else:
```

```
                current_lst.append(num)
```

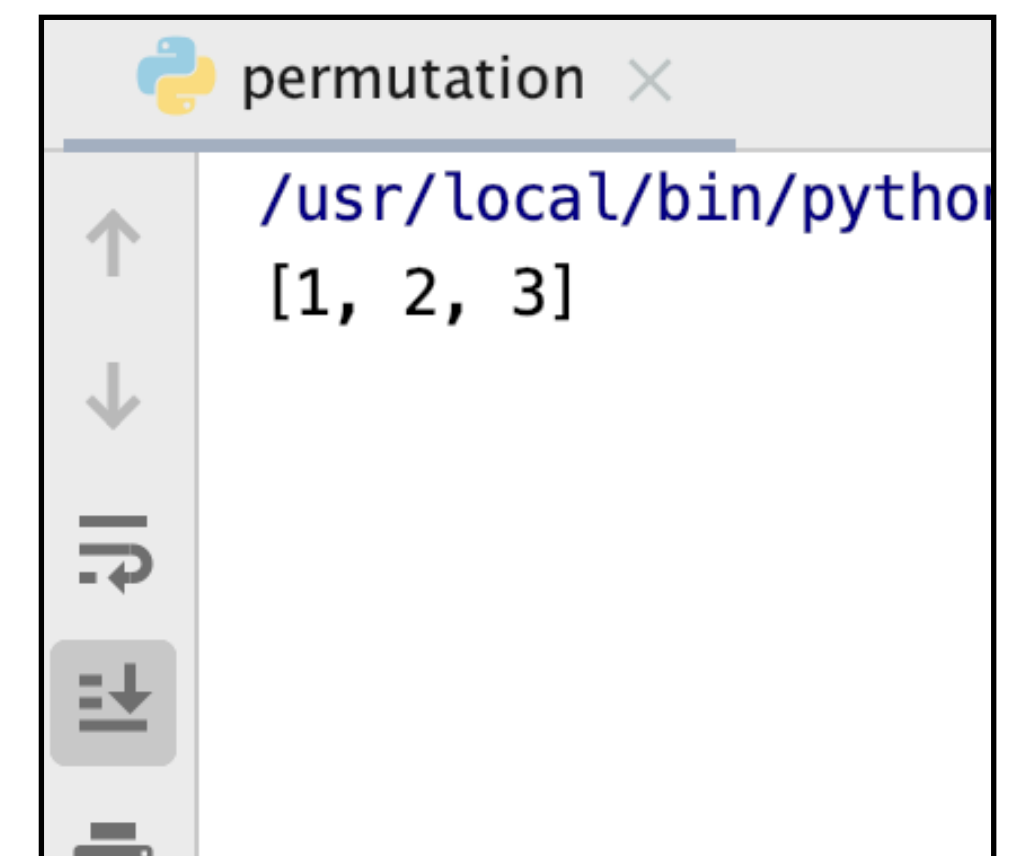
```
                permutation_helper(lst, current_lst, ans_len)
```

```
                current_lst.pop()
```

lst = [1, 2, 3]

current_lst = [1, 3]

ans_len = 3



A terminal window titled 'permutation' with a close button. The prompt is `/usr/local/bin/python`. The output is `[1, 2, 3]`. The terminal has a vertical toolbar on the left with icons for back, forward, search, and other navigation functions.

```
permutation x
/usr/local/bin/python
[1, 2, 3]
```

```
def main():
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
    if len(current_lst) == ans_len:
```

```
        print(current_lst)
```

```
    else:
```

```
        for num in lst:
```

```
            if num in current_lst:
```

```
                pass
```

```
            else:
```

```
                current_lst.append(num)
```

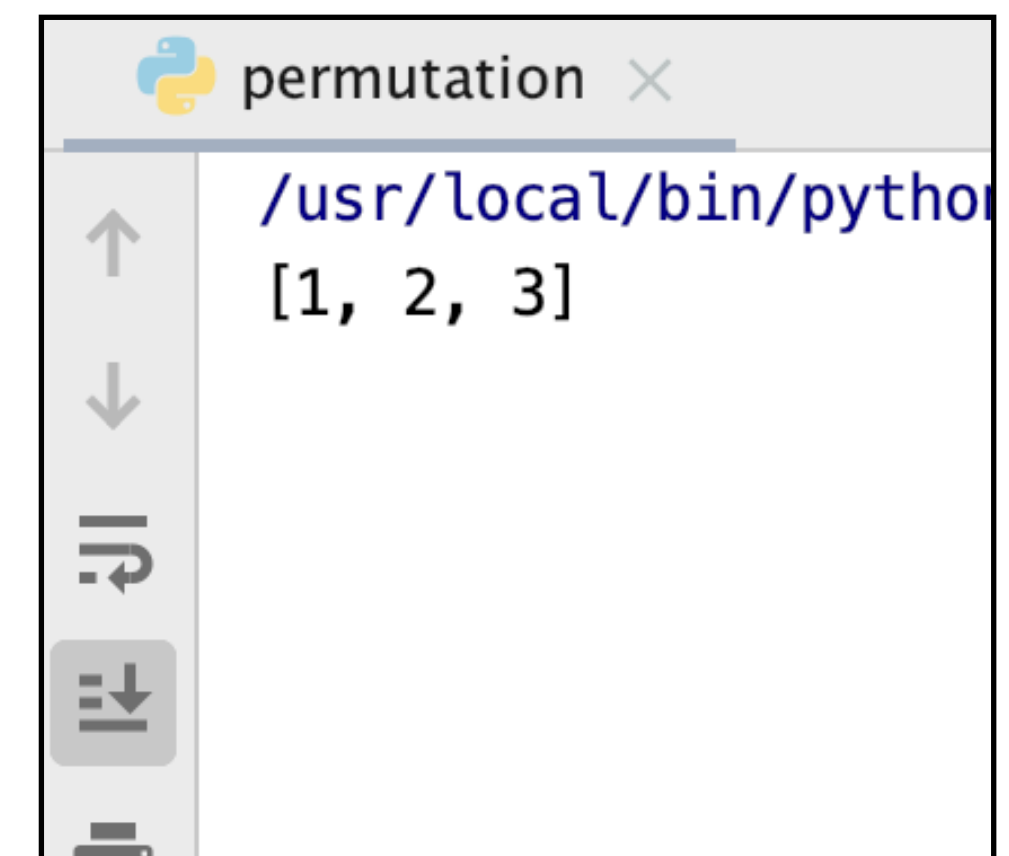
```
                permutation_helper(lst, current_lst, ans_len)
```

```
                current_lst.pop()
```

lst = [1, 2, 3]

current_lst = [1, 3]

ans_len = 3



```
permutation x
/usr/local/bin/python
[1, 2, 3]
```

```
def main():
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
    if len(current_lst) == ans_len:
```

```
        print(current_lst)
```

```
    else:
```

```
        for num in lst:
```

```
            if num in current_lst:
```

```
                pass
```

```
            else:
```

```
                current_lst.append(num)
```

```
                permutation_helper(lst, current_lst, ans_len)
```

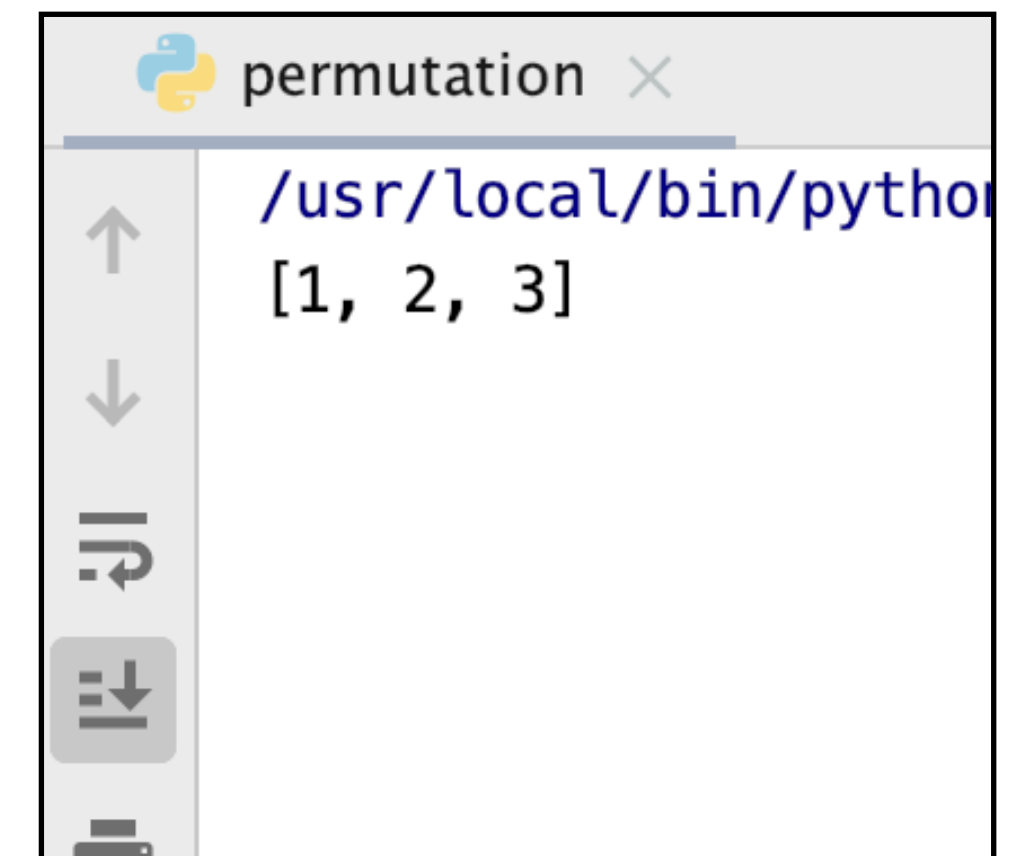
```
                current_lst.pop()
```

num = 1

lst = [1, 2, 3]

current_lst = [1, 3]

ans_len = 3



A terminal window titled "permutation" with a close button. The prompt is `/usr/local/bin/python`. The input is `[1, 2, 3]`. The window has a vertical toolbar on the left with icons for back, forward, search, and other navigation functions.


```
def main():
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
    if len(current_lst) == ans_len:
```

```
        print(current_lst)
```

```
    else:
```

num = 1

```
        for num in lst:
```

```
            if num in current_lst:
```

```
                pass
```

```
            else:
```

```
                current_lst.append(num)
```

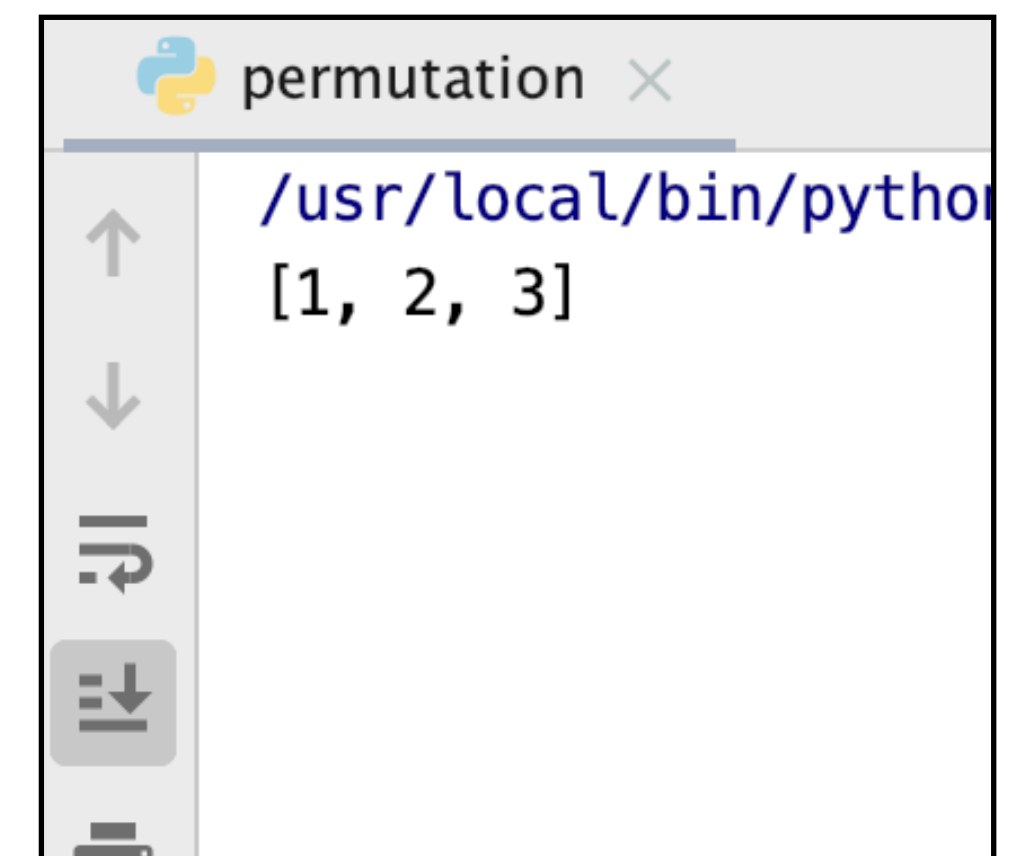
```
                permutation_helper(lst, current_lst, ans_len)
```

```
                current_lst.pop()
```

lst = [1, 2, 3]

current_lst = [1, 3]

ans_len = 3



A terminal window titled "permutation" with a close button. It shows the command `/usr/local/bin/python` and the output `[1, 2, 3]`. The terminal has a vertical toolbar on the left with icons for back, forward, search, and other navigation functions.

```
def main():
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
    if len(current_lst) == ans_len:
```

```
        print(current_lst)
```

```
    else:
```

num = 1

```
        for num in lst:
```

```
            if num in current_lst:
```

```
                pass
```

```
            else:
```

```
                current_lst.append(num)
```

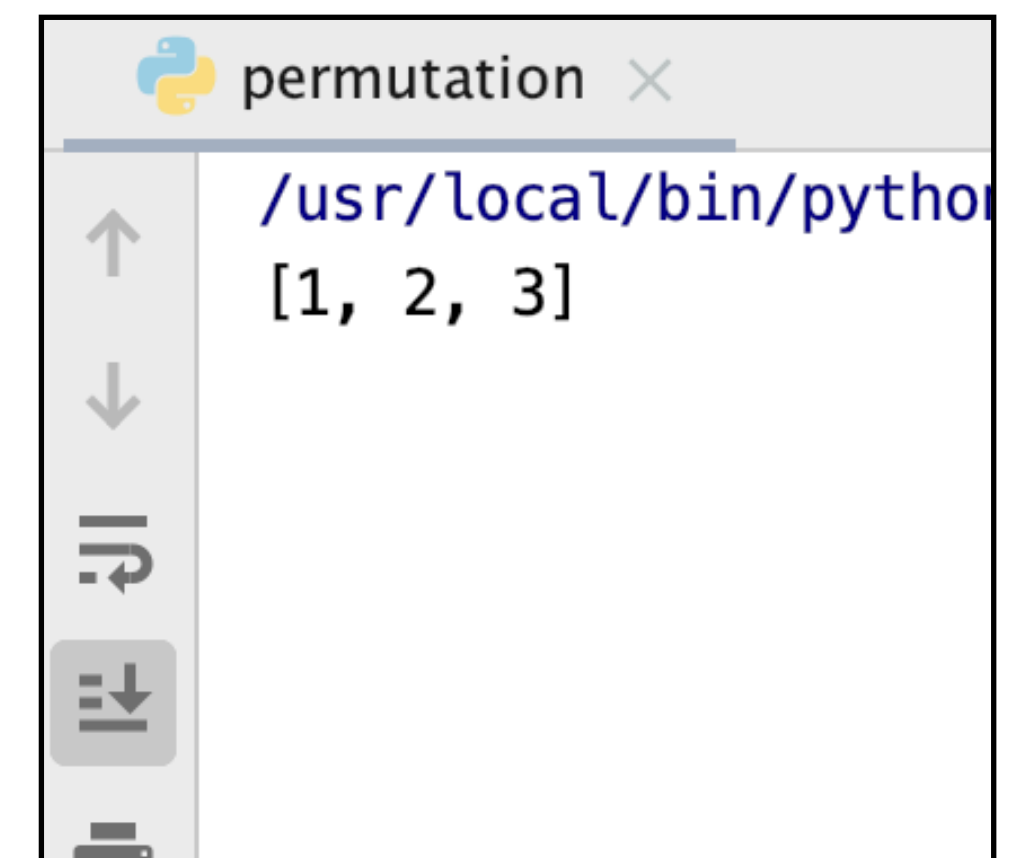
```
                permutation_helper(lst, current_lst, ans_len)
```

```
                current_lst.pop()
```

lst = [1, 2, 3]

current_lst = [1, 3]

ans_len = 3



A terminal window titled "permutation" with a close button. It shows the command `/usr/local/bin/python` being executed, followed by the output `[1, 2, 3]`. The terminal has a vertical toolbar on the left with icons for back, forward, search, and other navigation functions.


```
def main():
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
    if len(current_lst) == ans_len:
```

```
        print(current_lst)
```

```
    else:
```

```
        for num in lst:
```

```
            if num in current_lst:
```

```
                pass
```

```
            else:
```

```
                current_lst.append(num)
```

```
                permutation_helper(lst, current_lst, ans_len)
```

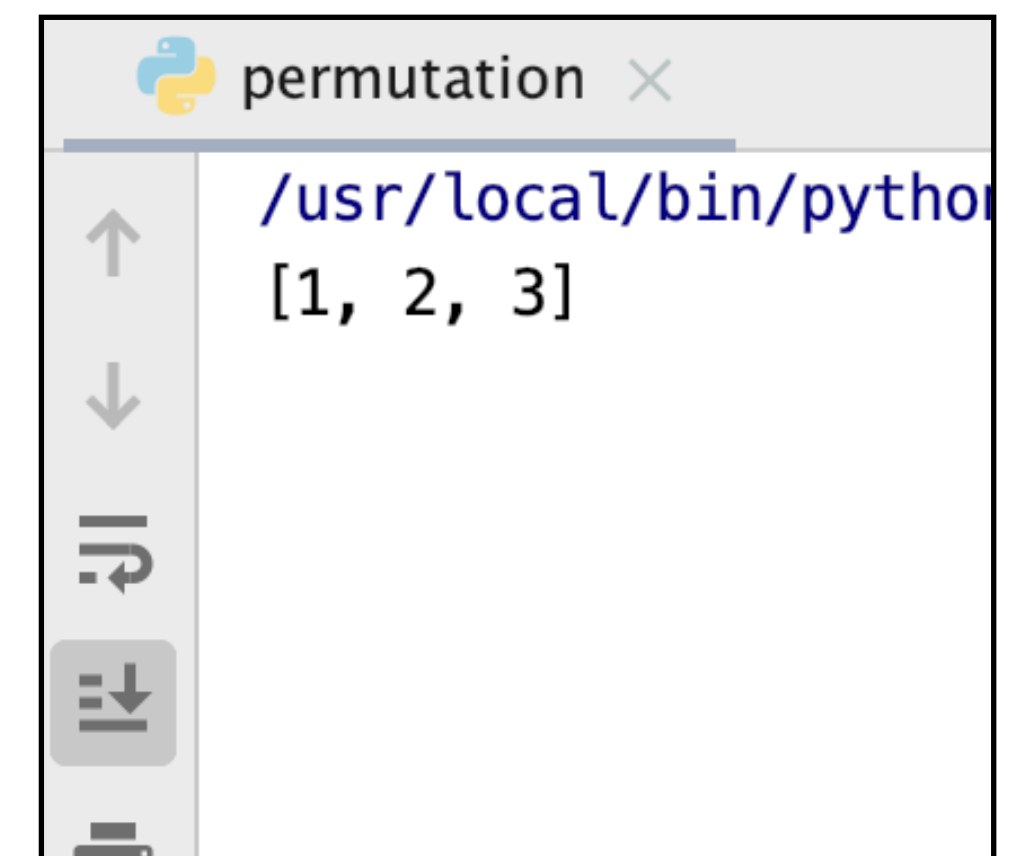
```
                current_lst.pop()
```

num = 2

lst = [1, 2, 3]

current_lst = [1, 3]

ans_len = 3



```
permutation x
/usr/local/bin/python
[1, 2, 3]
```

```
def main():
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
    if len(current_lst) == ans_len:
```

```
        print(current_lst)
```

```
    else:
```

num = 2

```
        for num in lst:
```

```
            if num in current_lst:
```

```
                pass
```

```
            else:
```

```
                current_lst.append(num)
```

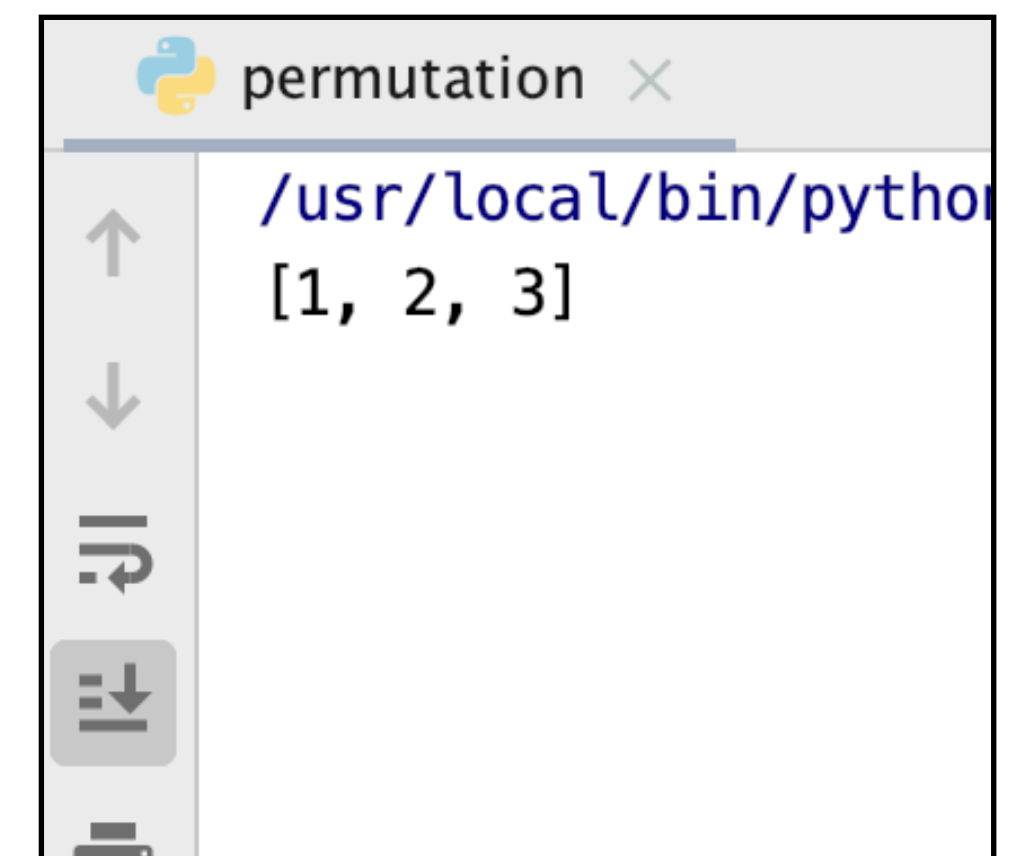
```
                permutation_helper(lst, current_lst, ans_len)
```

```
                current_lst.pop()
```

lst = [1, 2, 3]

current_lst = [1, 3]

ans_len = 3



A terminal window titled "permutation" with a close button. It shows the command `/usr/local/bin/python` and the output `[1, 2, 3]`. The terminal has a vertical toolbar on the left with icons for back, forward, search, and other navigation functions.

```
def main():
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
    if len(current_lst) == ans_len:
```

```
        print(current_lst)
```

```
    else:
```

num = 2

```
        for num in lst:
```

```
            if num in current_lst:
```

```
                pass
```

```
            else:
```

```
                current_lst.append(num)
```

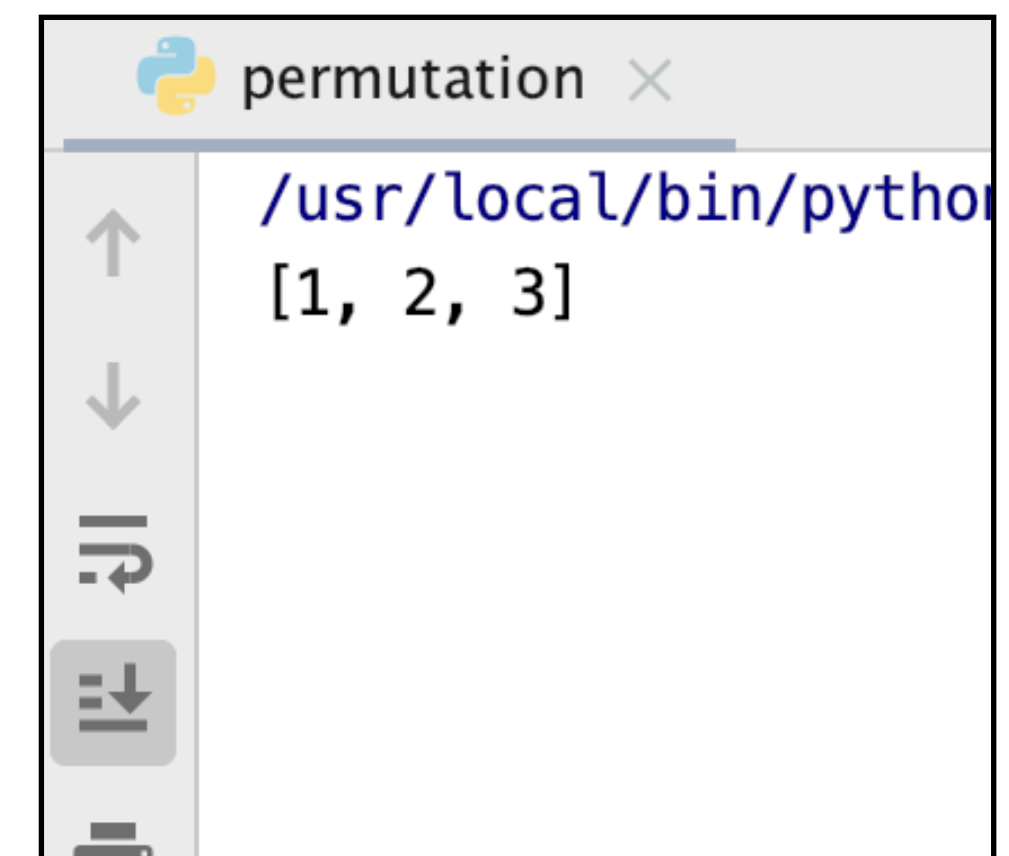
```
                permutation_helper(lst, current_lst, ans_len)
```

```
                current_lst.pop()
```

lst = [1, 2, 3]

current_lst = [1, 3, 2]

ans_len = 3



```
permutation x  
/usr/local/bin/python  
[1, 2, 3]
```

```
def main():
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
    if len(current_lst) == ans_len:
```

```
        print(current_lst)
```

```
    else:
```

num = 2

```
        for num in lst:
```

```
            if num in current_lst:
```

```
                pass
```

```
            else:
```

```
                current_lst.append(num)
```

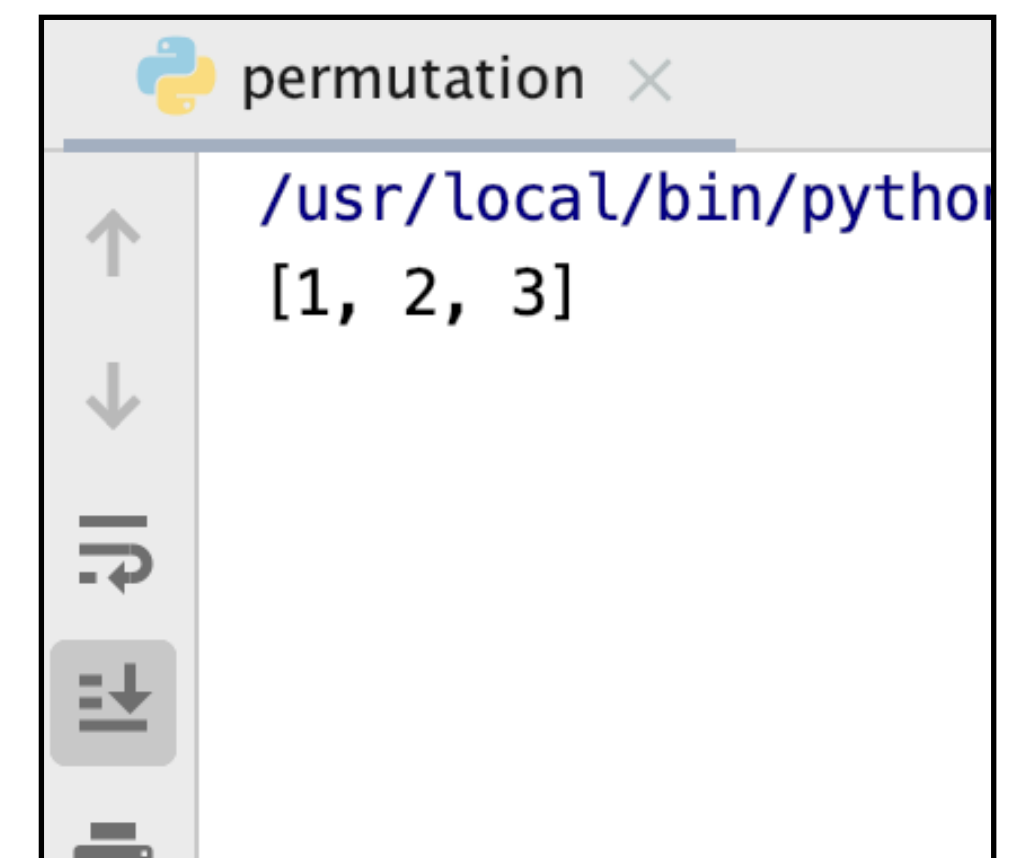
```
                permutation_helper(lst, current_lst, ans_len)
```

```
                current_lst.pop()
```

lst = [1, 2, 3]

current_lst = [1, 3, 2]

ans_len = 3



A terminal window titled "permutation" with a close button. It shows the command `/usr/local/bin/python` and the output `[1, 2, 3]`. The terminal has a vertical toolbar on the left with icons for back, forward, search, and other navigation functions.

```
def main():
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
    if len(current_lst) == ans_len:
```

```
        print(current_lst)
```

```
    else:
```

```
        for num in lst:
```

```
            if num in current_lst:
```

```
                pass
```

```
            else:
```

```
                current_lst.append(num)
```

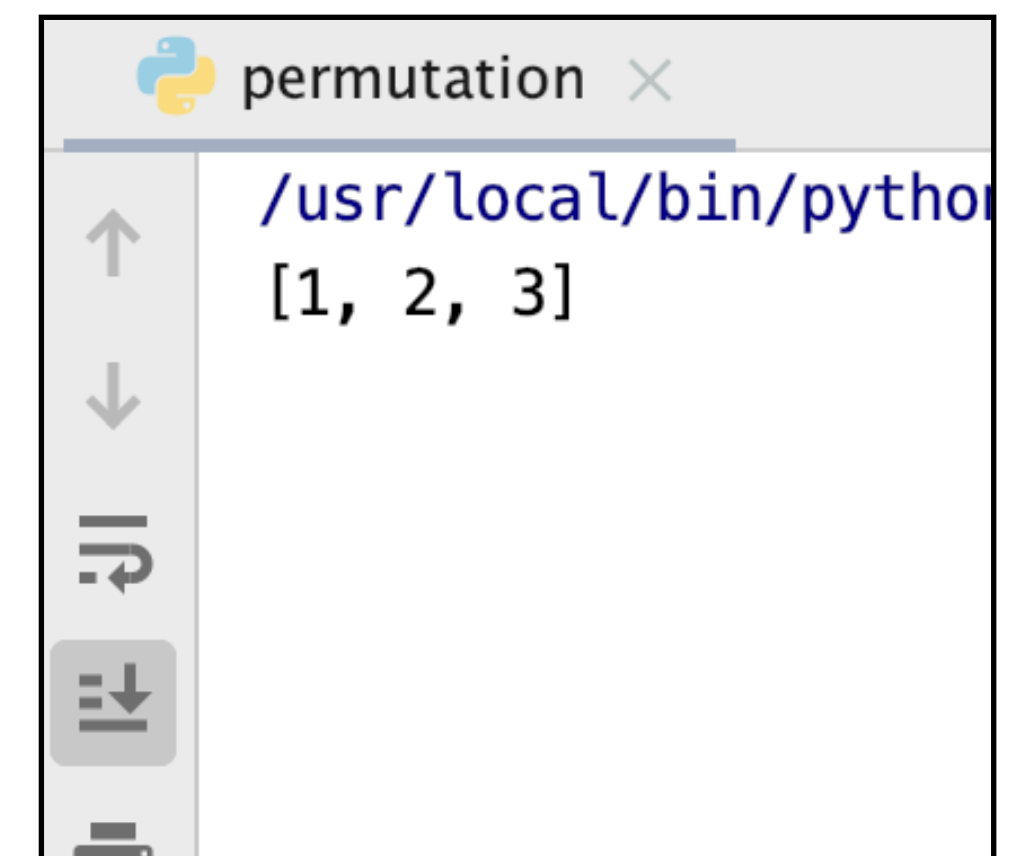
```
                permutation_helper(lst, current_lst, ans_len)
```

```
                current_lst.pop()
```

lst = [1, 2, 3]

current_lst = [1, 3, 2]

ans_len = 3



```
permutation x
/usr/local/bin/python
[1, 2, 3]
```

```

def main():
def permutation_helper(lst, current_lst, ans_len):
def permutation_helper(lst, current_lst, ans_len):
def permutation_helper(lst, current_lst, ans_len):
def permutation_helper(lst, current_lst, ans_len):
    if len(current_lst) == ans_len:
        print(current_lst)
    else:
        for num in lst:
            if num in current_lst:
                pass
            else:
                current_lst.append(num)
                permutation_helper(lst, current_lst, ans_len)
                current_lst.pop()

```

lst = [1, 2, 3]

current_lst = [1, 3, 2]

ans_len = 3

```

permutation x
/usr/local/bin/python
[1, 2, 3]

```



```
def main():
    def permutation_helper(lst, current, ans, len):
    def permutation_helper(lst, current, ans, len):
```

Age (years)	Percentage (%)
18	10
20	25
22	35
24	45
26	55
28	65
30	75
32	85
34	90
36	95
38	98
40	100
42	100
44	100
46	100
48	100
50	100
52	100
54	100
56	100
58	100
60	100
62	100
64	100
65	100

```
def permutation_helper(let current let ans len):
```

```
def permutation_helper(lst current lst ans len):
```

```
def permutation_helper(lst, current, lst_ans, len):
```

```
def permutation_helper(lst, current_lst, ans_len):
    if len(current_lst) == ans_len:
        print(current_lst)
    else:
        for num in lst:
            if num in current_lst:
                pass
            else:
                current_lst.append(num)
                permutation_helper(lst, current_lst, ans_len)
                current_lst.pop()
```

```
if len(current_lst) == ans_len:
```

```
print(current_lst)
```

```
else:
```

```
for num in lst:
```

```
if num in current_lst:
```

pass

```
else:
```

```
current_lst.append(num)
```

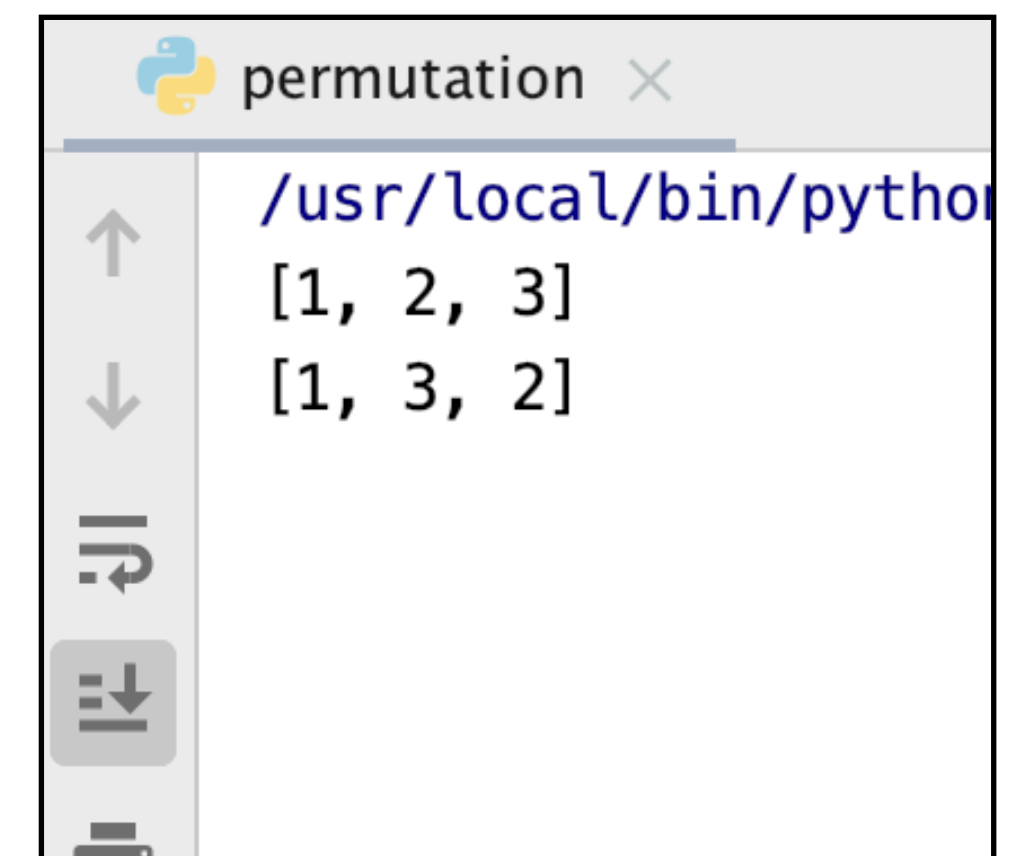
```
permutation_helper(lst, current_lst, ans_len)
```

```
current_lst.pop()
```

Ist = [1, 2, 3]

```
current_lst = [ 1, 3, 2 ]
```

ans_len = 3



```
def main():
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
    if len(current_lst) == ans_len:
```

```
        print(current_lst)
```

```
    else:
```

num = 2

```
        for num in lst:
```

```
            if num in current_lst:
```

```
                pass
```

```
            else:
```

```
                current_lst.append(num)
```

```
                permutation_helper(lst, current_lst, ans_len)
```

```
                current_lst.pop()
```

lst = [1, 2, 3]

current_lst = [1, 3, 2]

ans_len = 3

```
permutation x
/usr/local/bin/python
[1, 2, 3]
[1, 3, 2]
```



```
def main():
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
    if len(current_lst) == ans_len:
```

```
        print(current_lst)
```

```
    else:
```

num = 2

```
        for num in lst:
```

```
            if num in current_lst:
```

```
                pass
```

```
            else:
```

```
                current_lst.append(num)
```

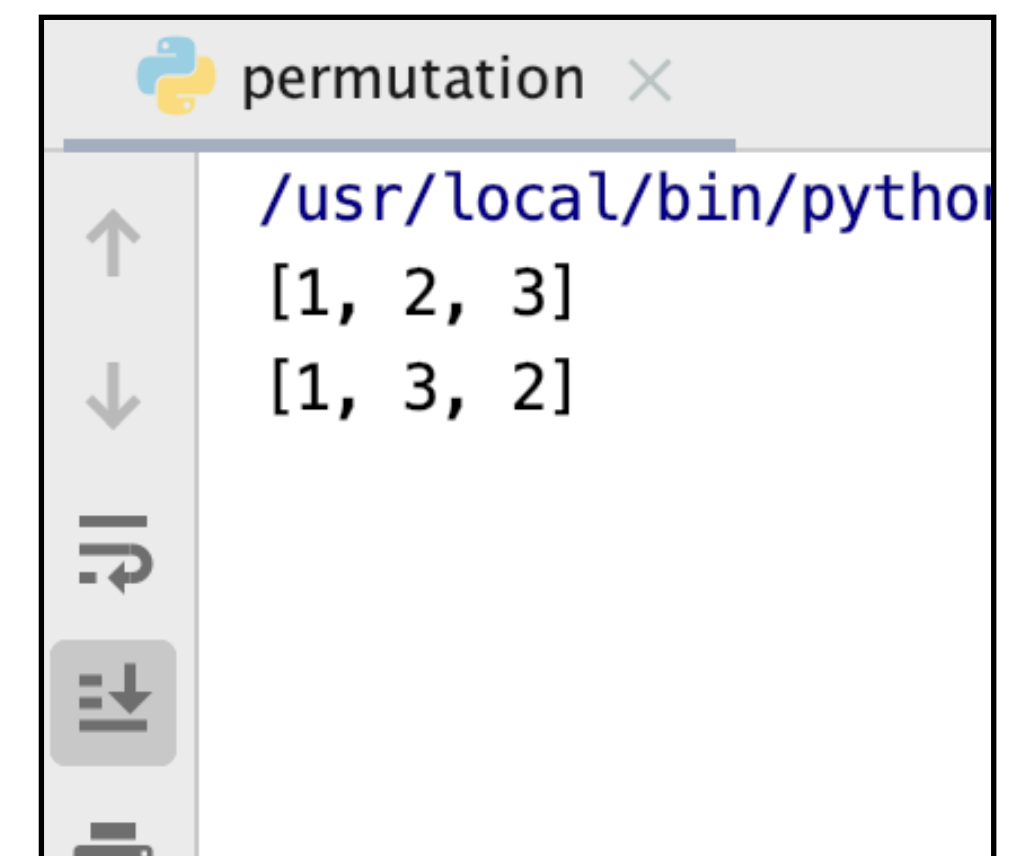
```
                permutation_helper(lst, current_lst, ans_len)
```

```
                current_lst.pop()
```

lst = [1, 2, 3]

current_lst = [1, 3]

ans_len = 3



A terminal window titled "permutation" with a close button. It shows the command `/usr/local/bin/python` and the output of the program, which lists the permutations [1, 2, 3] and [1, 3, 2]. The terminal has a vertical toolbar on the left with icons for back, forward, search, and other navigation functions.

```
permutation x
/usr/local/bin/python
[1, 2, 3]
[1, 3, 2]
```

```

def main():
    def permutation_helper(lst, current_lst, ans_len):
    def permutation_helper(lst, current_lst, ans_len):
def permutation_helper(lst, current_lst, ans_len):
def permutation_helper(lst, current_lst, ans_len):
    if len(current_lst) == ans_len:
        print(current_lst)
    else:
        for num in lst:
            if num in current_lst:
                pass
            else:
                current_lst.append(num)
                permutation_helper(lst, current_lst, ans_len)
                current_lst.pop()

```

num = 3

lst = [1, 2, 3]

current_lst = [1, 3]

ans_len = 3

```

permutation x
/usr/local/bin/python
[1, 2, 3]
[1, 3, 2]

```

```
def main():
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
    if len(current_lst) == ans_len:
```

```
        print(current_lst)
```

```
    else:
```

num = 3

```
        for num in lst:
```

```
            if num in current_lst:
```

```
                pass
```

```
            else:
```

```
                current_lst.append(num)
```

```
                permutation_helper(lst, current_lst, ans_len)
```

```
                current_lst.pop()
```

lst = [1, 2, 3]

current_lst = [1, 3]

ans_len = 3

```
permutation x
/usr/local/bin/python
[1, 2, 3]
[1, 3, 2]
```

```
def main():
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
    if len(current_lst) == ans_len:
```

```
        print(current_lst)
```

```
    else:
```

num = 3

```
        for num in lst:
```

```
            if num in current_lst:
```

```
                pass
```

```
            else:
```

```
                current_lst.append(num)
```

```
                permutation_helper(lst, current_lst, ans_len)
```

```
                current_lst.pop()
```

lst = [1, 2, 3]

current_lst = [1, 3]

ans_len = 3

```
permutation x
/usr/local/bin/python
[1, 2, 3]
[1, 3, 2]
```

```
def main():
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
    if len(current_lst) == ans_len:
```

```
        print(current_lst)
```

```
    else:
```

```
        for num in lst:
```

num = 3

```
            if num in current_lst:
```

```
                pass
```

```
            else:
```

```
                current_lst.append(num)
```

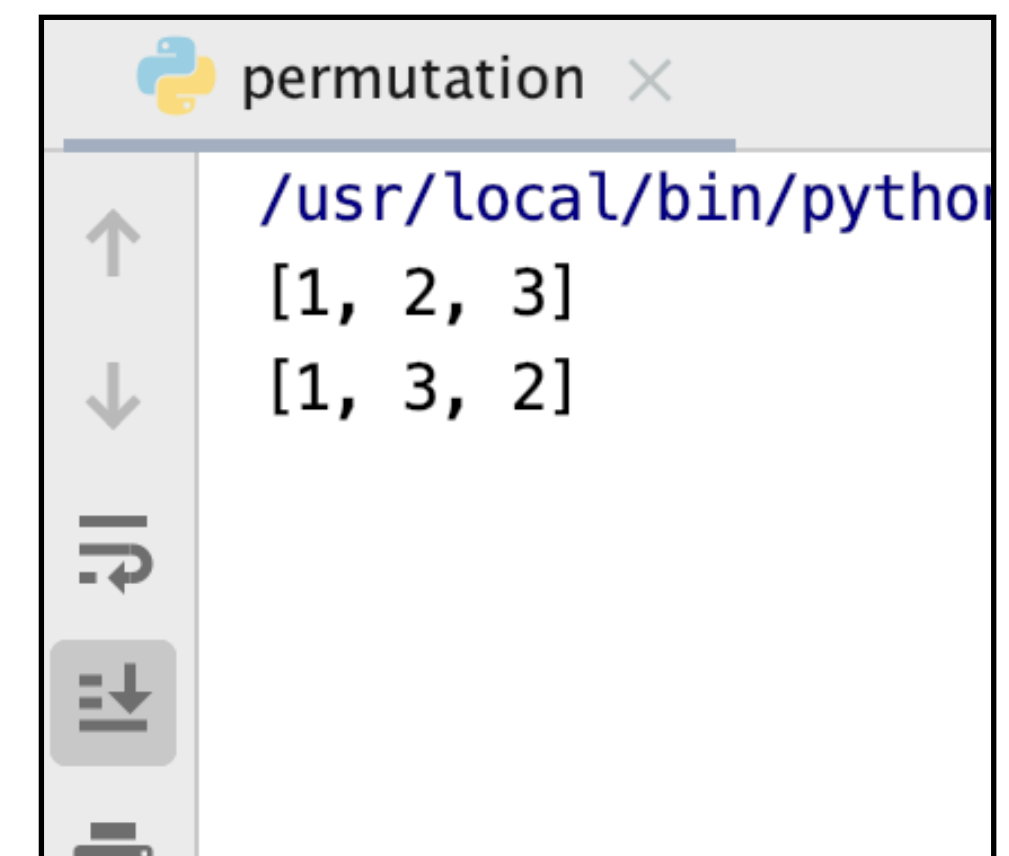
```
                permutation_helper(lst, current_lst, ans_len)
```

```
                current_lst.pop()
```

lst = [1, 2, 3]

current_lst = [1, 3]

ans_len = 3



A terminal window titled "permutation" with a close button. It shows the command `/usr/local/bin/python` and its output: `[1, 2, 3]` and `[1, 3, 2]`. The terminal has a vertical toolbar on the left with icons for back, forward, search, and other navigation functions.

```
permutation x
/usr/local/bin/python
[1, 2, 3]
[1, 3, 2]
```

```
def main():
```

```
def permutation(lst):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
    if len(current_lst) == ans_len:
```

```
        print(current_lst)
```

```
    else:
```

```
        for num in lst:
```

num = 3

```
            if num in current_lst:
```

```
                pass
```

```
            else:
```

```
                current_lst.append(num)
```

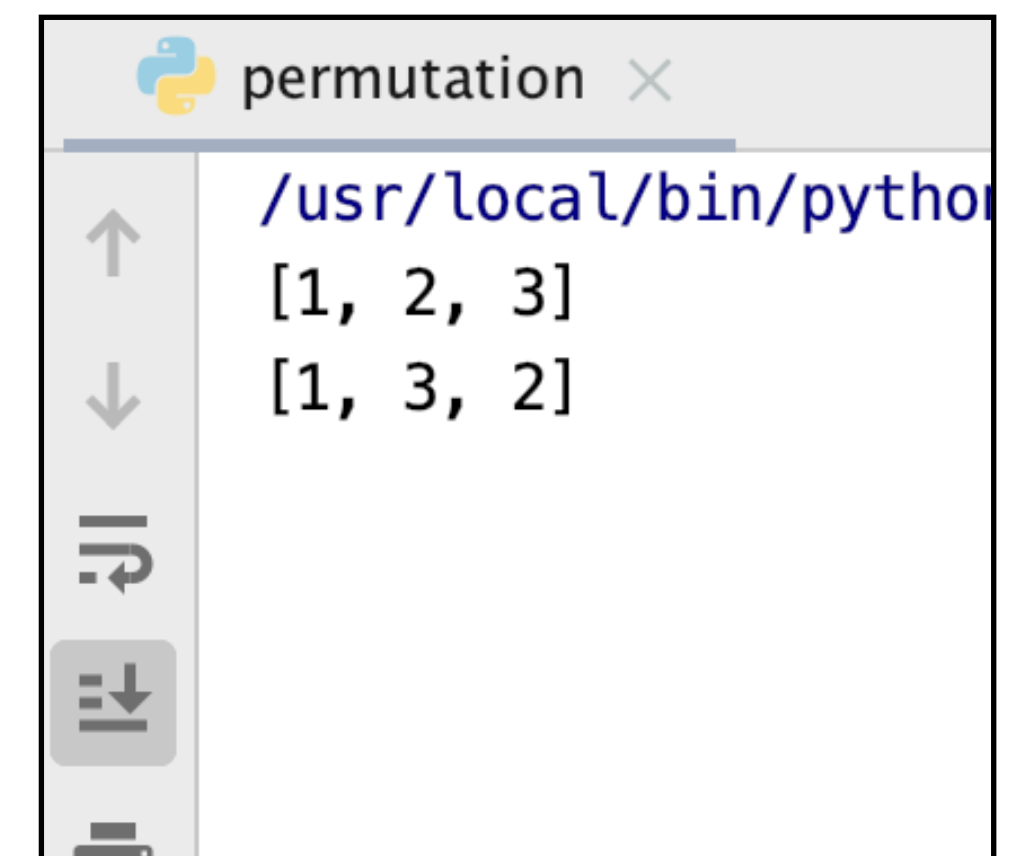
```
                permutation_helper(lst, current_lst, ans_len)
```

```
                current_lst.pop()
```

lst = [1, 2, 3]

current_lst = [1]

ans_len = 3



A terminal window titled "permutation" with a close button. It shows the command `/usr/local/bin/python` and the output of the program, which lists the permutations [1, 2, 3] and [1, 3, 2]. The terminal has a vertical toolbar on the left with icons for back, forward, search, and other navigation functions.

```
permutation ×  
/usr/local/bin/python  
[1, 2, 3]  
[1, 3, 2]
```



```
def main():
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
    if len(current_lst) == ans_len:
```

```
        print(current_lst)
```

```
    else:
```

```
        for num in lst:
```

```
            if num in current_lst:
```

```
                pass
```

```
            else:
```

```
                current_lst.append(num)
```

```
                permutation_helper(lst, current_lst, ans_len)
```

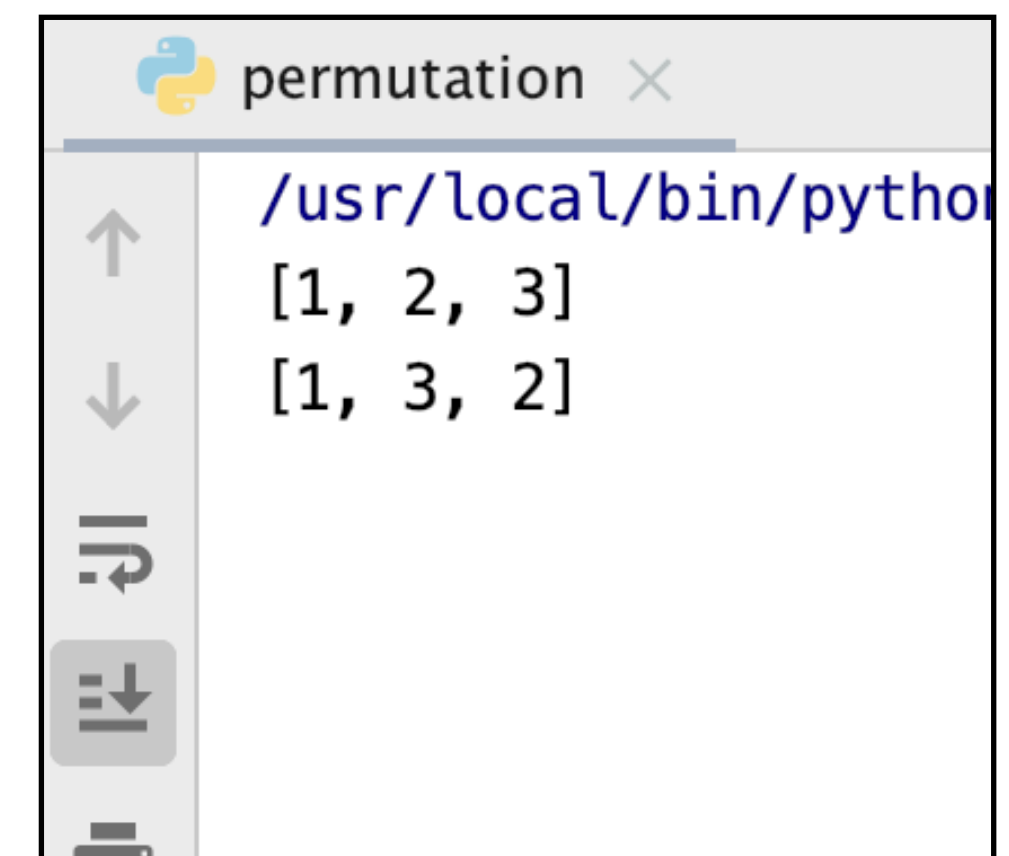
```
                current_lst.pop()
```

num = 1

lst = [1, 2, 3]

current_lst = [1]

ans_len = 3



```
permutation x
/usr/local/bin/python
[1, 2, 3]
[1, 3, 2]
```

```

def main():
    def permutation_helper(lst, current_lst, ans_len):
        if len(current_lst) == ans_len:
            print(current_lst)
        else:
            for num in lst:
                if num in current_lst:
                    pass
                else:
                    current_lst.append(num)
                    permutation_helper(lst, current_lst, ans_len)
                    current_lst.pop()

```

num = 1

lst = [1, 2, 3]

current_lst = []

ans_len = 3

```

permutation x
/usr/local/bin/python
[1, 2, 3]
[1, 3, 2]

```



```
def main():
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
    if len(current_lst) == ans_len:
```

```
        print(current_lst)
```

```
    else:
```

```
        for num in lst:
```

num = 2

```
            if num in current_lst:
```

```
                pass
```

```
            else:
```

```
                current_lst.append(num)
```

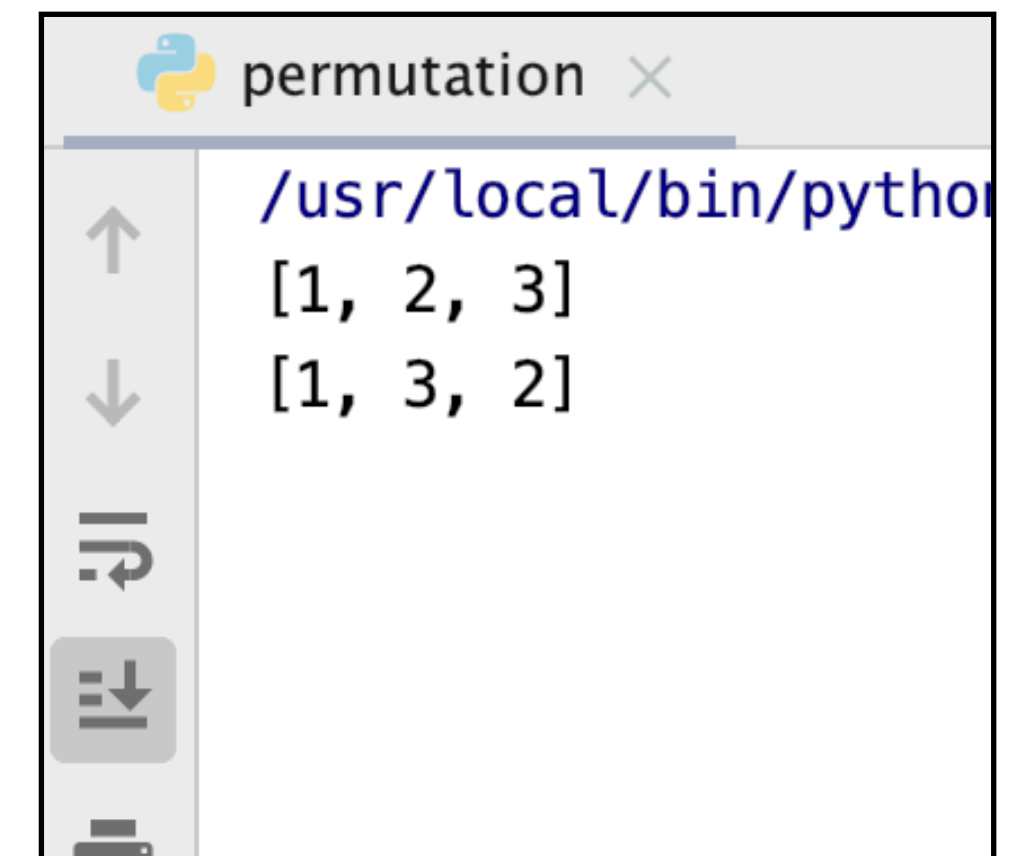
```
                permutation_helper(lst, current_lst, ans_len)
```

```
                current_lst.pop()
```

lst = [1, 2, 3]

current_lst = []

ans_len = 3



```
permutation x
/usr/local/bin/python
[1, 2, 3]
[1, 3, 2]
```

```

def main():
    def permutation_helper(lst, current_lst, ans_len):
        if len(current_lst) == ans_len:
            print(current_lst)
        else:
            for num in lst:
                if num in current_lst:
                    pass
                else:
                    current_lst.append(num)
                    permutation_helper(lst, current_lst, ans_len)
                    current_lst.pop()

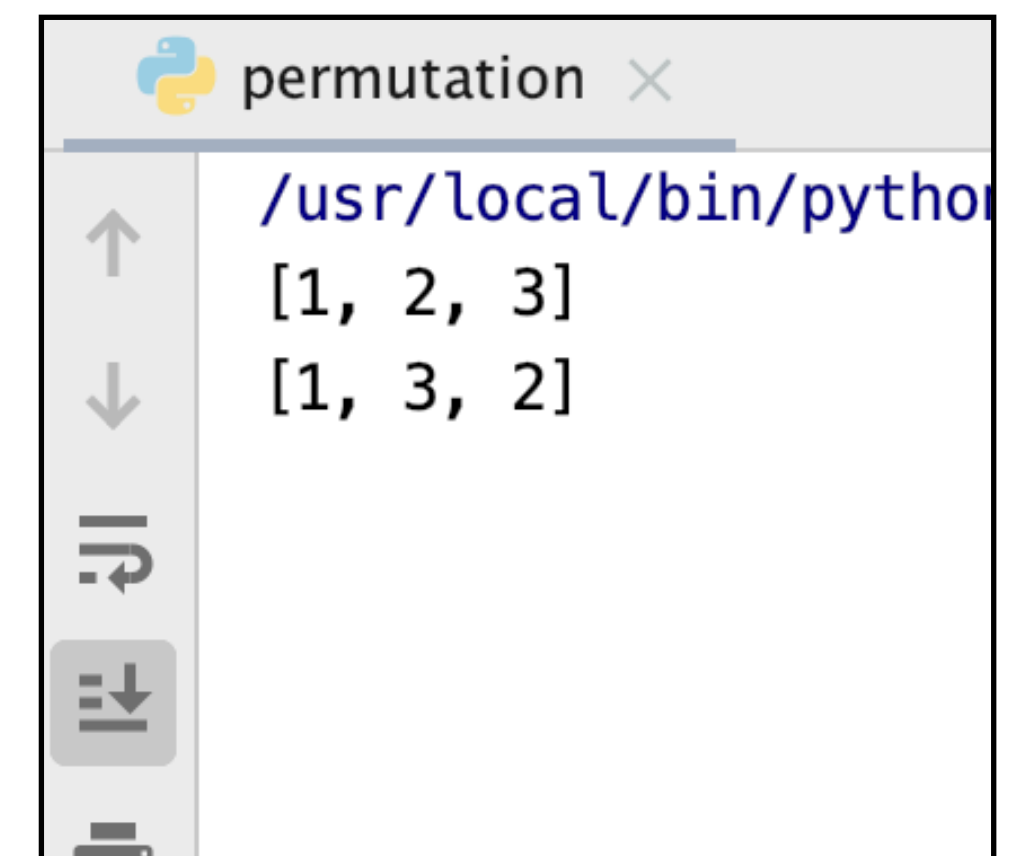
```

num = 2

lst = [1, 2, 3]

current_lst = []

ans_len = 3



```

permutation x
/usr/local/bin/python
[1, 2, 3]
[1, 3, 2]

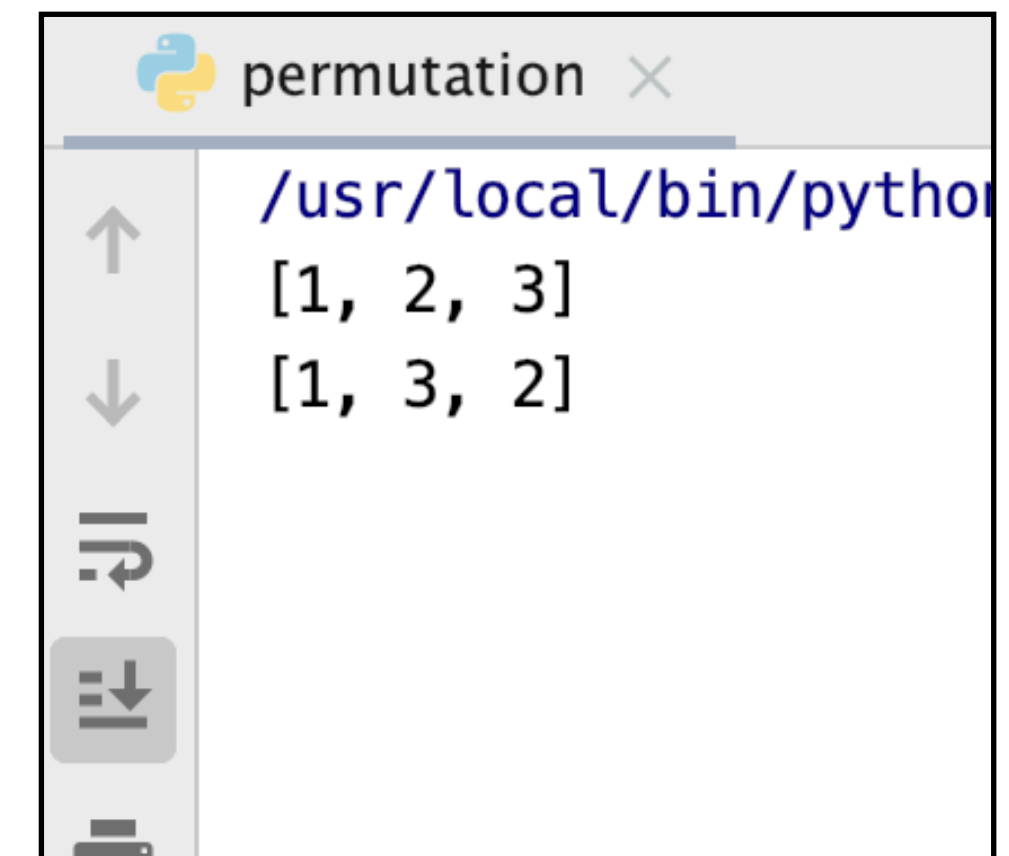
```

```
def main():  
    def permutation_helper(lst, current_lst, ans_len):  
        if len(current_lst) == ans_len:  
            print(current_lst)  
        else:  
            for num in lst:  
                if num in current_lst:  
                    pass  
                else:  
                    current_lst.append(num)  
                    permutation_helper(lst, current_lst, ans_len)  
                    current_lst.pop()
```

lst = [1, 2, 3]

current_lst = []

ans_len = 3



The screenshot shows a terminal window with the title 'permutation'. The command prompt is '/usr/local/bin/python'. The output of the program is two lines: '[1, 2, 3]' and '[1, 3, 2]'. The terminal has a standard interface with a cursor at the end of the command line and a vertical scrollbar on the left.

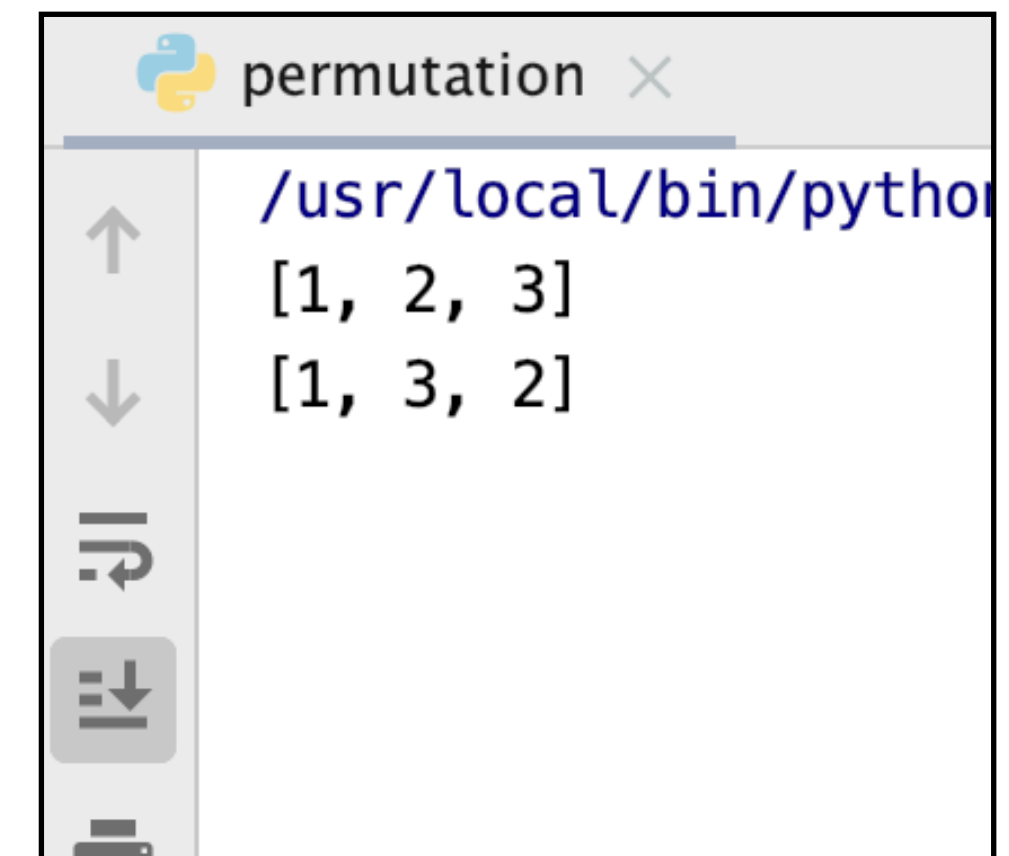
```
permutation ×  
/usr/local/bin/python  
[1, 2, 3]  
[1, 3, 2]
```

```
def main():  
    def permutation_helper(lst, current_lst, ans_len):  
        if len(current_lst) == ans_len:  
            print(current_lst)  
        else:  
            for num in lst:  
                if num in current_lst:  
                    pass  
                else:  
                    current_lst.append(num)  
                    permutation_helper(lst, current_lst, ans_len)  
                    current_lst.pop()
```

lst = [1, 2, 3]

current_lst = [2]

ans_len = 3



```
permutation x  
/usr/local/bin/python  
[1, 2, 3]  
[1, 3, 2]
```

```
def main():
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
    if len(current_lst) == ans_len:
```

```
        print(current_lst)
```

```
    else:
```

```
        for num in lst:
```

num = 2

```
            if num in current_lst:
```

```
                pass
```

```
            else:
```

```
                current_lst.append(num)
```

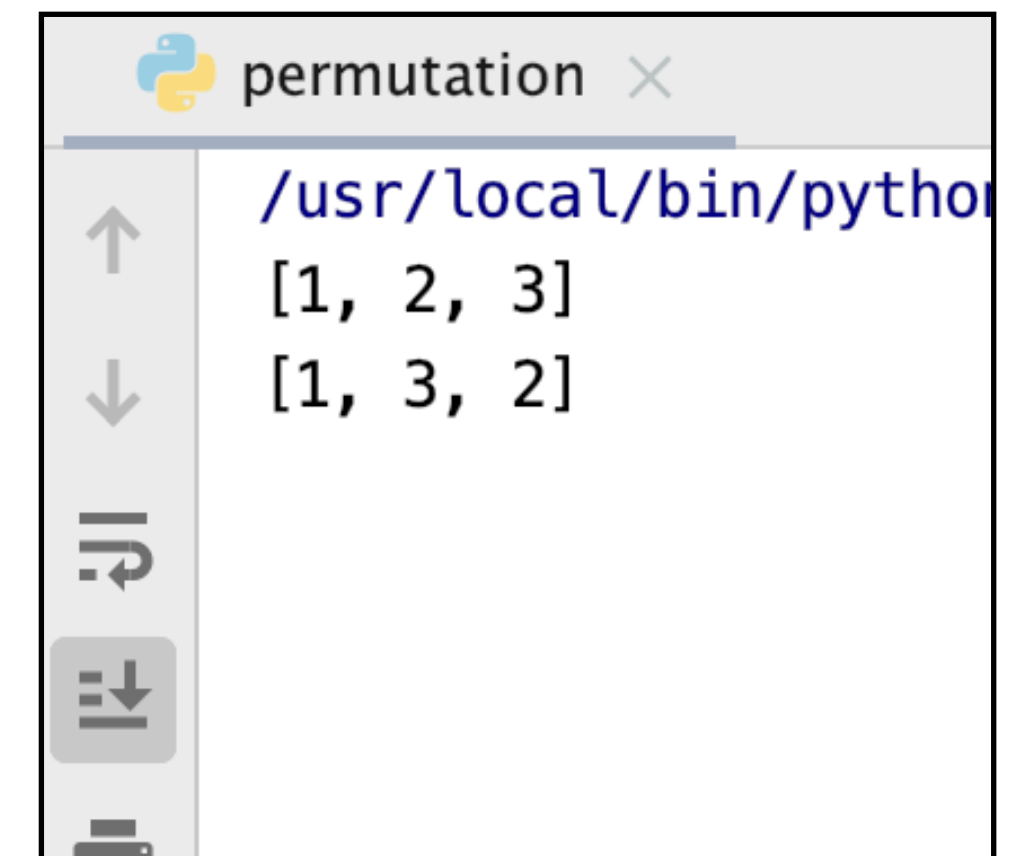
```
                permutation_helper(lst, current_lst, ans_len)
```

```
                current_lst.pop()
```

lst = [1, 2, 3]

current_lst = [2]

ans_len = 3



```
permutation x
/usr/local/bin/python
[1, 2, 3]
[1, 3, 2]
```

```
def main():
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
    if len(current_lst) == ans_len:
```

```
        print(current_lst)
```

```
    else:
```

```
        for num in lst:
```

num = 2

```
            if num in current_lst:
```

```
                pass
```

```
            else:
```

```
                current_lst.append(num)
```

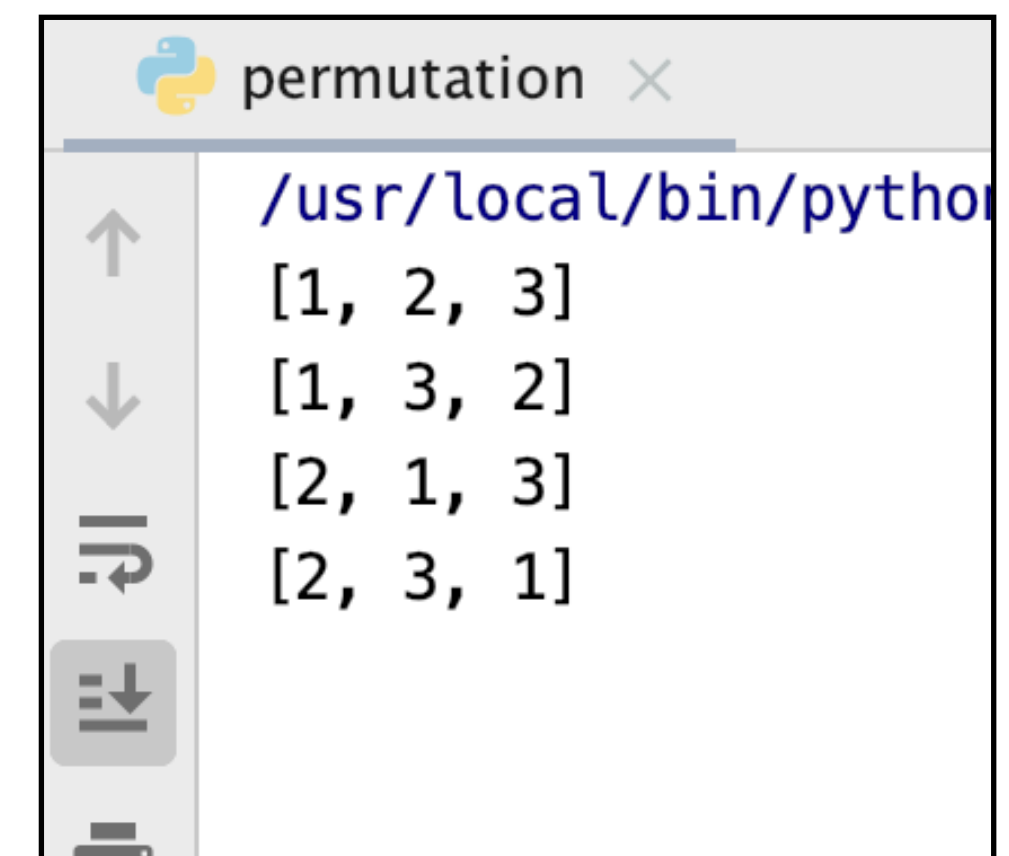
```
                permutation_helper(lst, current_lst, ans_len)
```

```
                current_lst.pop()
```

lst = [1, 2, 3]

current_lst = [2]

ans_len = 3



```
permutation x
/usr/local/bin/python
[1, 2, 3]
[1, 3, 2]
[2, 1, 3]
[2, 3, 1]
```

```

def main():
    def permutation_helper(lst, current_lst, ans_len):
        if len(current_lst) == ans_len:
            print(current_lst)
        else:
            for num in lst:
                if num in current_lst:
                    pass
                else:
                    current_lst.append(num)
                    permutation_helper(lst, current_lst, ans_len)
                    current_lst.pop()

```

num = 2

lst = [1, 2, 3]

current_lst = []

ans_len = 3

```

permutation x
/usr/local/bin/python
[1, 2, 3]
[1, 3, 2]
[2, 1, 3]
[2, 3, 1]

```

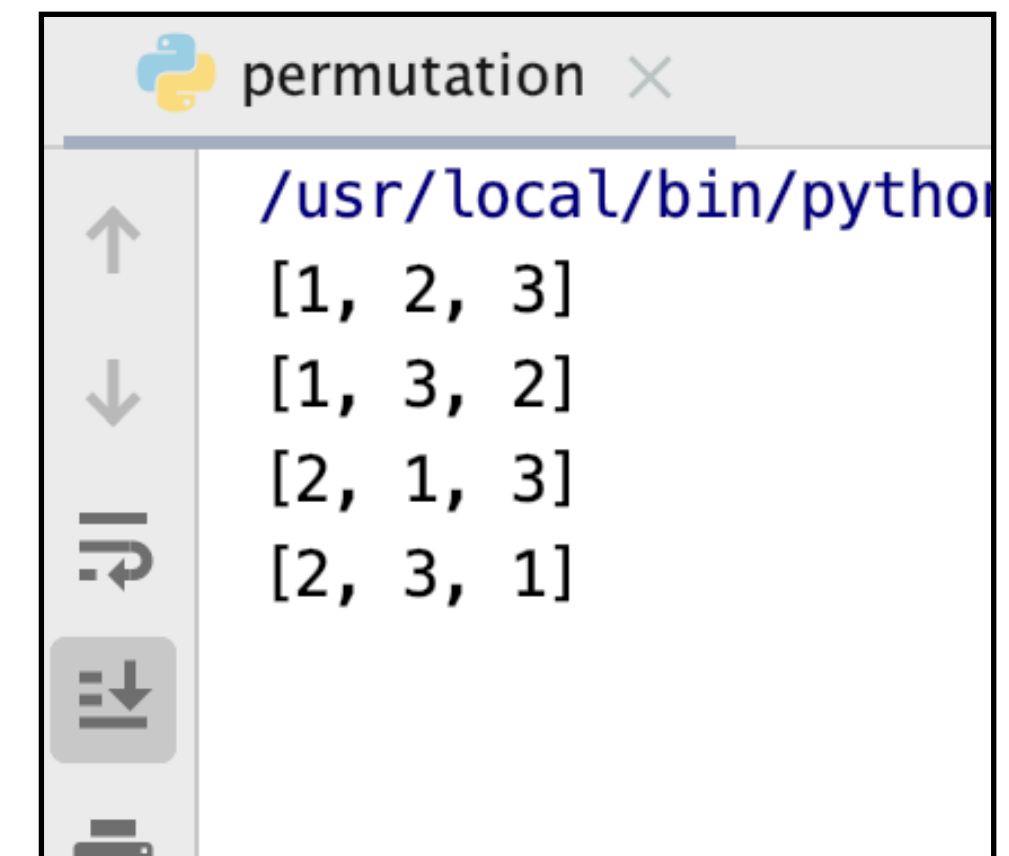


```
def main():  
    def permutation_helper(lst, current_lst, ans_len):  
        if len(current_lst) == ans_len:  
            print(current_lst)  
        else:  
            for num in lst:  
                if num in current_lst:  
                    pass  
                else:  
                    current_lst.append(num)  
                    permutation_helper(lst, current_lst, ans_len)  
                    current_lst.pop()
```

lst = [1, 2, 3]

current_lst = []

ans_len = 3



```
permutation x  
/usr/local/bin/python  
[1, 2, 3]  
[1, 3, 2]  
[2, 1, 3]  
[2, 3, 1]
```

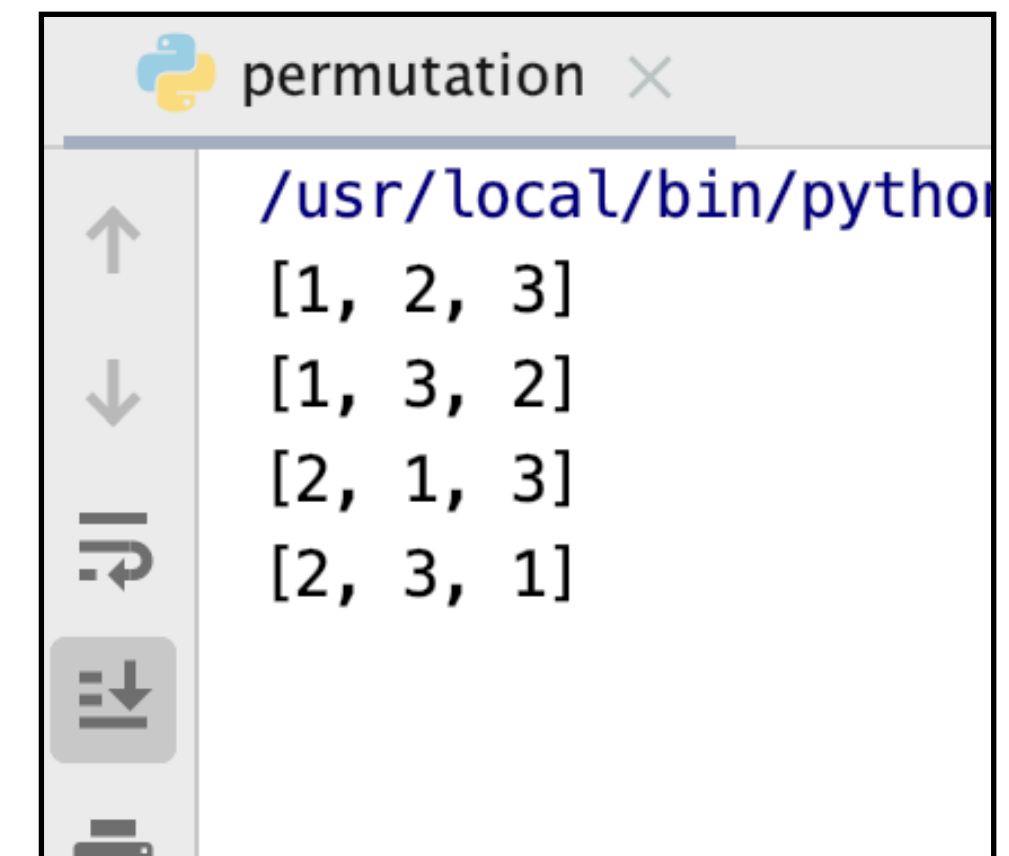


```
def main():  
    def permutation_helper(lst, current_lst, ans_len):  
        if len(current_lst) == ans_len:  
            print(current_lst)  
        else:  
            for num in lst:  
                if num in current_lst:  
                    pass  
                else:  
                    current_lst.append(num)  
                    permutation_helper(lst, current_lst, ans_len)  
                    current_lst.pop()
```

lst = [1, 2, 3]

current_lst = []

ans_len = 3



```
permutation x  
/usr/local/bin/python  
[1, 2, 3]  
[1, 3, 2]  
[2, 1, 3]  
[2, 3, 1]
```

```
def main():
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
    if len(current_lst) == ans_len:
```

```
        print(current_lst)
```

```
    else:
```

```
        for num in lst:
```

```
            if num in current_lst:
```

```
                pass
```

```
            else:
```

```
                current_lst.append(num)
```

```
                permutation_helper(lst, current_lst, ans_len)
```

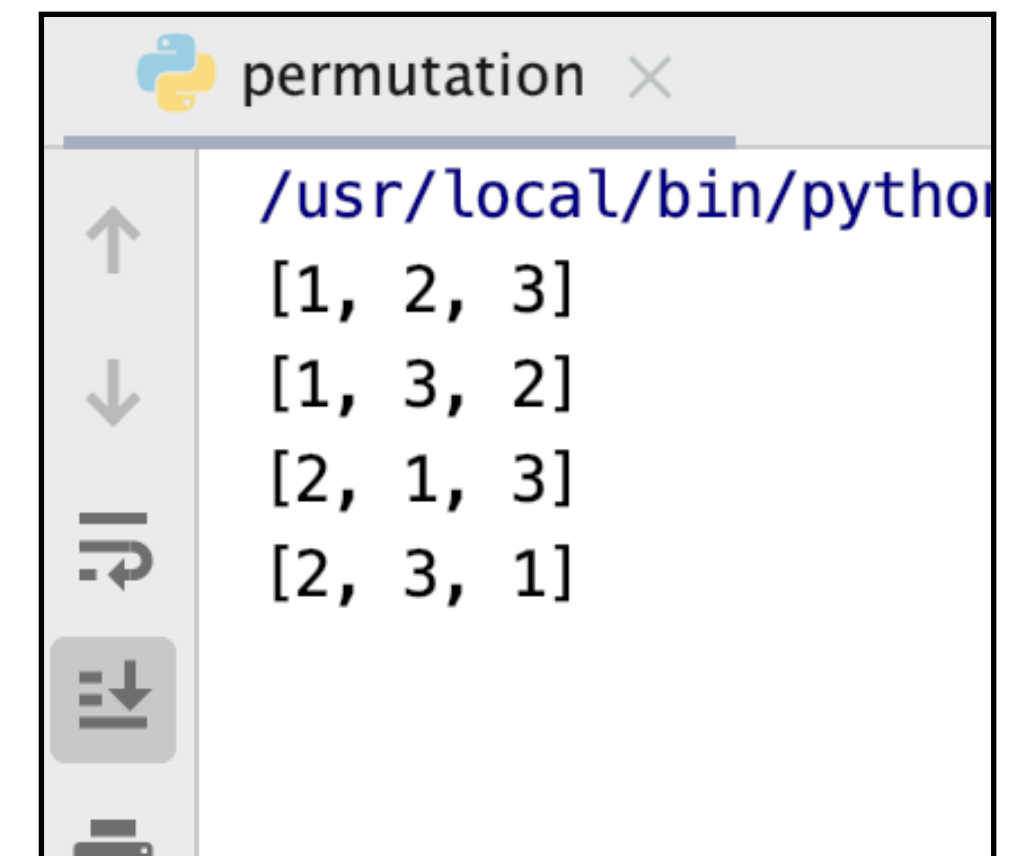
```
                current_lst.pop()
```

num = 3

lst = [1, 2, 3]

current_lst = []

ans_len = 3



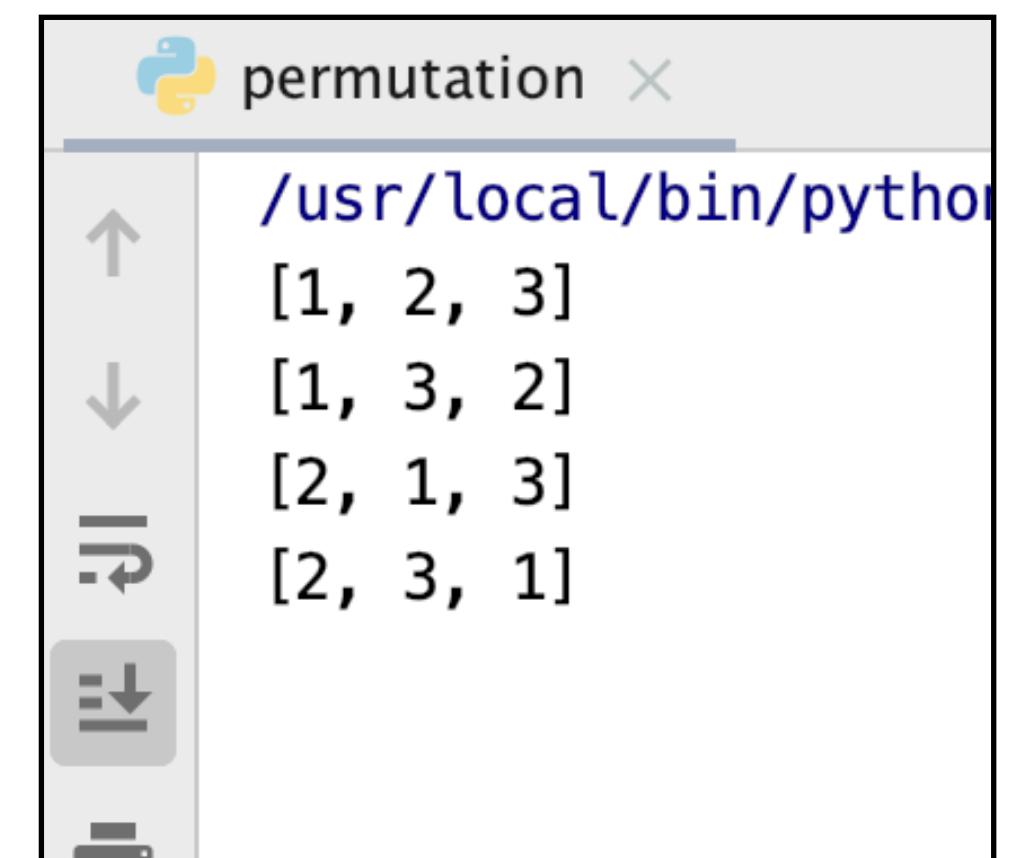
```
permutation x
/usr/local/bin/python
[1, 2, 3]
[1, 3, 2]
[2, 1, 3]
[2, 3, 1]
```

```
def main():  
    def permutation_helper(lst, current_lst, ans_len):  
        if len(current_lst) == ans_len:  
            print(current_lst)  
        else:  
            for num in lst:                num = 3  
                if num in current_lst:  
                    pass  
                else:  
                    current_lst.append(num)  
                    permutation_helper(lst, current_lst, ans_len)  
                    current_lst.pop()
```

lst = [1, 2, 3]

current_lst = [3]

ans_len = 3



```
permutation x  
/usr/local/bin/python  
[1, 2, 3]  
[1, 3, 2]  
[2, 1, 3]  
[2, 3, 1]
```

```
def main():
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
    if len(current_lst) == ans_len:
```

```
        print(current_lst)
```

```
    else:
```

```
        for num in lst:
```

num = 3

```
            if num in current_lst:
```

```
                pass
```

```
            else:
```

```
                current_lst.append(num)
```

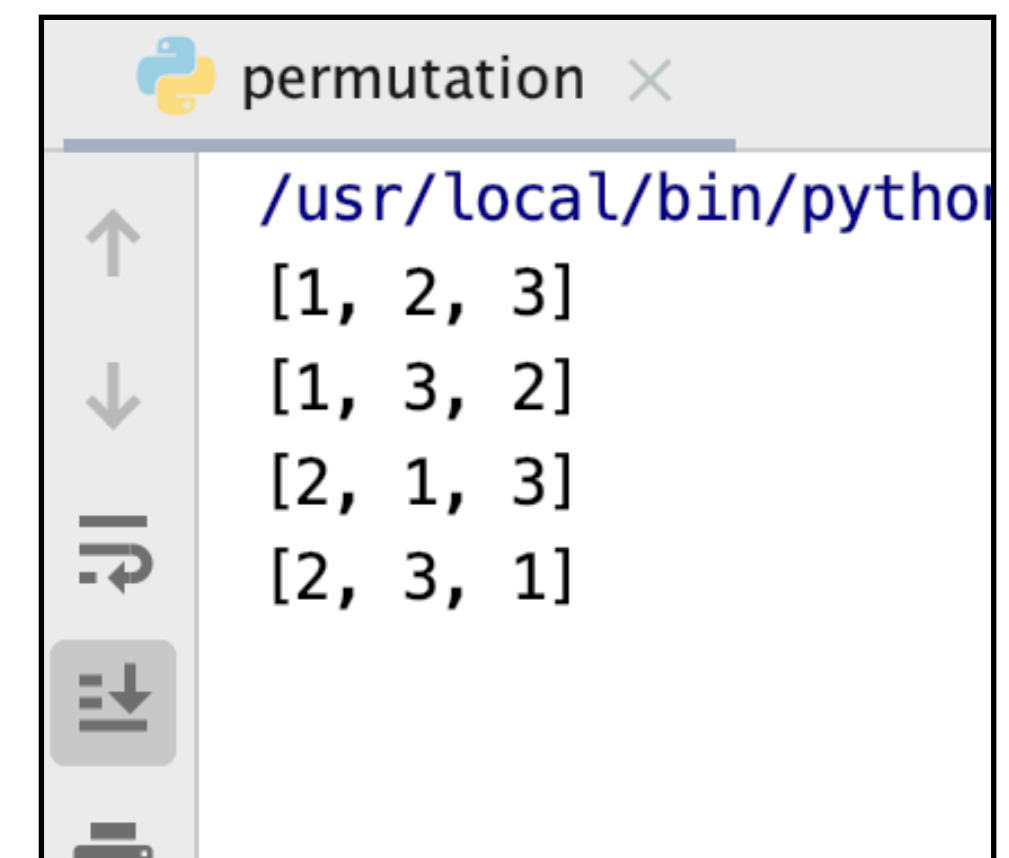
```
                permutation_helper(lst, current_lst, ans_len)
```

```
                current_lst.pop()
```

lst = [1, 2, 3]

current_lst = [3]

ans_len = 3



```
permutation x
/usr/local/bin/python
[1, 2, 3]
[1, 3, 2]
[2, 1, 3]
[2, 3, 1]
```

```
def main():
```

```
def permutation_helper(lst, current_lst, ans_len):
```

```
    if len(current_lst) == ans_len:
```

```
        print(current_lst)
```

```
    else:
```

```
        for num in lst:
```

num = 3

```
            if num in current_lst:
```

```
                pass
```

```
            else:
```

```
                current_lst.append(num)
```

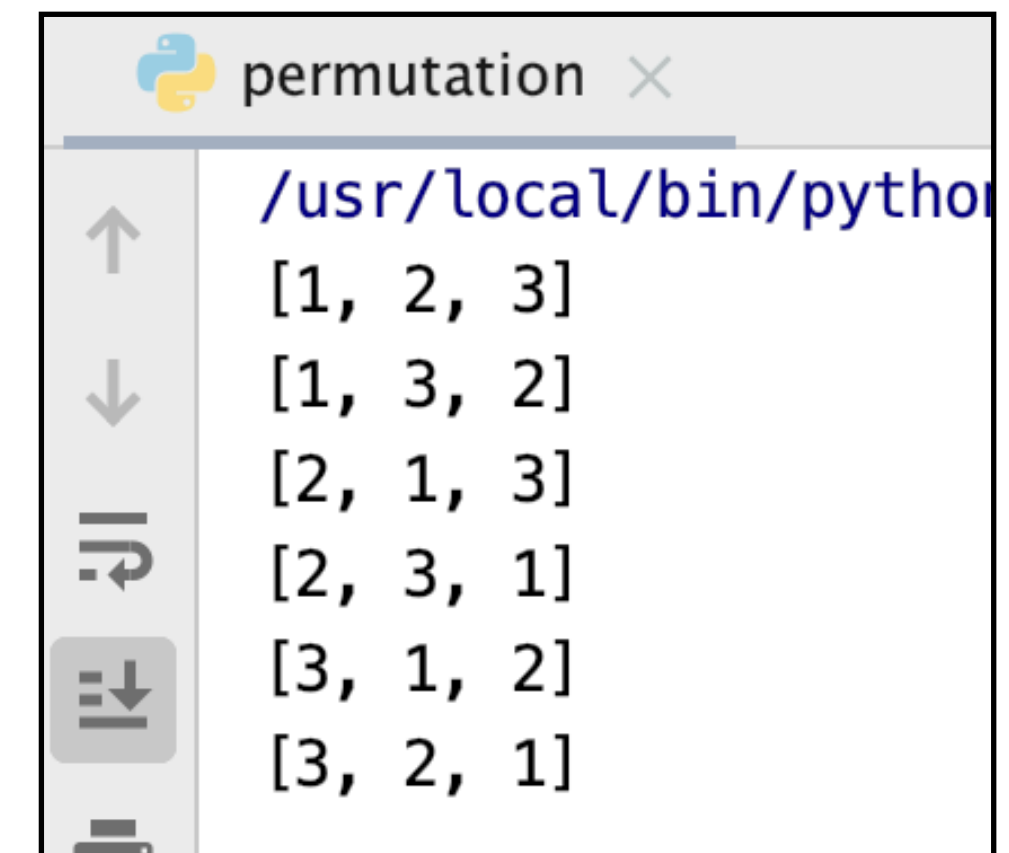
```
                permutation_helper(lst, current_lst, ans_len)
```

```
                current_lst.pop()
```

lst = [1, 2, 3]

current_lst = [3]

ans_len = 3



```
permutation x
/usr/local/bin/python
[1, 2, 3]
[1, 3, 2]
[2, 1, 3]
[2, 3, 1]
[3, 1, 2]
[3, 2, 1]
```

```

def main():
    def permutation_helper(lst, current_lst, ans_len):
        if len(current_lst) == ans_len:
            print(current_lst)
        else:
            for num in lst:
                if num in current_lst:
                    pass
                else:
                    current_lst.append(num)
                    permutation_helper(lst, current_lst, ans_len)
                    current_lst.pop()

```

num = 3

lst = [1, 2, 3]

current_lst = []

ans_len = 3

```

permutation x
/usr/local/bin/python
[1, 2, 3]
[1, 3, 2]
[2, 1, 3]
[2, 3, 1]
[3, 1, 2]
[3, 2, 1]

```



```
1
2 def main():
3     subsets(['a', 'b', 'c', 'd'])
4
5
6 def subsets(lst):
7     subsets_helper(lst, [])
8
9
10 def subsets_helper(lst, chosen):
11     if len(lst) == 0:
12         print(chosen)
13     else:
14         # Choose
15         element = lst.pop()
16
17         # Explore without
18         subsets_helper(lst, chosen)
19
20         # Explore with
21         chosen.append(element)
22         subsets_helper(lst, chosen)
23
24         # Unchoose
25         chosen.pop()
26         lst.append(element)
27
```

lst = ['a', 'b'] / chosen = []

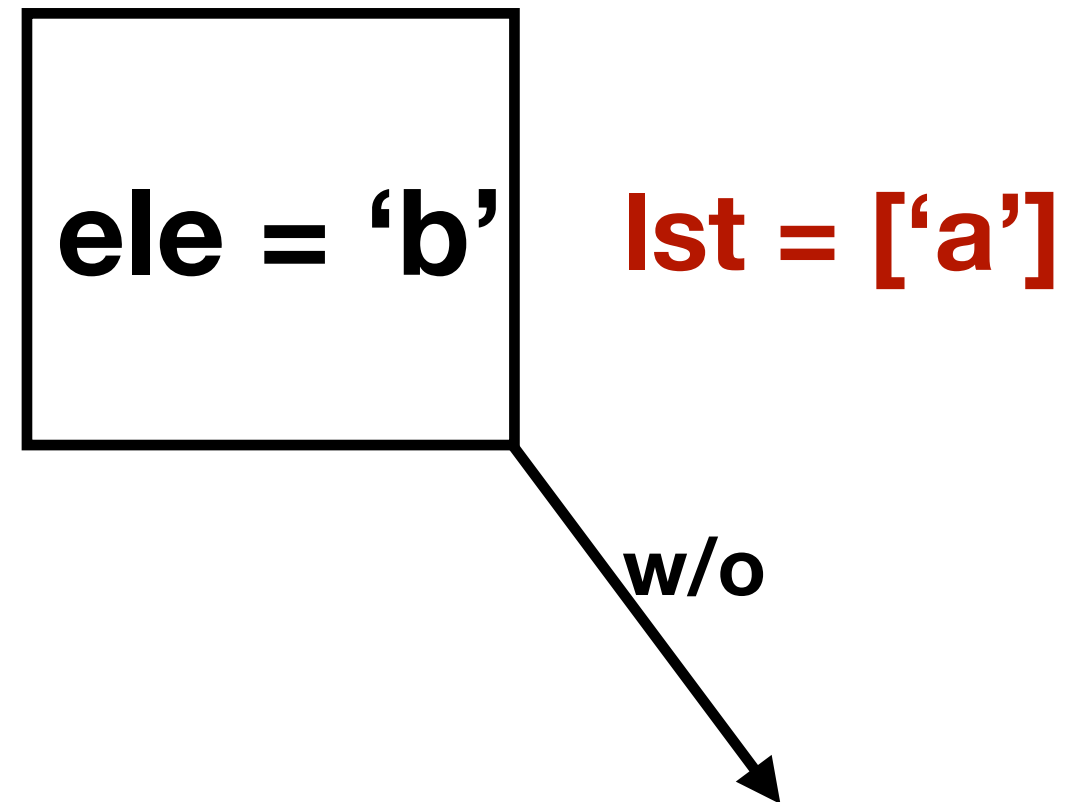
lst = ['a', 'b'] / chosen = []

ele = 'b'

lst = ['a', 'b'] / chosen = []

ele = 'b' **lst = ['a']**

lst = ['a', 'b'] / chosen = []



lst = ['a', 'b'] / chosen = []

ele = 'b' **lst = ['a']**

w/o

lst = ['a'] / chosen = []

lst = ['a', 'b'] / chosen = []

ele = 'b' lst = ['a']

w/o

lst = ['a'] / chosen = []

ele = 'a'

lst = ['a', 'b'] / chosen = []

ele = 'b' lst = ['a']

w/o

lst = ['a'] / chosen = []

ele = 'a' lst = []

lst = ['a', 'b'] / chosen = []

ele = 'b' lst = ['a']

w/o

lst = ['a'] / chosen = []

ele = 'a' lst = []

w/o

lst = ['a', 'b'] / chosen = []

ele = 'b' **lst = ['a']**

w/o

lst = ['a'] / chosen = []

ele = 'a' **lst = []**

w/o

Base case!!

lst = ['a', 'b'] / chosen = []

ele = 'b' lst = ['a']

w/o

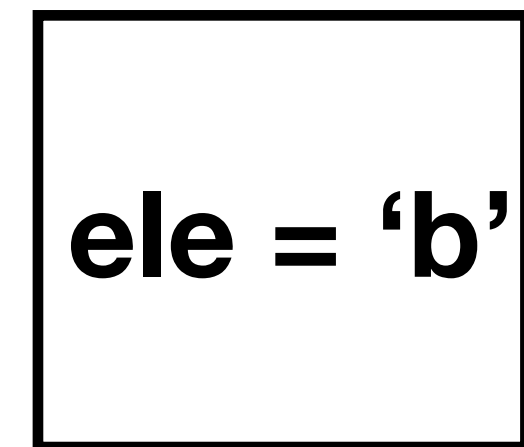
lst = ['a'] / chosen = []

ele = 'a' lst = []

w/o

**Base case!!
chosen = []**

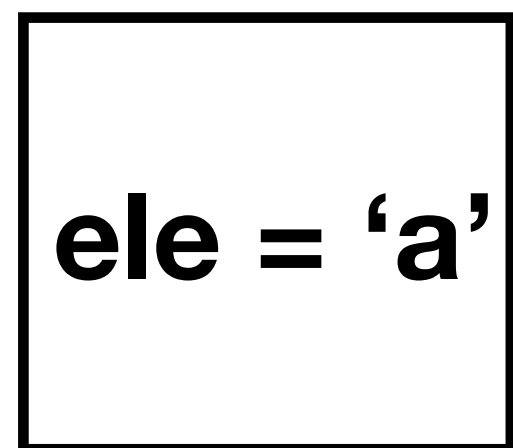
lst = ['a', 'b'] / chosen = []



lst = ['a']

w/o

lst = ['a'] / chosen = []



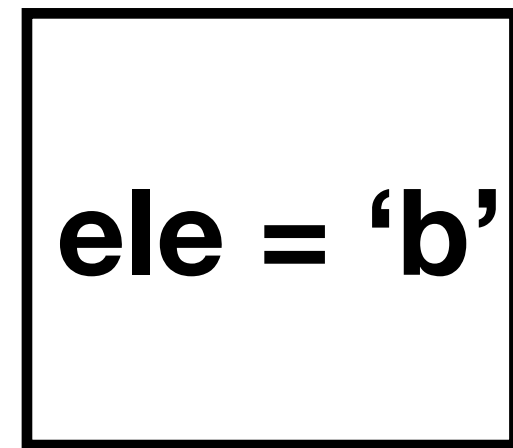
lst = []

w/

w/o

Base case!!
chosen = []

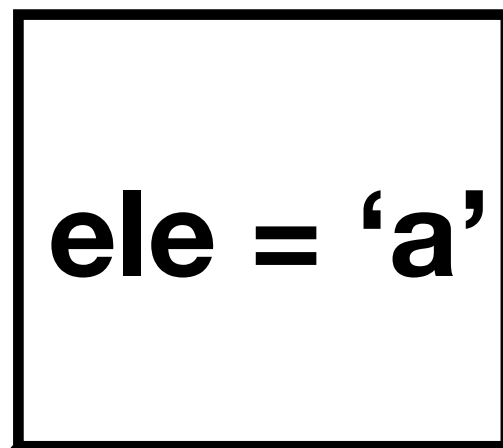
lst = ['a', 'b'] / chosen = []



lst = ['a']

w/o

lst = ['a'] / chosen = []



lst = []

w/

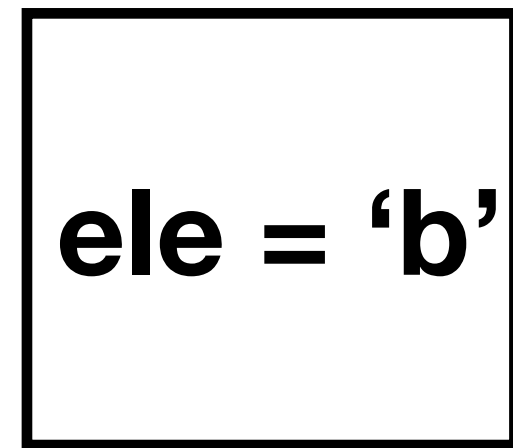
w/o

Base case!!

Base case!!

chosen = []

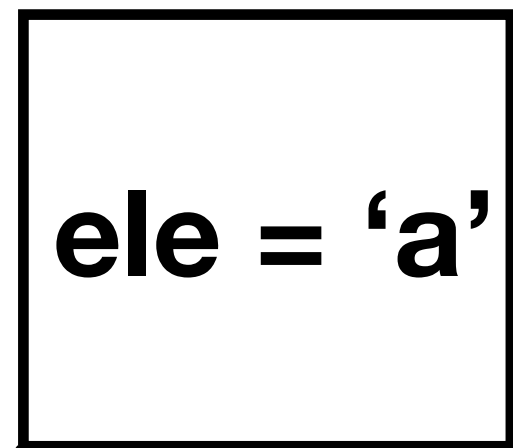
lst = ['a', 'b'] / chosen = []



lst = ['a']

w/o

lst = ['a'] / chosen = []



lst = []

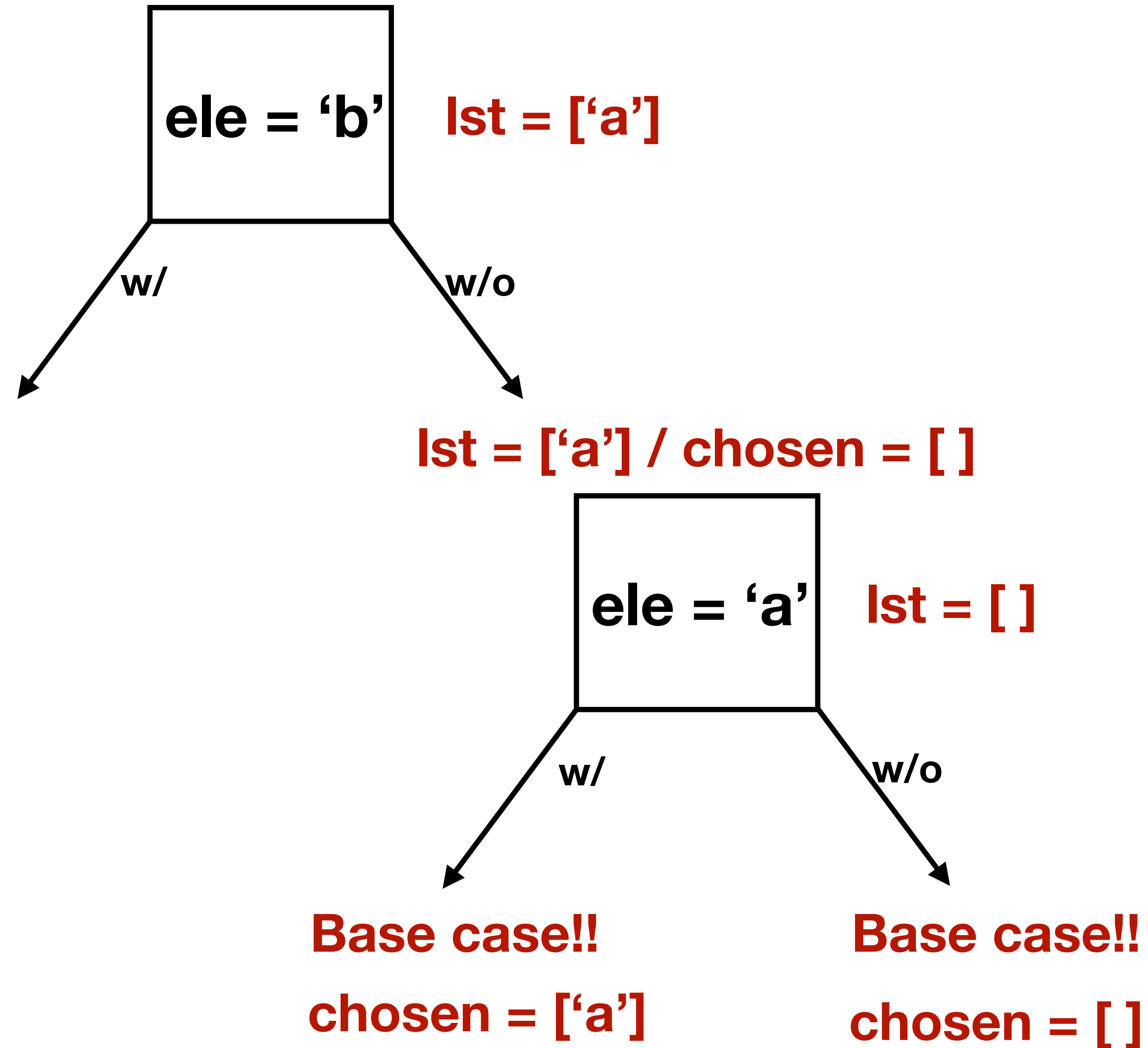
w/

w/o

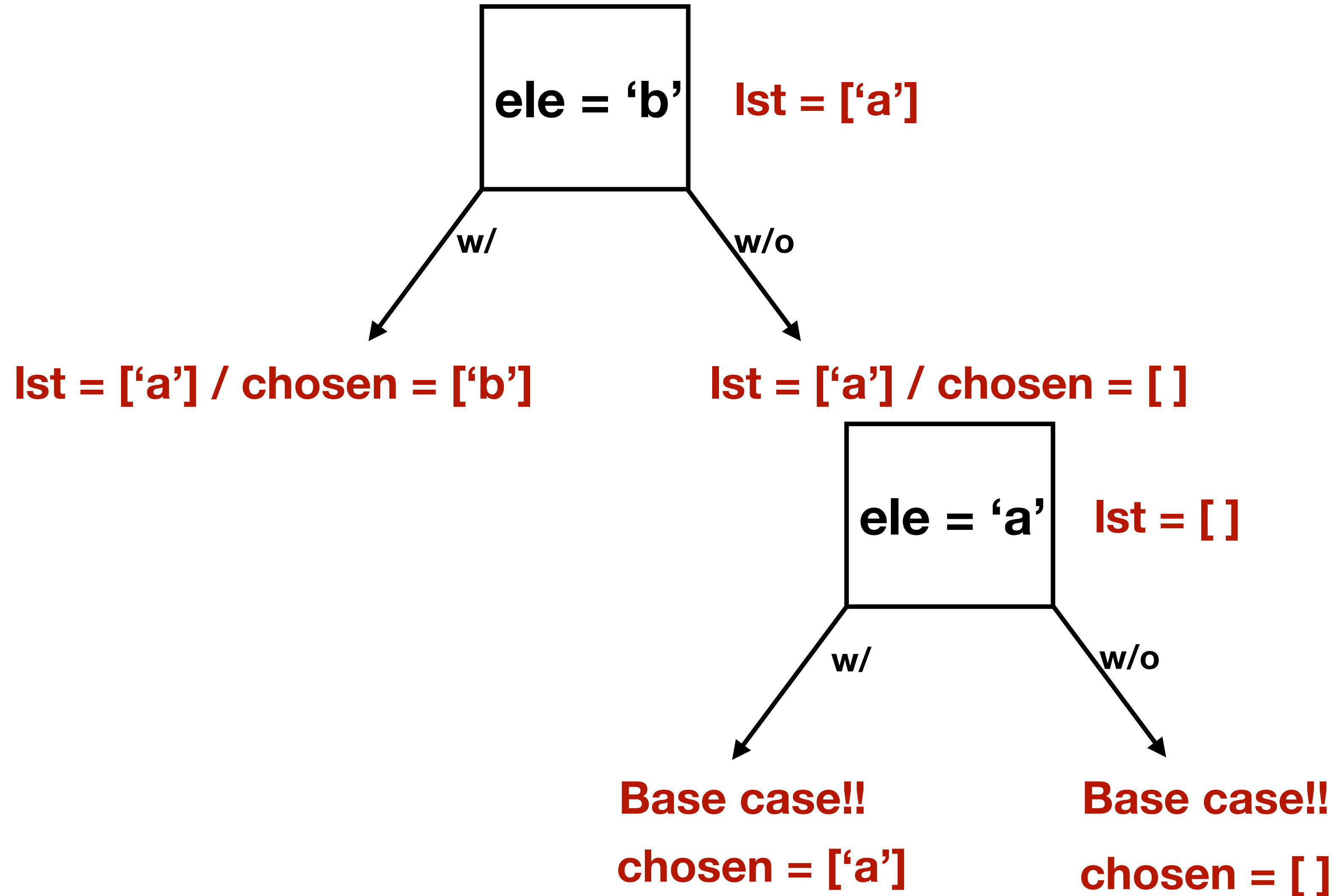
Base case!!
chosen = ['a']

Base case!!
chosen = []

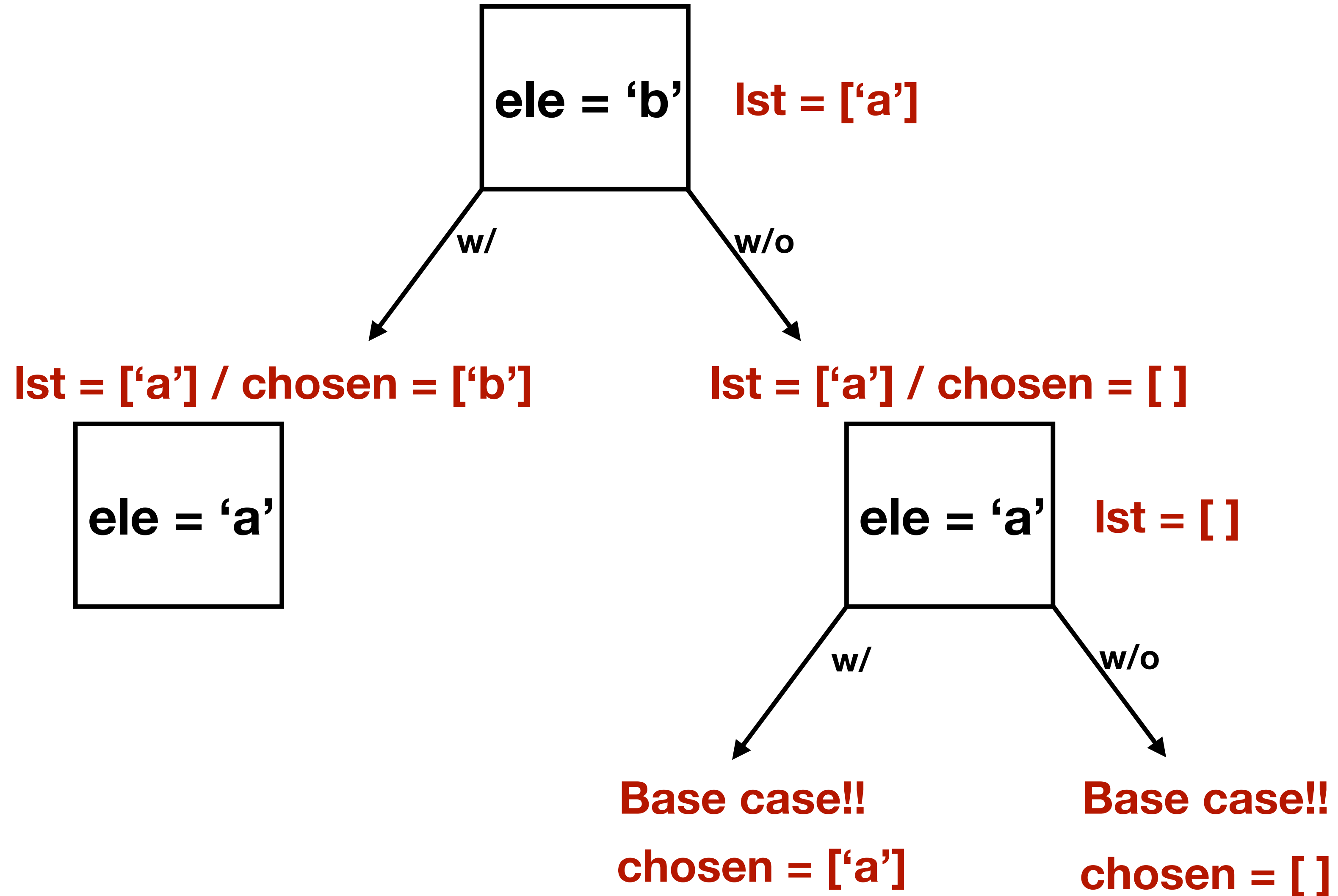
lst = ['a', 'b'] / chosen = []



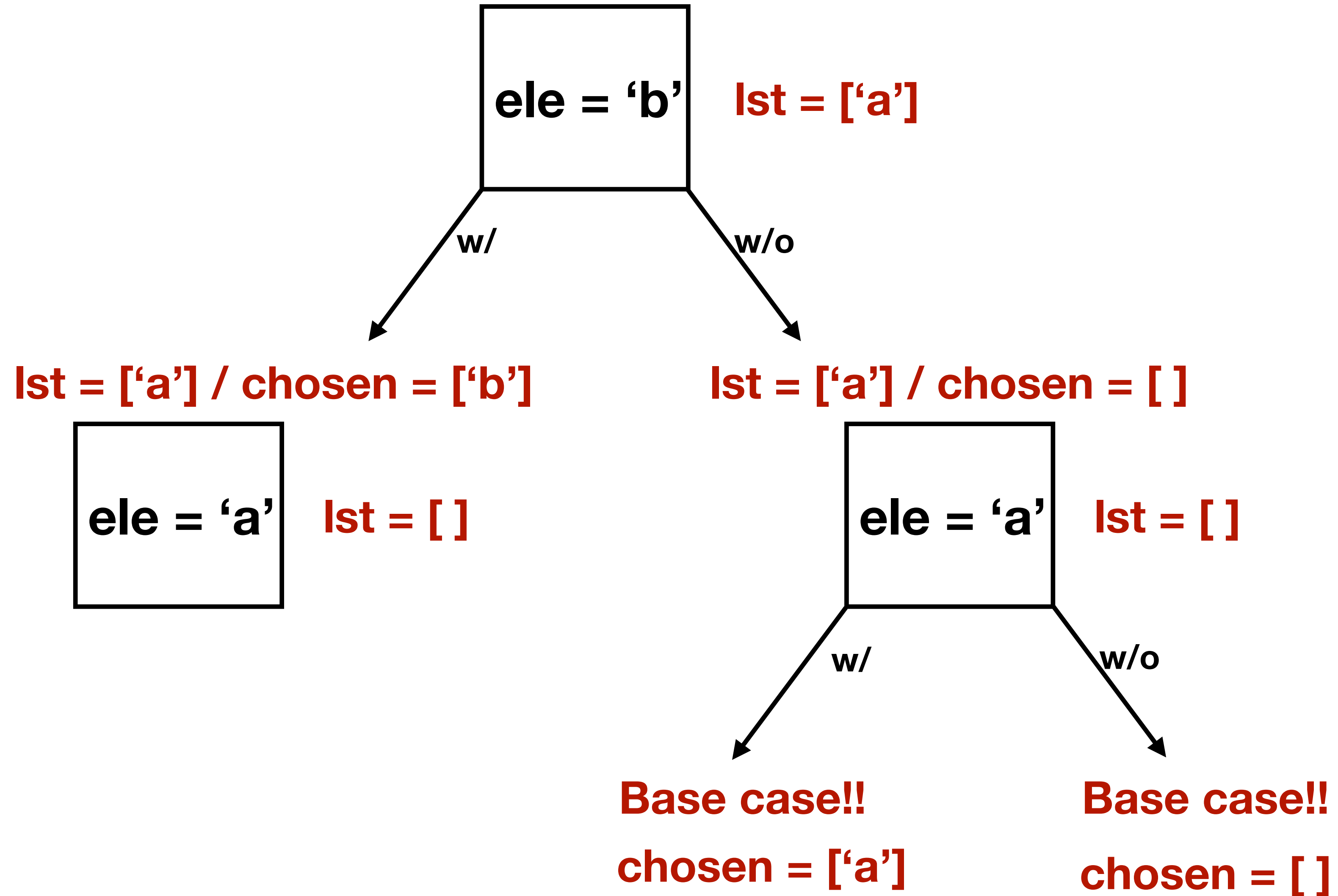
lst = ['a', 'b'] / chosen = []



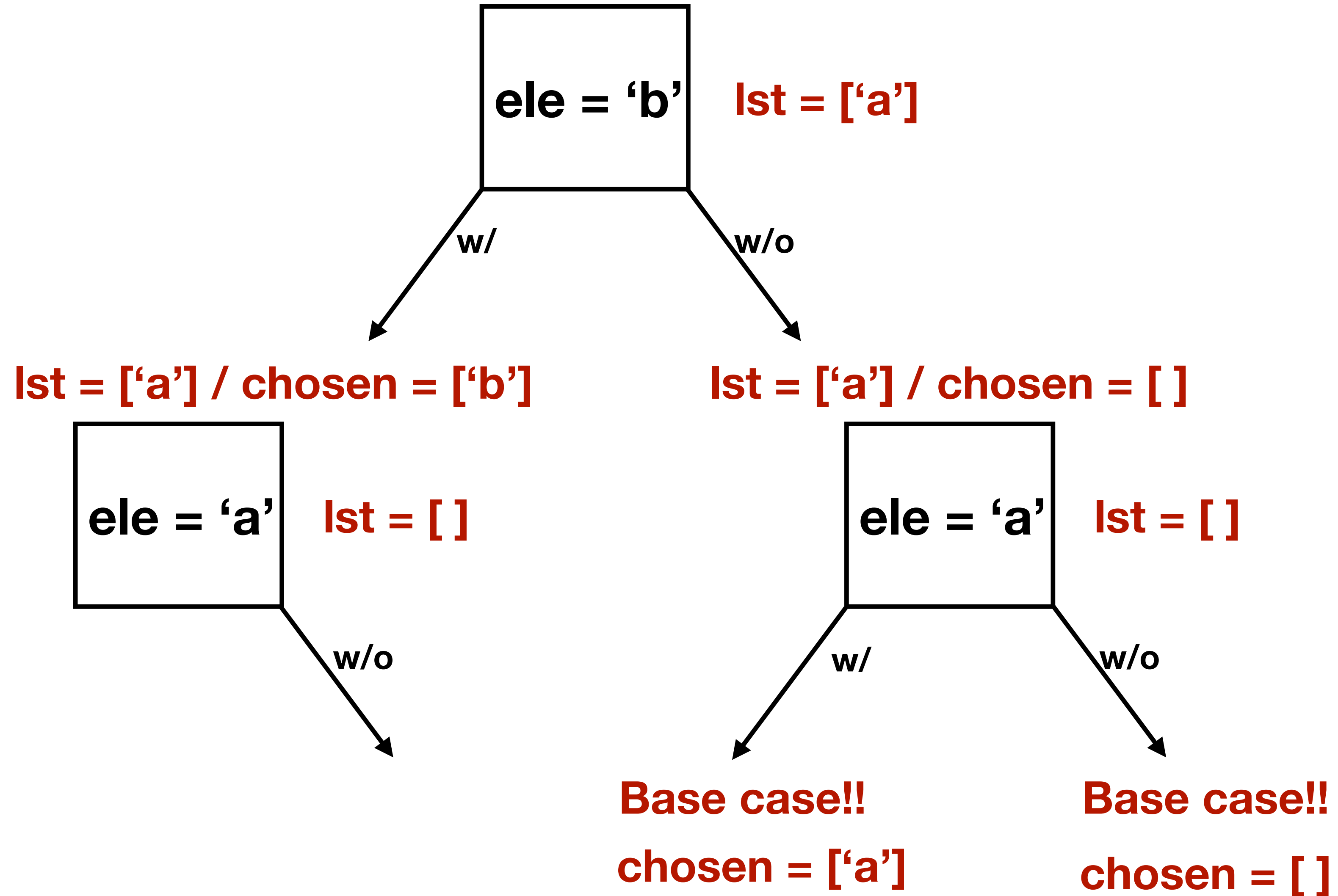
lst = ['a', 'b'] / chosen = []



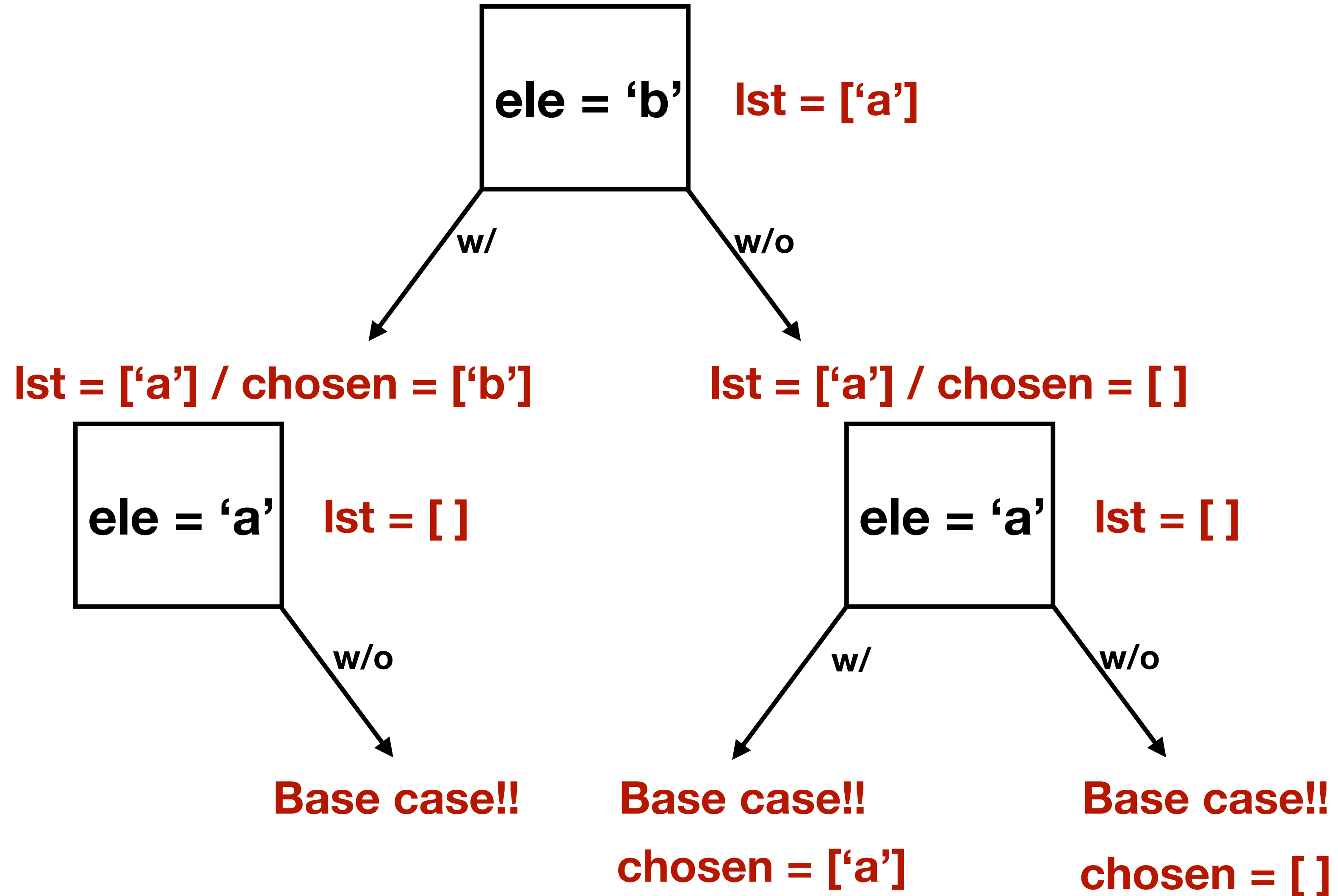
lst = ['a', 'b'] / chosen = []



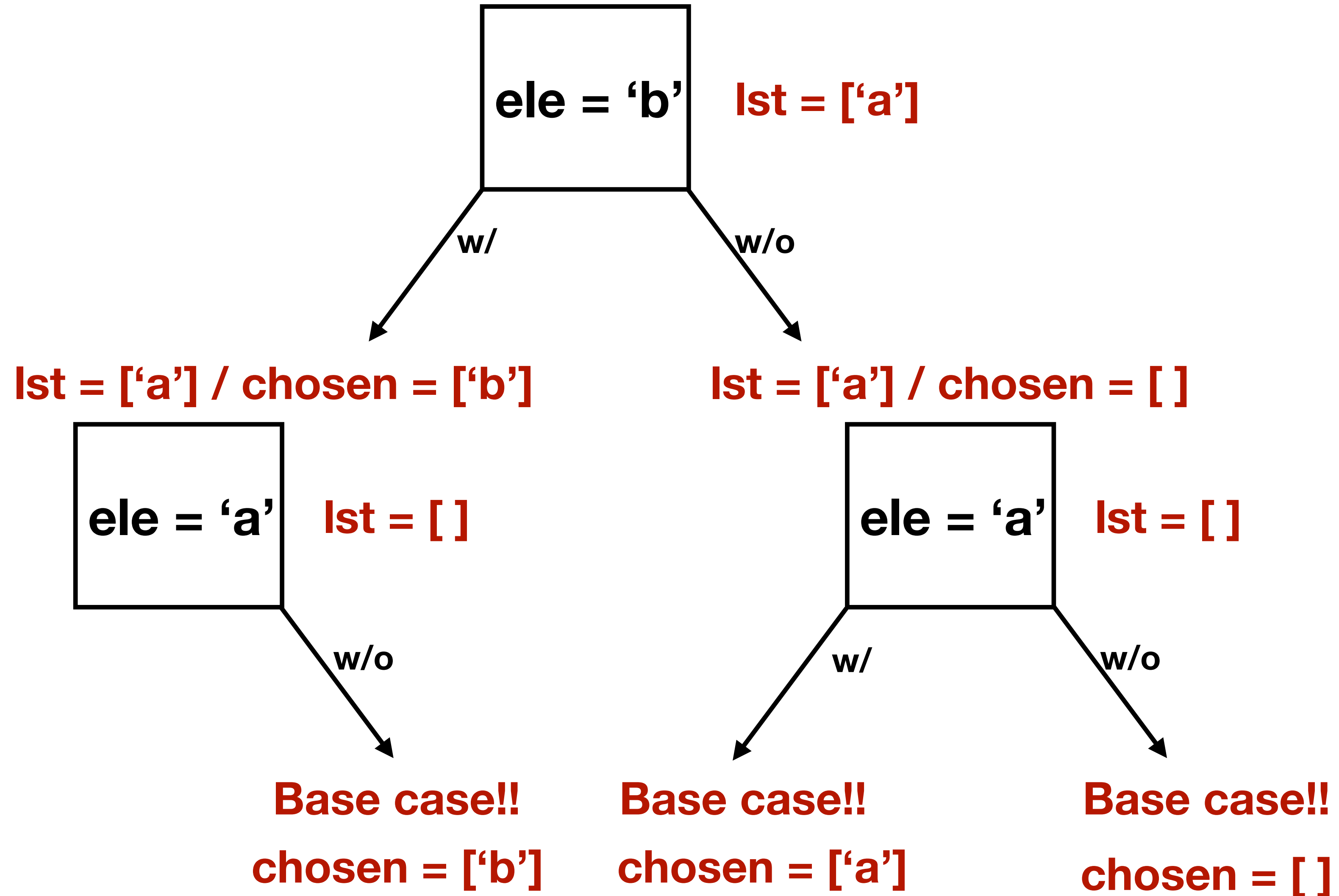
lst = ['a', 'b'] / chosen = []



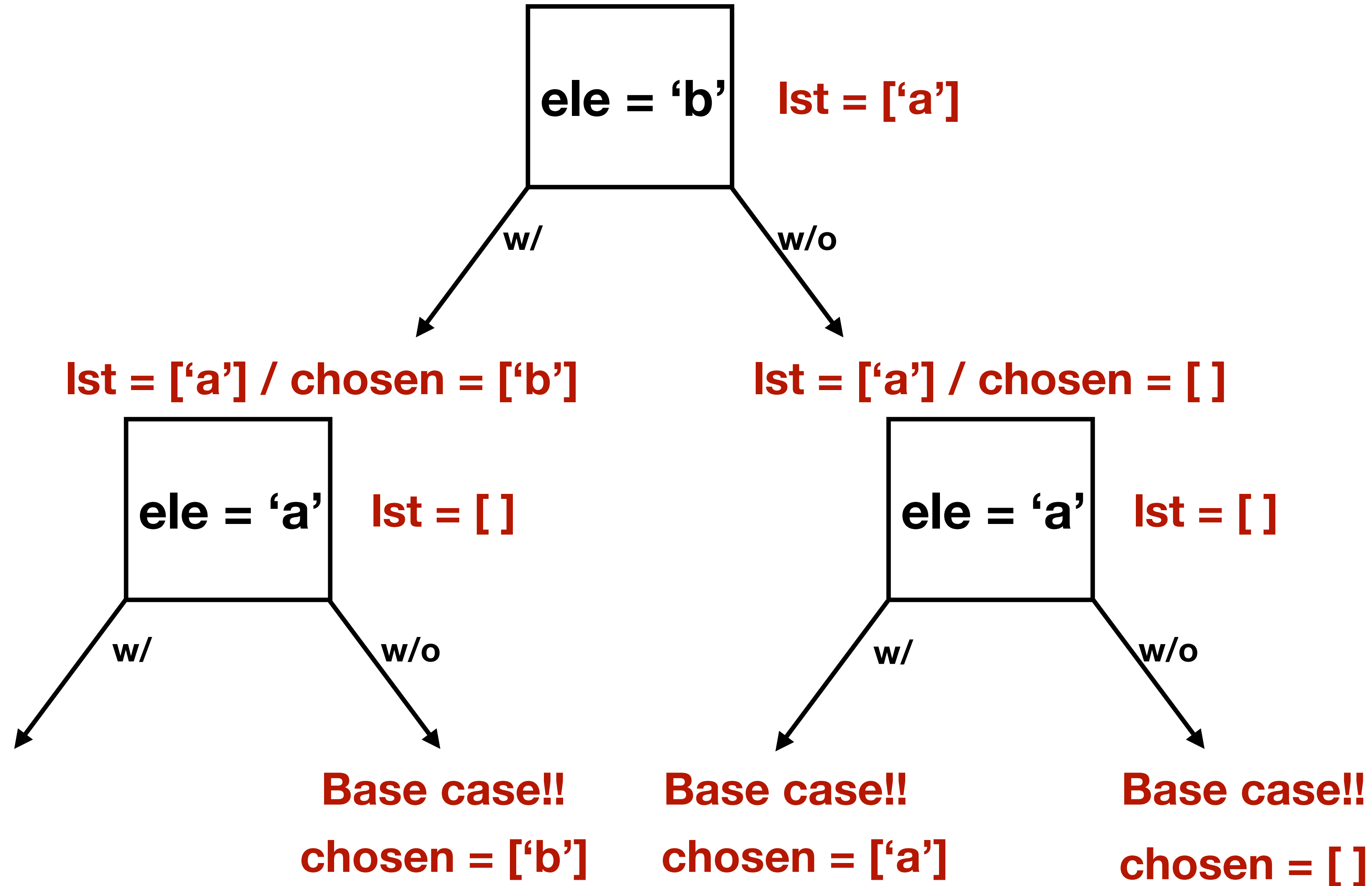
lst = ['a', 'b'] / chosen = []



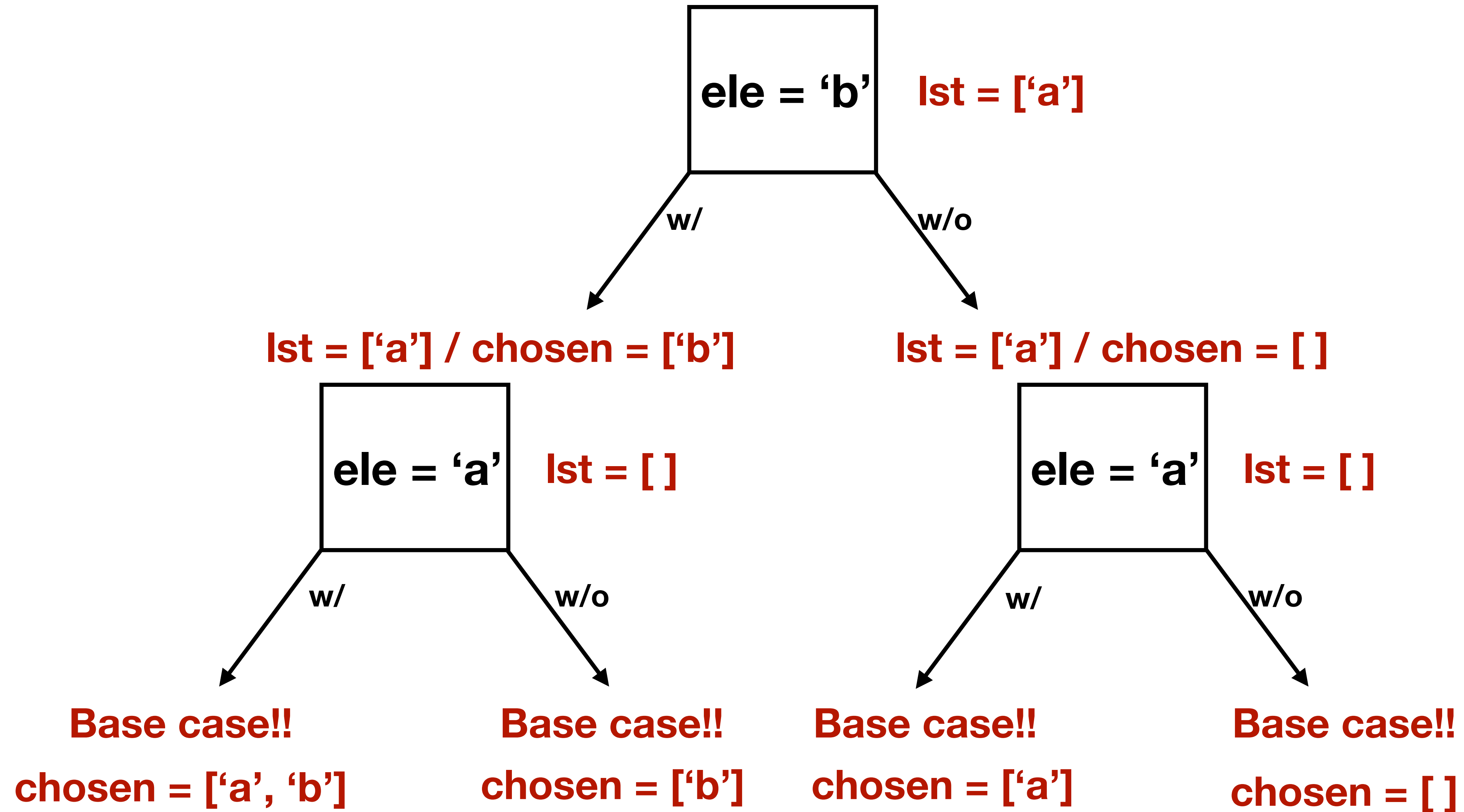
lst = ['a', 'b'] / chosen = []

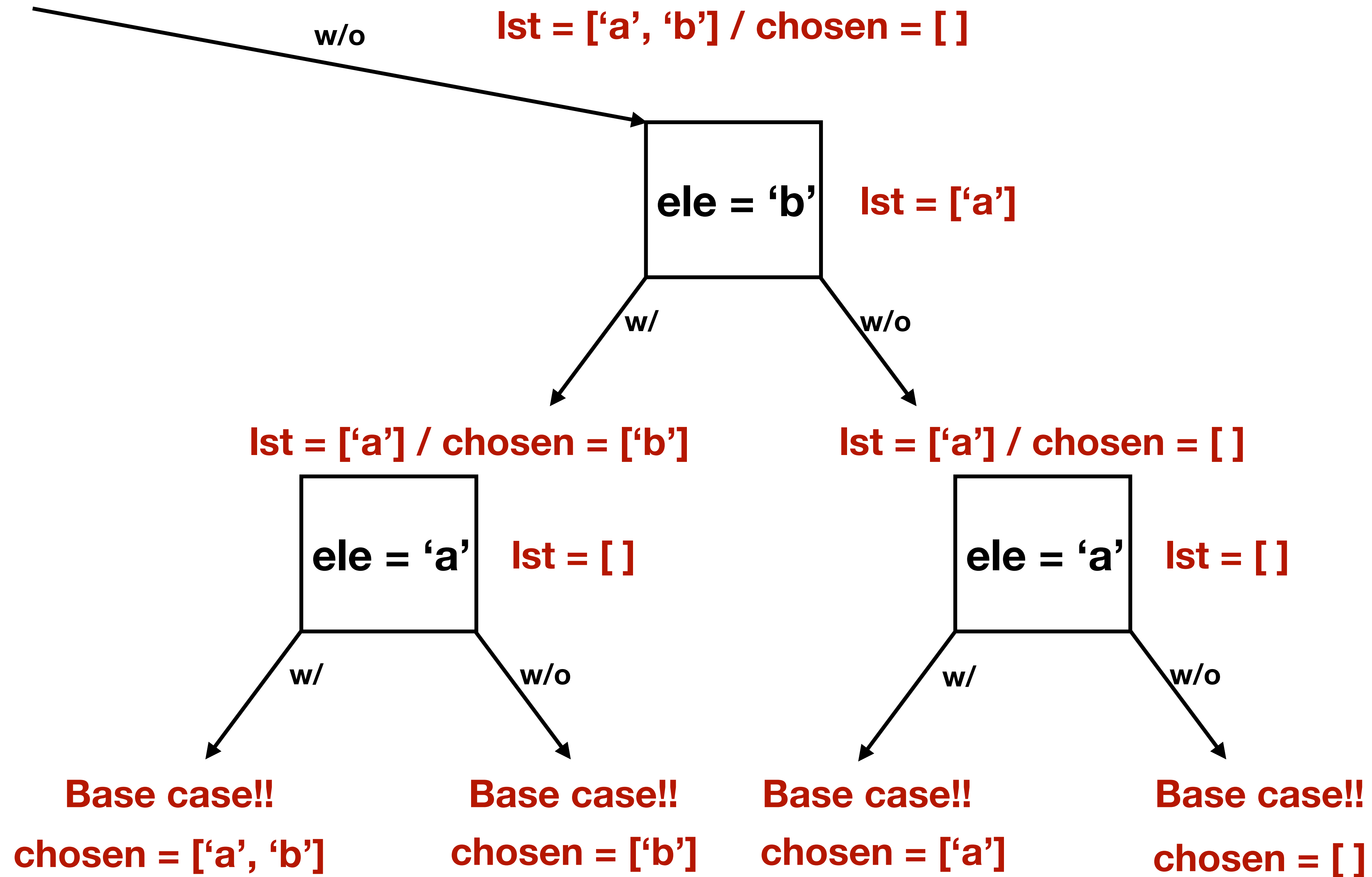


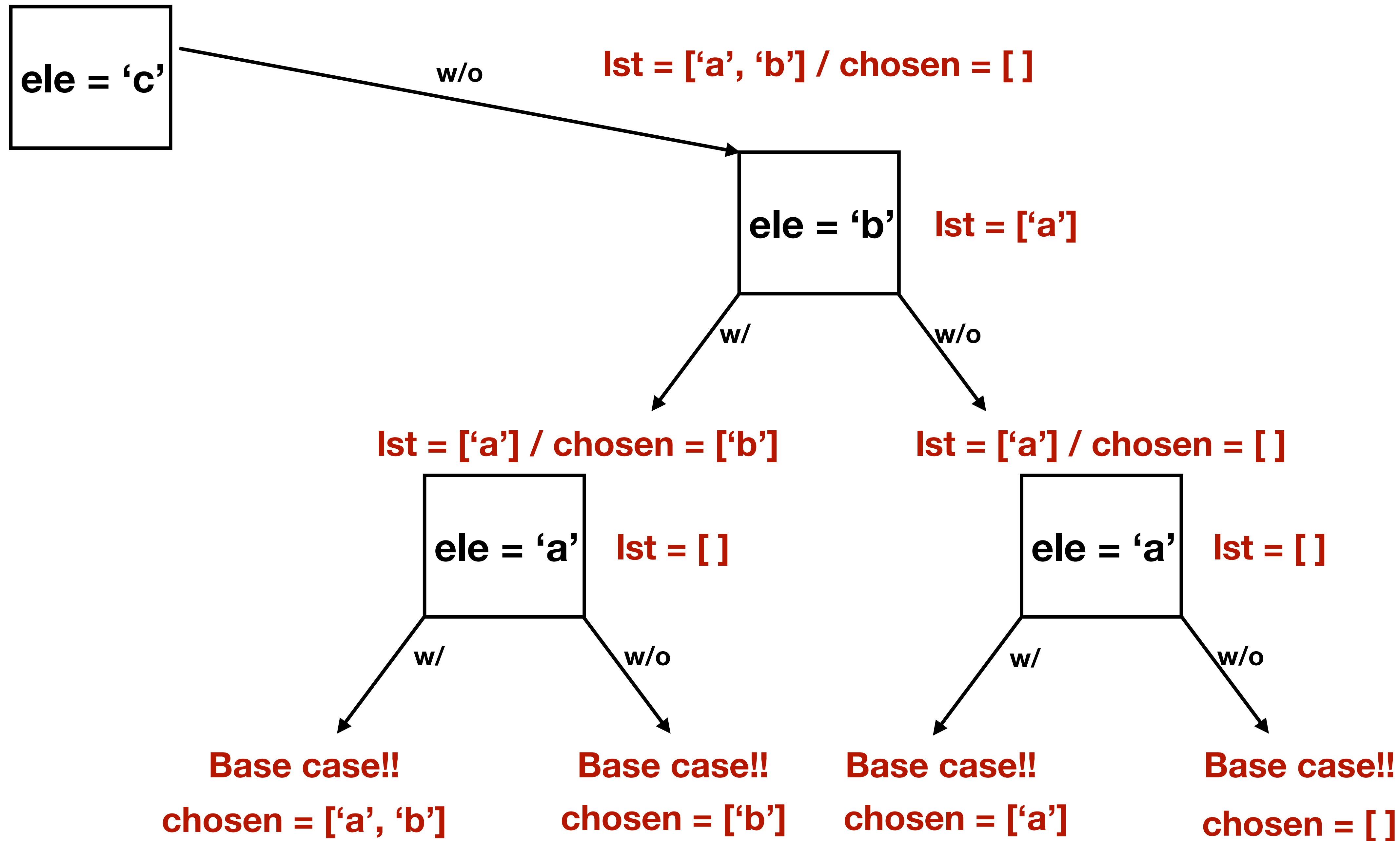
lst = ['a', 'b'] / chosen = []



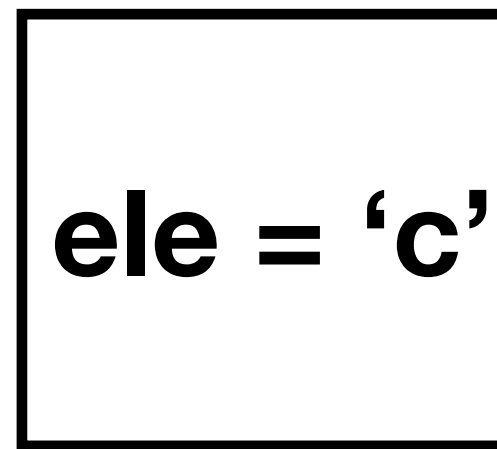
lst = ['a', 'b'] / chosen = []





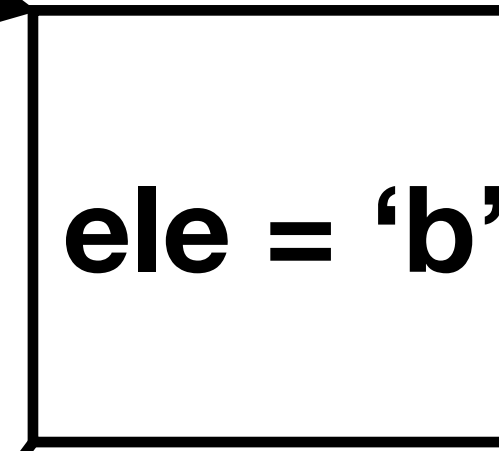


lst = ['a', 'b', 'c'] / chosen = []



w/o

lst = ['a', 'b'] / chosen = []



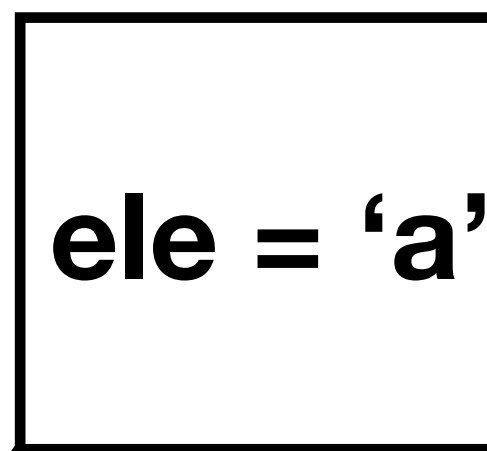
lst = ['a']

w/

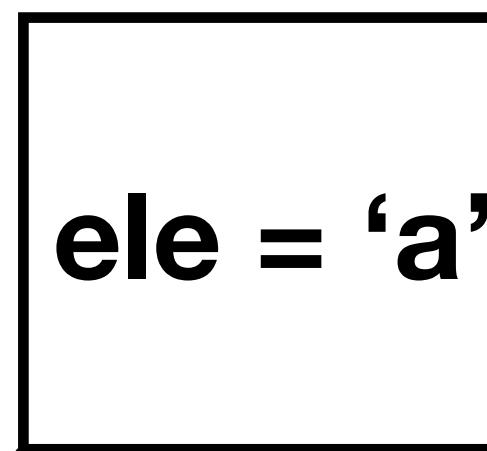
w/o

lst = ['a'] / chosen = ['b']

lst = ['a'] / chosen = []



lst = []



lst = []

w/

w/o

w/

w/o

Base case!!

chosen = ['a', 'b']

Base case!!

chosen = ['b']

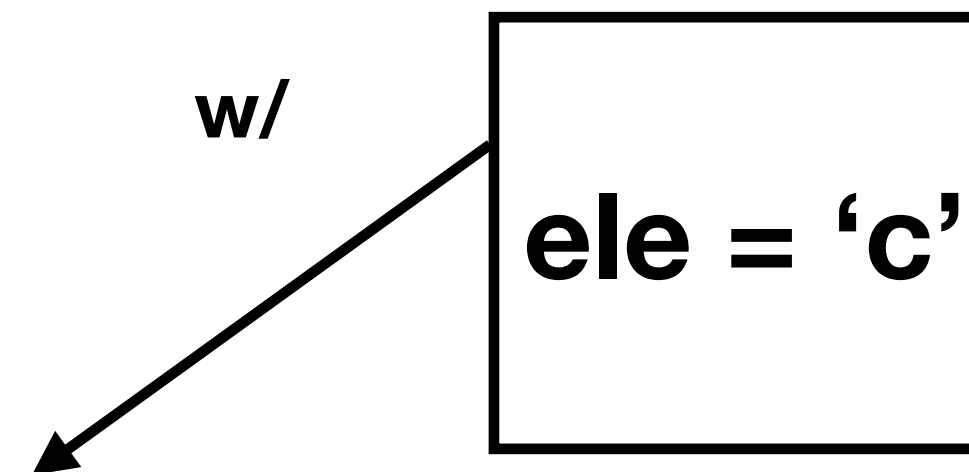
Base case!!

chosen = ['a']

Base case!!

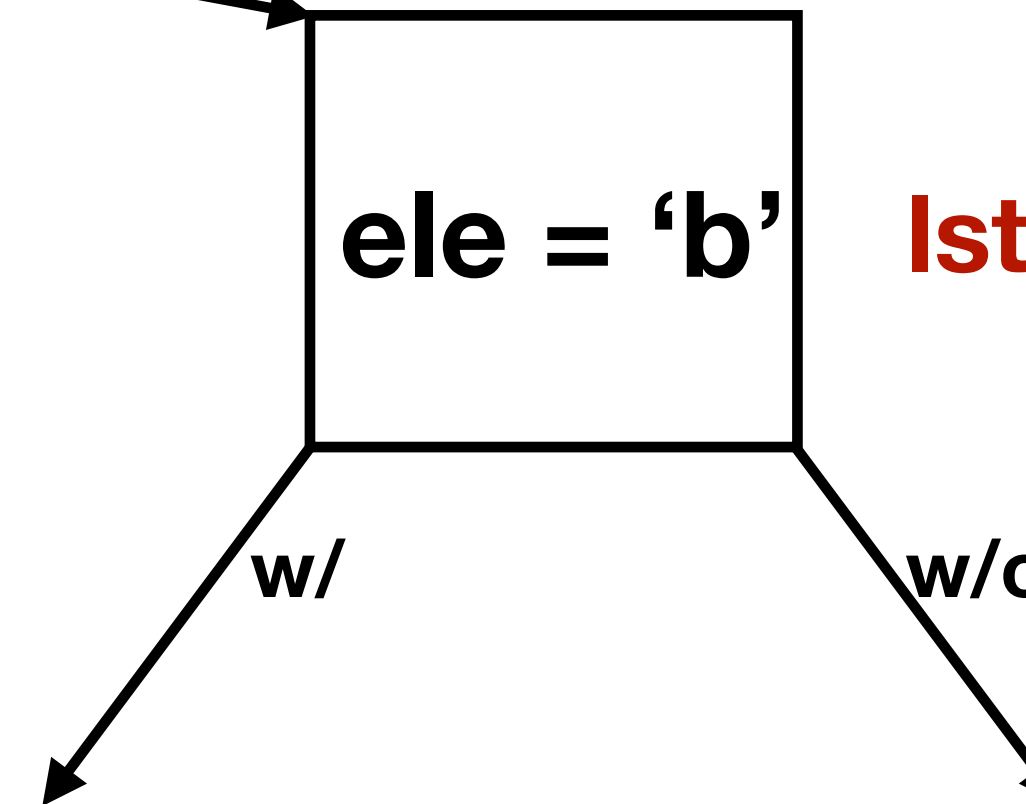
chosen = []

lst = ['a', 'b', 'c'] / chosen = []



w/o

lst = ['a', 'b'] / chosen = []



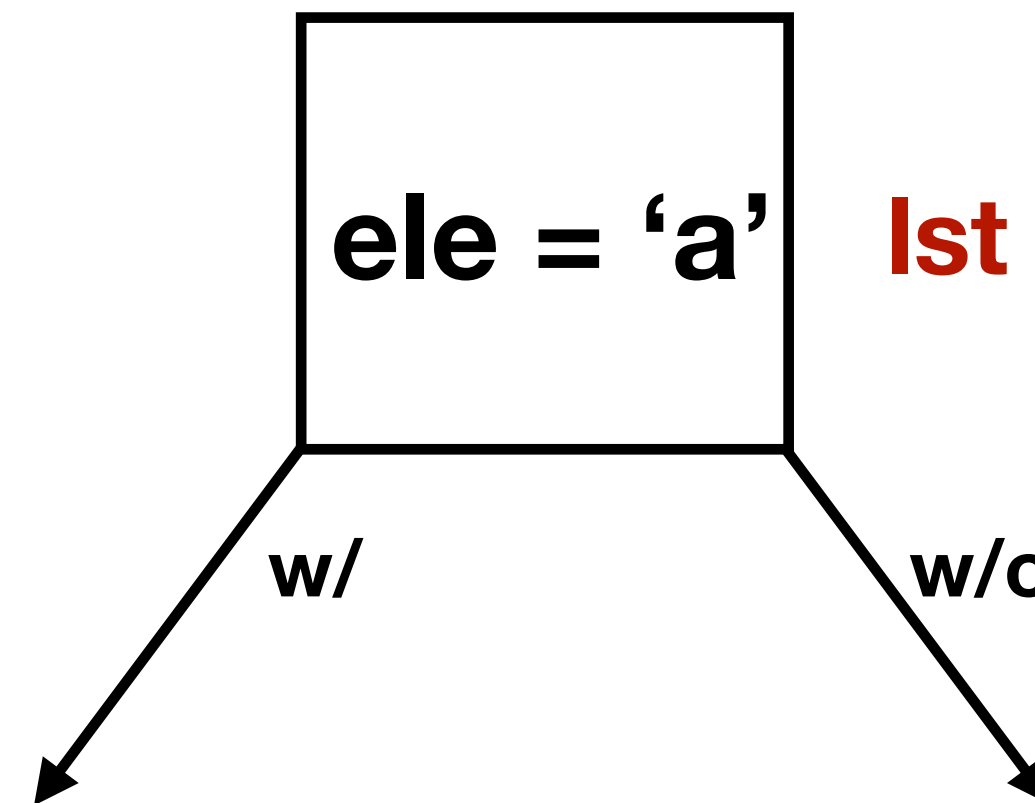
lst = ['a']

w/

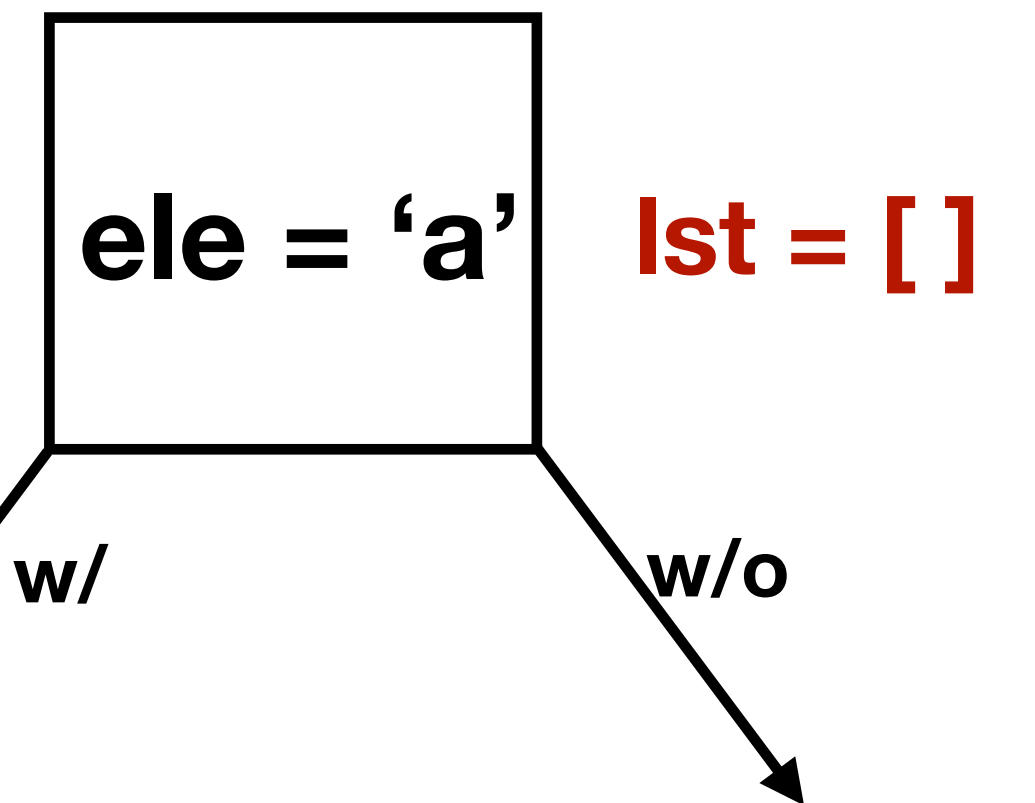
w/o

lst = ['a'] / chosen = ['b']

lst = ['a'] / chosen = []



lst = []



lst = []

Base case!!
chosen = ['a', 'b']

Base case!!
chosen = ['b']

Base case!!
chosen = ['a']

Base case!!
chosen = []

