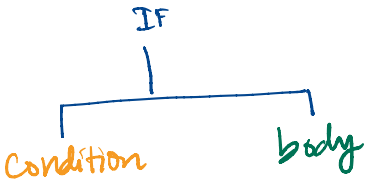
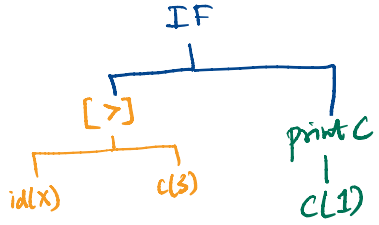
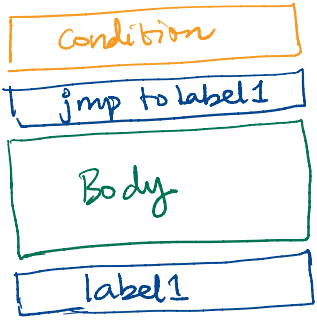
IF:



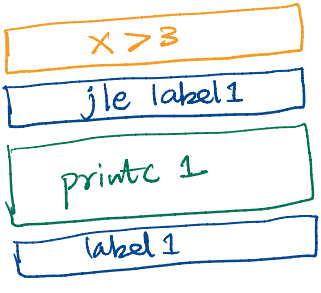
Eg: if (x > 3){
printc 1;
}



ASSEMBLY BLOCK:

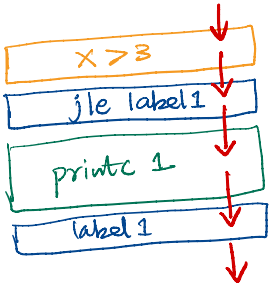


* Note: Think about the condition for the first jump!
hint: 'jne for '=='

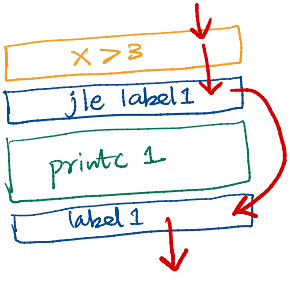


EXAMPLE EXECUTIONS:

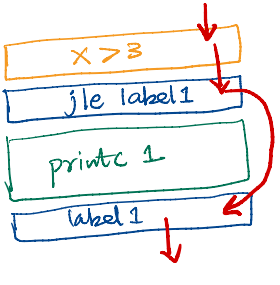
x = 5



x = 3

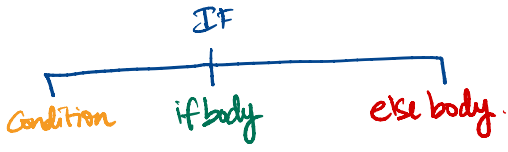


x = 1

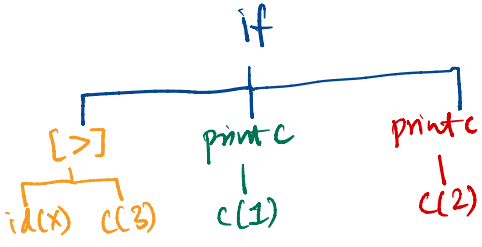


Note: x>3 will be implemented with cmpg instructions.

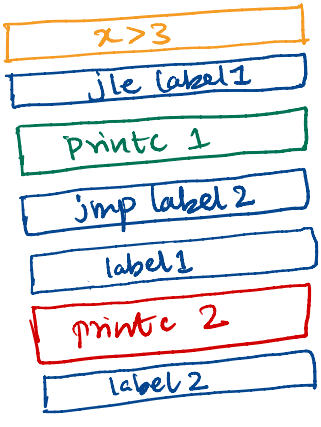
IF - ELSE



Eg:
```
if (x>3) {
    printc 1;
}
else {
    print 2;
}
```



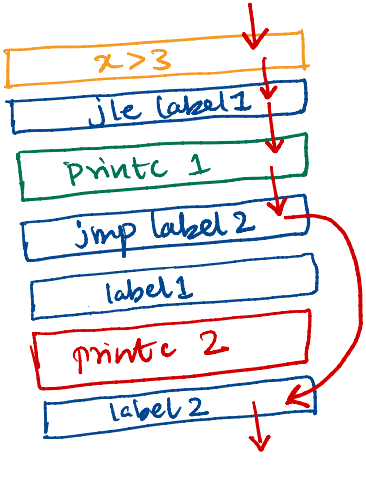
ASSEMBLY BLOCK:



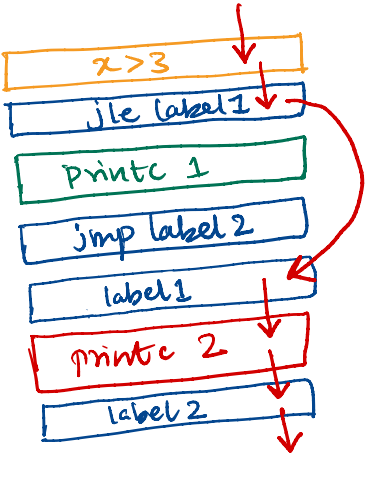
* Note: Think about the condition for the first jump!
Hint: 'jne for '=='

EXAMPLE EXECUTION FLOW:

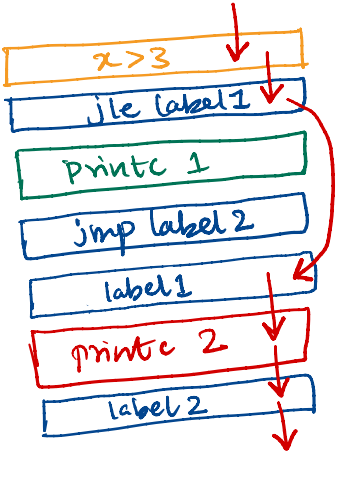
x = 5



x = 3

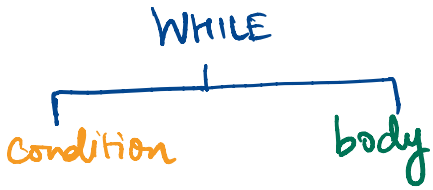


x = 1

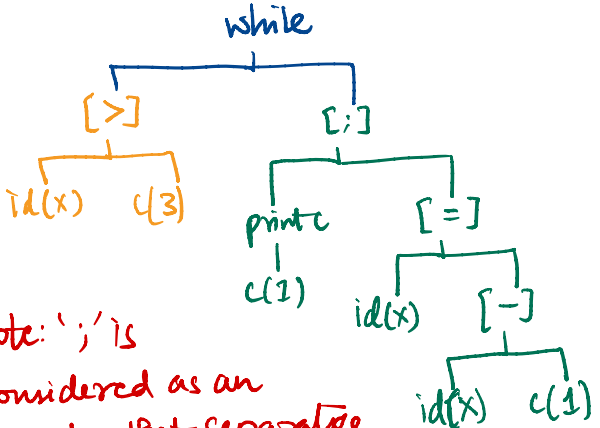


Note: x>3 will be implemented with cmpq instructions.

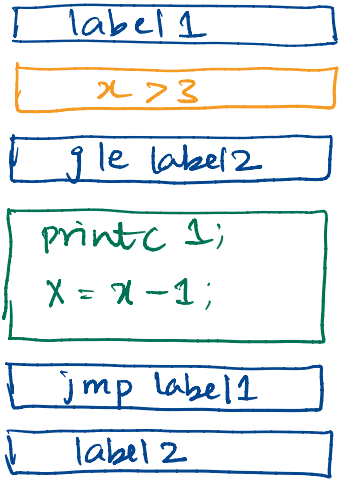
WHILE:



Eg:
```
while (x>3) {
    printc 1;
    x = x - 1;
};
```



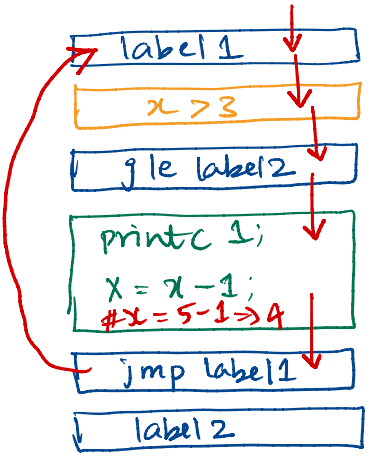
ASSEMBLY BLOCK



Note: ';' is considered as an operator that separates 2 statements. It is implemented already.

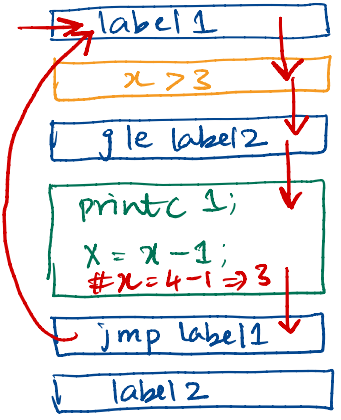
EXAMPLE EXECUTION:

Starting X=5



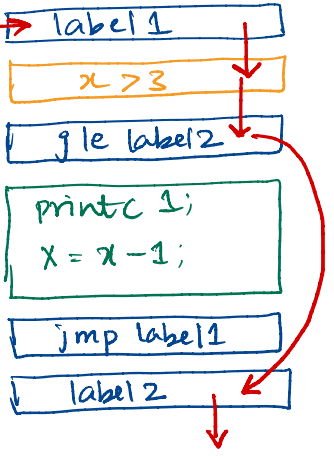
in the loop

when X = 4



in the loop

when X = 3



exits loop.

Notes from IF & IF-ELSE implementations apply here as well.