

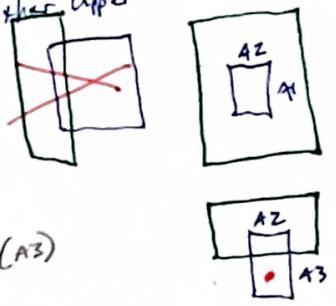
(1)

AAAUV  
AAUVR  
AALV  
AAIUV  
AUVVR  
AUUV  
AHIV  
ALLV  
ALIV  
AIIV  
UHUV  
UULV  
UNIV  
ULLV  
UL-I  
UIII  
L-LLV  
L-LI  
L-II  
III

$BO = \text{Below Other}$   $AO = \text{Above Other}$   $AOL = \text{Along Other Lower}$   
 $AOU = \text{Along Other Upper}$

All All Upper (other, A1, A2, A3)  
 # makes one chunk

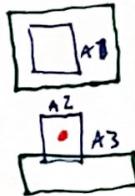
$\text{dict}[A1] = \text{self}[A1]$   
 $\text{dict}[A2] = \text{self}[A2]$   
 $\text{dict}[A3] = (\text{self}[A3][0], \text{other}[A3][0] \rightarrow BO(A3))$   
 yield From Dict(dict)



All All Lower

# makes one chunk

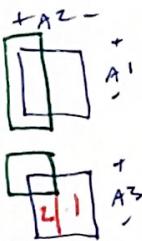
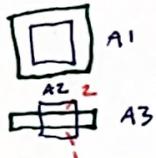
$\text{dict}[A1] = \text{self}[A1]$   
 $\text{dict}[A2] = \text{self}[A2]$   
 $\text{dict}[A3] = (\text{other}[A3][1], \text{self}[A3][1] \rightarrow AO(A3))$   
 yield From Dict(dict)



All All Inside

# makes two chunks

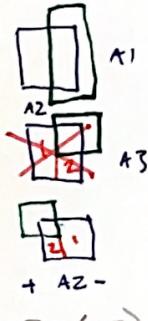
$\text{dict}[A1] = \text{self}[A1]$   
 $\text{dict}[A2] = \text{self}[A2]$   
 $\text{dict}[A3] = (\text{self}[A3][0], \text{other}[A3][0] \rightarrow BO(A3))$   
 yield From Dict(dict)  
 $\text{dict}[A3] = (\text{other}[A3][1], \text{self}[A3][1] \rightarrow AO(A3))$   
 yield From Dict(dict)



All Upper Lower

# makes two chunks

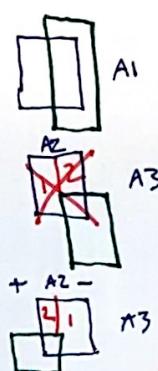
$\text{dict}[A1] = \text{self}[A1]$   
 $\text{dict}[A2] = (\text{self}[A2][0], \text{other}[A2][0] \rightarrow BO(A2))$   
 $\text{dict}[A3] = \text{self}[A3]$   
 yield From Dict(dict)  
 $\text{dict}[A2] = (\text{other}[A2][0], \text{self}[A2][1] \rightarrow AOL(A2))$   
 $\text{dict}[A3] = (\text{self}[A3][0], \text{other}[A3][0] \rightarrow BO(A3))$   
 yield From Dict(dict)



All Upper Lower

# 2 chunks

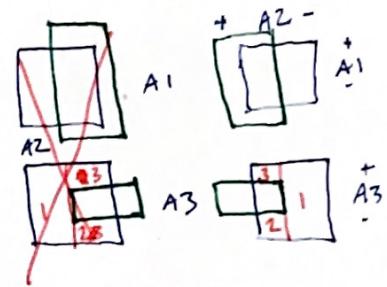
$\text{dict}[A1] = \text{self}[A1]$   
 $\text{dict}[A2] = (\text{self}[A2][0], \text{other}[A2][0] \rightarrow BO(A2))$   
 $\text{dict}[A3] = (\text{other}[A3][1], \text{self}[A3][1] \rightarrow self[A3])$   
 yield From Dict(dict)  
 $\text{dict}[A2] = (\text{other}[A2][0], \text{self}[A2][1] \rightarrow AOL(A2))$   
 $\text{dict}[A3] = (\text{other}[A3][1], \text{self}[A3][1] \rightarrow AO(A3))$   
 yield From Dict(dict)



(2)

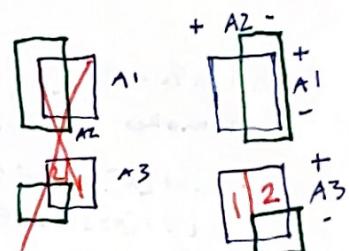
## All Upper Inner

#3 chunks

 $\text{dict}[A1] = \text{self}[4]$  $\text{dict}[A2] = (\text{self}[A2][0], \text{other}[A2][0]) \rightarrow \text{AO}(A2)$  $\text{dict}[A3] = \text{self}[A3]$ ①  $\text{yield From Dict}(\text{dict})$  $\text{dict}[A2] = (\text{other}[A2][0], \text{self}[A2][1]) \rightarrow \text{AO}(A2)$  $\text{dict}[A3] = (\text{self}[A3][0], \text{other}[A3][0]) \rightarrow \text{BO}(A3)$ ②  $\text{yield From Dict}(\text{dict})$  $\text{dict}[A3] = (\text{other}[A3][1], \text{self}[A3][1]) \rightarrow \text{AO}(A3)$ ③  $\text{yield From Dict}(\text{dict})$ 

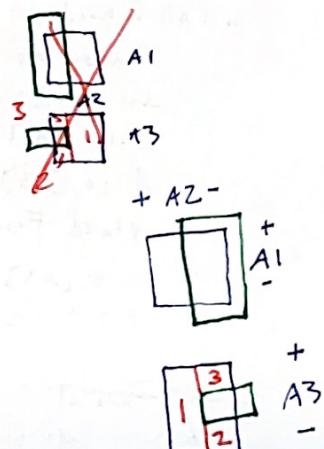
## All Lower Lower

#2 chunks

 $\text{dict}[A1] = \text{self}[A1]$  $\text{dict}[A2] = (\text{other}[A2][1], \text{self}[A2][1]) \rightarrow \text{AO}(A2)$  $\text{dict}[A3] = \text{self}[A3]$ ①  $\text{yield From Dict}(\text{dict})$  $\text{dict}[A2] = (\text{self}[A2][0], \text{other}[A2][1]) \rightarrow \text{AO}(A2)$  $\text{dict}[A3] = (\text{other}[A3][1], \text{self}[A3][1]) \rightarrow \text{AO}(A3)$ ②  $\text{yield From Dict}$ 

## All Lower Inner

#3 chunks

 $\text{dict}[A1] = \text{self}[A1]$  $\text{dict}[A2] = (\text{other}[A2][1], \text{self}[A2][1]) \rightarrow \text{AO}(A2)$  $\text{dict}[A3] = \text{self}[A3]$ ①  $\text{yield From Dict}(\text{dict})$  $\text{dict}[A2] = (\text{self}[A2][0], \text{other}[A2][1]) \rightarrow \text{AO}(A2)$  $\text{dict}[A3] = (\text{self}[A3][0], \text{other}[A3][0]) \rightarrow \text{BO}(A3)$ ②  $\text{yield From Dict}(\text{dict})$  $\text{dict}[A2] = (\text{other}[A2][1], \text{self}[A2][1]) \rightarrow \text{AO}(A2)$ ③  $\text{yield From Dict}(\text{dict})$ 

(3)

All Inner Inner

# 4 chunks

$$\text{dict}[A1] = \text{self}[A1]$$

$$\text{dict}[A2] = (\text{self}[A2][0], \text{other}[A2][1]) \rightarrow \text{BO}(A2)$$

$$\text{dict}[A3] = \text{self}[A3]$$

① yield from Dict(dict)

$$\text{dict}[A2] = (\text{other}[A2][0]+1, \text{self}[A2][1]) \rightarrow \text{AO}(A2)$$

② yield from Dict(dict)

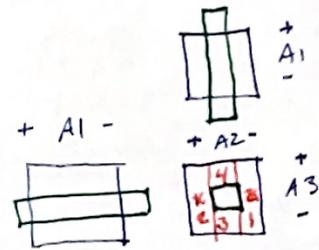
$$\text{dict}[A2] = (\text{other}[A2][0], \text{other}[A2][1]) \rightarrow \text{other}(A2)$$

$$\text{dict}[A3] = (\text{self}[A3][0], \text{other}[A3][1]) \rightarrow \text{BO}(A3)$$

③ yield from Dict(dict)

$$\text{dict}[A3] = (\text{other}[A3][0]+1, \text{self}[A3][1]) \rightarrow \text{AO}(A3)$$

④ yield from Dict(dict)



Upper Upper

# 3 chunks

$$\text{dict}[A1] = \text{self}[A1]$$

$$\text{dict}[A2] = \text{self}[A2]$$

$$\text{dict}[A3] = (\text{self}[A3][0], \text{other}[A3][1]) \rightarrow \text{BO}(A3)$$

① yield from Dict(dict)

$$\text{dict}[A2] = (\text{self}[A2][0], \text{other}[A2][1]) \rightarrow \text{BO}(A2)$$

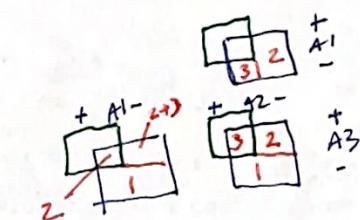
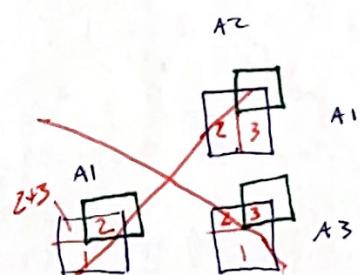
$$\text{dict}[A3] = (\text{other}[A3][0], \text{self}[A3][1]) \rightarrow \text{AOU}(A3)$$

② yield from Dict(dict)

$$\text{dict}[A2] = (\text{self}[A2][0], \text{other}[A2][1]) \rightarrow \text{BO}(A2)$$

$$\text{dict}[A3] = (\text{other}[A3][0], \text{self}[A3][1]) \rightarrow \text{AOU}(A3)$$

③ yield from Dict(dict)



Upper Upper Lower

# 3 chunks

$$\text{dict}[A1] = \text{self}[A1]$$

$$\text{dict}[A2] = \text{self}[A2]$$

$$\text{dict}[A3] = (\text{other}[A3][0]+1, \text{self}[A3][0]) \rightarrow \text{AO}(A3)$$

① yield from Dict(dict)

$$\text{dict}[A2] = (\text{self}[A2][0], \text{other}[A2][1]) \rightarrow \text{BO}(A2)$$

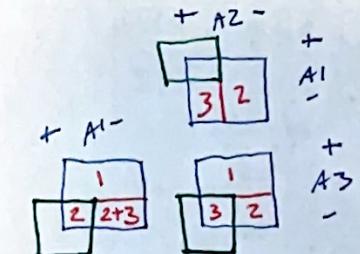
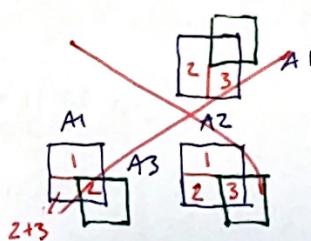
$$\text{dict}[A3] = (\text{self}[A3][0], \text{other}[A3][1]) \rightarrow \text{AOU}(A3)$$

② yield from Dict(dict)

$$\text{dict}[A1] = (\text{self}[A1][0], \text{other}[A1][1]) \rightarrow \text{BO}(A1)$$

$$\text{dict}[A2] = (\text{other}[A2][0], \text{self}[A2][1]) \rightarrow \text{AOU}(A2)$$

③ yield from Dict(dict)



(4)

Upper Lower  
Upper Inner

# 3 chunks

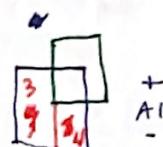
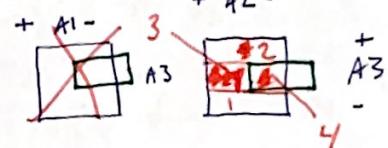
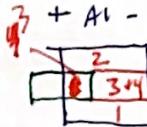
4

$\text{dict}[A1] = \text{self}[A1]$

$\text{dict}[A2] = \text{self}[A2]$

$\text{dict}[A3] = (\text{self}[A3][0], \text{other}[A3][0] \rightarrow B0(A3))$

①  $\text{yield From Dict}(dict)$



$\text{dict}[A3] = (\text{other}[A3][1] \rightarrow, \text{self}[A3][1]) AO(A3)$

②  $\text{yield From Dict}(dict)$

$\text{dict}[A2] = (\text{other}[A2][1] \rightarrow, \text{self}[A2][1]) AO(A2)$

$\text{dict}[A3] = \text{other}[A3]$

③  $\text{yield From Dict}(dict)$

$\text{dict}[A1][1] = \text{other}[A1][0] \rightarrow B0(A1)$

$\text{dict}[A2] = (\text{self}[A2][0], \text{other}[A2][1]) AO(A2)$

④  ~~$\text{yield From Dict}(dict)$~~

Upper Upper Inner

# 4 chunks

$\text{dict}[A1] = \text{self}[A1]$

$\text{dict}[A2] = \text{self}[A2]$

$\text{dict}[A3] = (\text{self}[A3][0], \text{other}[A3][0] \rightarrow B0(A3))$

①  $\text{yield From Dict}(dict)$

$\text{dict}[A3] = (\text{other}[A3][1] \rightarrow, \text{self}[A3][1]) AO(A3)$

②  $\text{yield From Dict}(dict)$

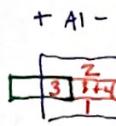
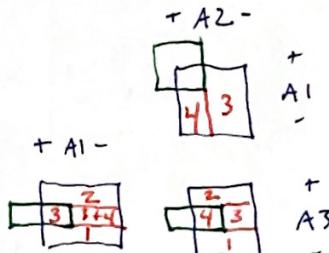
$\text{dict}[A2][1] = \text{other}[A2][0] \rightarrow B0(A2)$

③  $\text{yield From Dict}(dict) \quad \text{dict}[A3] = \text{other}[A3]$

$\text{dict}[A1][1] = \text{other}[A1][0] \rightarrow B0(A1)$

$\text{dict}[A2] = (\text{other}[A2][0], \text{self}[A2][1]) AO(A2)$

④  $\text{yield From Dict}(dict)$



(5)

Upper Lower Lower  
#3 Chunks

dict[A1] = self[A1]

dict[A2] = self[A2]

dict[A3] = (other[A3][1], self[A3][1]) AOL(A3)

① yield FromDict(dict)

dict[A2] = (other[A2][1], self[A2][1]) AOL(A2)

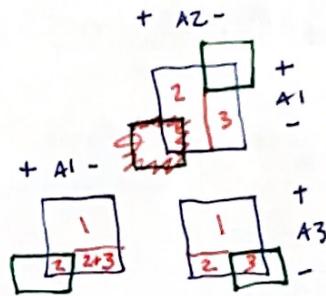
dict[A3] = (self[A3][0], other[A3][1]) AOL(A3)

② yield from Dict(dict)

dict[A1][1] = other[A1][0] or AOL(A1) BO(A1)

dict[A2] = (other[A2][1], self[A2][1]) AOL(A2)

③ yield FromDict(dict)



Upper Inner Inner

#5 Chunks

dict[A1] = self[A1]

dict[A2] = self[A2]

dict[A3] = (self[A3][0], other[1][0]) BO(A3)

① yield FromDict(dict)

dict[A3] = (other[A3][1], self[A3][1]) AOL(A3)

② yield FromDict(dict)

dict[A1][1] = other[A1][0] or AOL(A1) BO(A1)

dict[A3] = other[A3]

③ yield FromDict(dict)

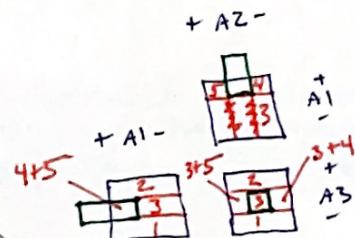
dict[A1] = (other[A1][0]) AOL(A1)

dict[A2] = BO(A2)

④ yield FromDict(dict)

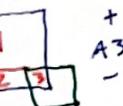
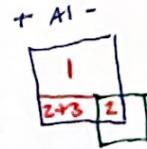
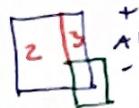
dict[A2] = AOL(A2)

⑤ yield FromDict(dict)



(6)

+ A2 -



def LowerLowerLower

#3 chunks

dict[A1] = self[A1]

dict[A2] = self[A2]

dict[A3] = AO(A3)

① yield from dict(A3)

dict[A2] = AO(A2)

dict[A3] = AOL(A3)

② yield from dict(~~dict~~)

dict[A1] = AO(A1)

dict[A2] = AOL(A2)

③ yield from dict(dict)

def LowerLowerInner

dict[A1] = self[A1]

dict[A2] = self[A2]

dict[A3] = TSO(A3)

① yield from dict(dict)

dict[A3] = AO(A3)

② yield from dict(dict)

dict[A2] = AD(A2)

dict[A3] = other[A3]

③ yield from dict(dict)

dict[A1] = AO(A1)

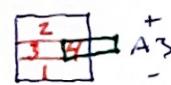
dict[A2] = AOL(A2)

④ yield from dict(dict)

+ A2 -



+ A1 -



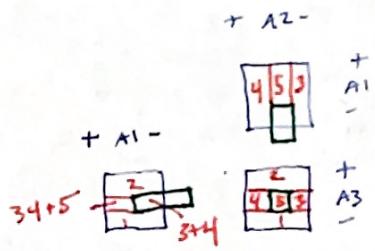
+ A3 -

(7)

```

def LowerLowerInner
# 5 chunks
dict[A1] = self[A1]
dict[A2] = self[A2]
dict[A3] = BO(A3)
① yield From Dict(dict)
dict[A3] = AO(A3)
② yield From Dict(dict)
dict[A1] = other[A1]
dict[A2] = BO(A2)
dict[A3] = other[A3]
③ yield From Dict(dict)
dict[A2] = AO(A2)
④ yield From Dict(dict)
dict[A1] = other[A1] AO(A1)
dict[A2] = other[A2]
⑤ yield From Dict(dict)

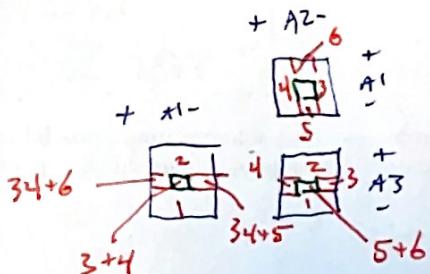
```



```

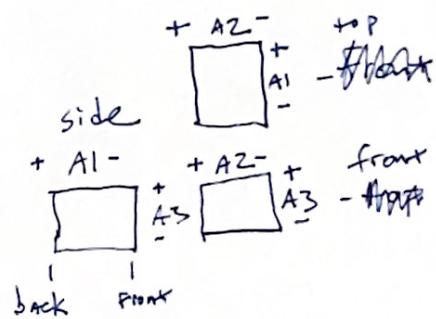
def InnerInnerInner
# 6 chunks
dict[A1] = self[A1]
dict[A2] = self[A2]
dict[A3] = BO(A3)
① yield From Dict(dict)
dict[A3] = AO(A3)
② yield From Dict(dict)
dict[A2] = BO(A2)
dict[A3] = other[A3]
③ yield From Dict(dict)
dict[A2] = AO(A2) dict[A3] = AO(A3)
④ yield From Dict(dict)
dict[A1] = BO(A1)
dict[A2] = other[A2]
⑤ yield From Dict(dict)
dict[A1] = AO(A1)
⑥ yield From Dict(dict)

```



if n.on: add n to itemsToAdd  
 for m in existing:  
     # Add m to itemsToAdd  
     if m.intersection(n):  
         # Add m to itemsToAdd  
         # find each Axis' intersection types  
         # AAA - no action. no part of m is retained

See page 8



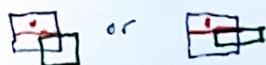
def AlongOtherUpper(self, other: Cuboid, axis: str) → Tuple[int, int]:  
     # return the area alongside the other's  
     # Area that intersects the upper  
     # part of this axis  
     return (other[axis][0], self[axis][1])



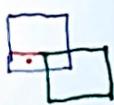
def BelowOther(self, other, axis: str) → Tuple[int, int]:  
     # return the area below the other's  
     # Area on this axis.  
     return (other[axis][0], other[axis][0] - 1)



def AboveOther(self, other: Cuboid, axis: str) → Tuple[int, int]:  
     # return the area above the other's  
     # Area on this axis.  
     return (other[axis][1] + 1, self[axis][1])



def AlongOtherLower  
     # return the area alongside the other's  
     # Area that intersects the lower  
     # part of this axis.  
     return (self[axis][0], other[axis][1])



⑧

Make a dictionary with keys = all combinations of intersection types and values the method to handle that type

~~for~~

For each new cube  $n$

For each existing cube  $e$

make tuple containing each axis' intersection type (none, all, upper, lower, inner)

If any is none, continue

make a tuple with sorted intersection types and another with their corresponding axis. eg if  $x$  is inner,  $y$  is upper, and  $z$  is all, tuples would be (all, upper, inner) and ('z', 'y', 'x')

call the method based on the intersection combination

Add each result of method to itemsToAdd

Add  $e$  to itemsToRemove

if  $n$  method is "on" Add  $n$  to itemsToRemove

Add itemsToAdd to set of cubes

remove itemsToRemove from set of cubes