

LØKKER

INF100

VÅR 2025

Torstein Strømme

I filen *joker.py*, skriv et program som ber brukeren om 5 tall. Disse representerer grunntallene vi får oppgitt når vi begynner å spille Joker. Deretter skal programmet skrive ut enten "opp" eller "ned", for hvert av de fem grunntallene. En kjøring av programmet kan se slik ut:

```
tall1 = 3  
tall2 = 4  
tall3 = 5  
tall4 = 6  
tall5 = 1  
opp  
opp  
ned  
ned  
opp
```

iterand. variabel som viser til et nytt element i den iterable for hver iterasjon av løkken

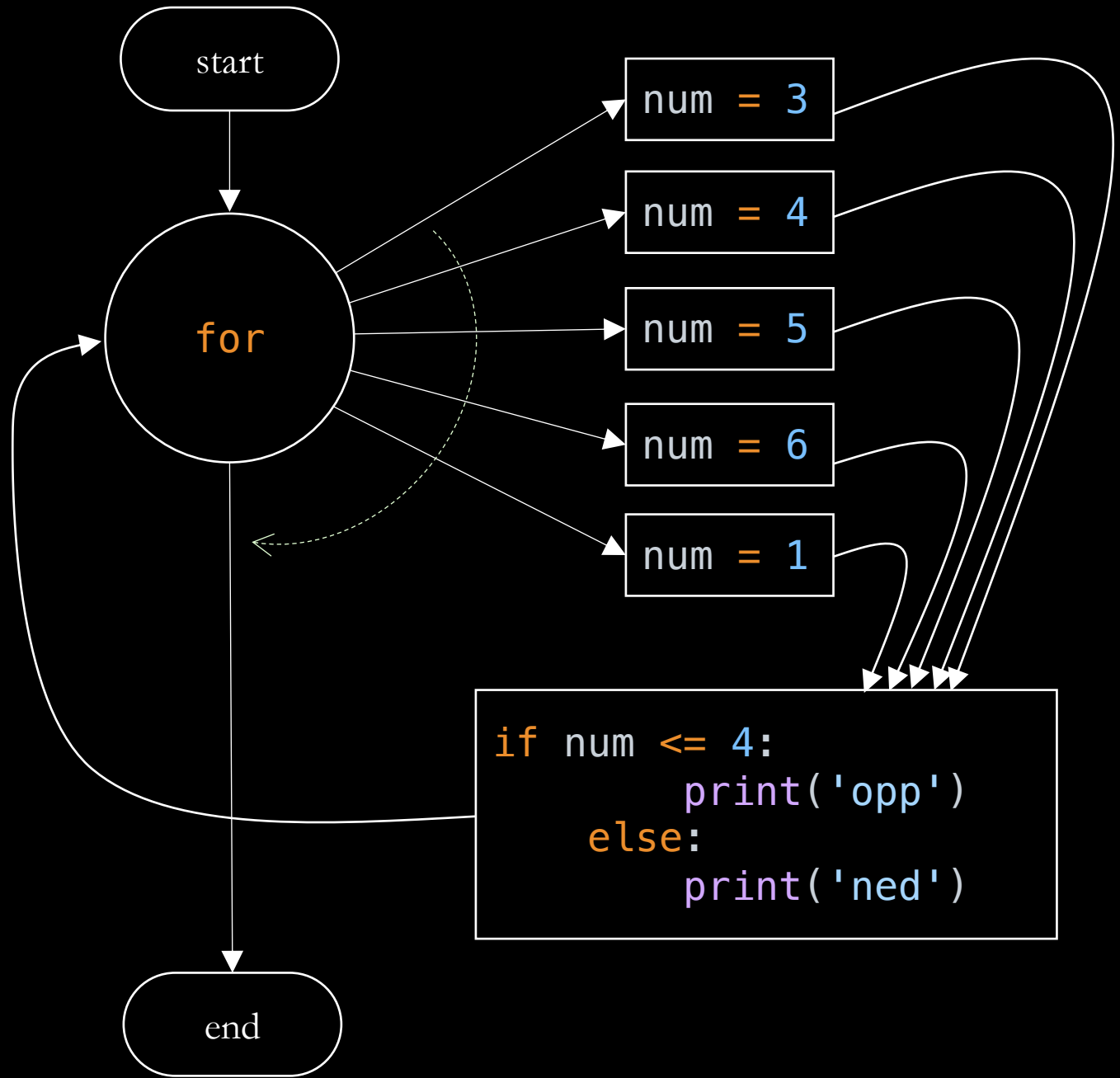
iterabel. en samling av elementer (f. eks. en liste) som skal itereres over

```
for num in [3, 4, 5, 6, 1]:  
    if num <= 4:  
        print('opp')  
    else:  
        print('ned')
```

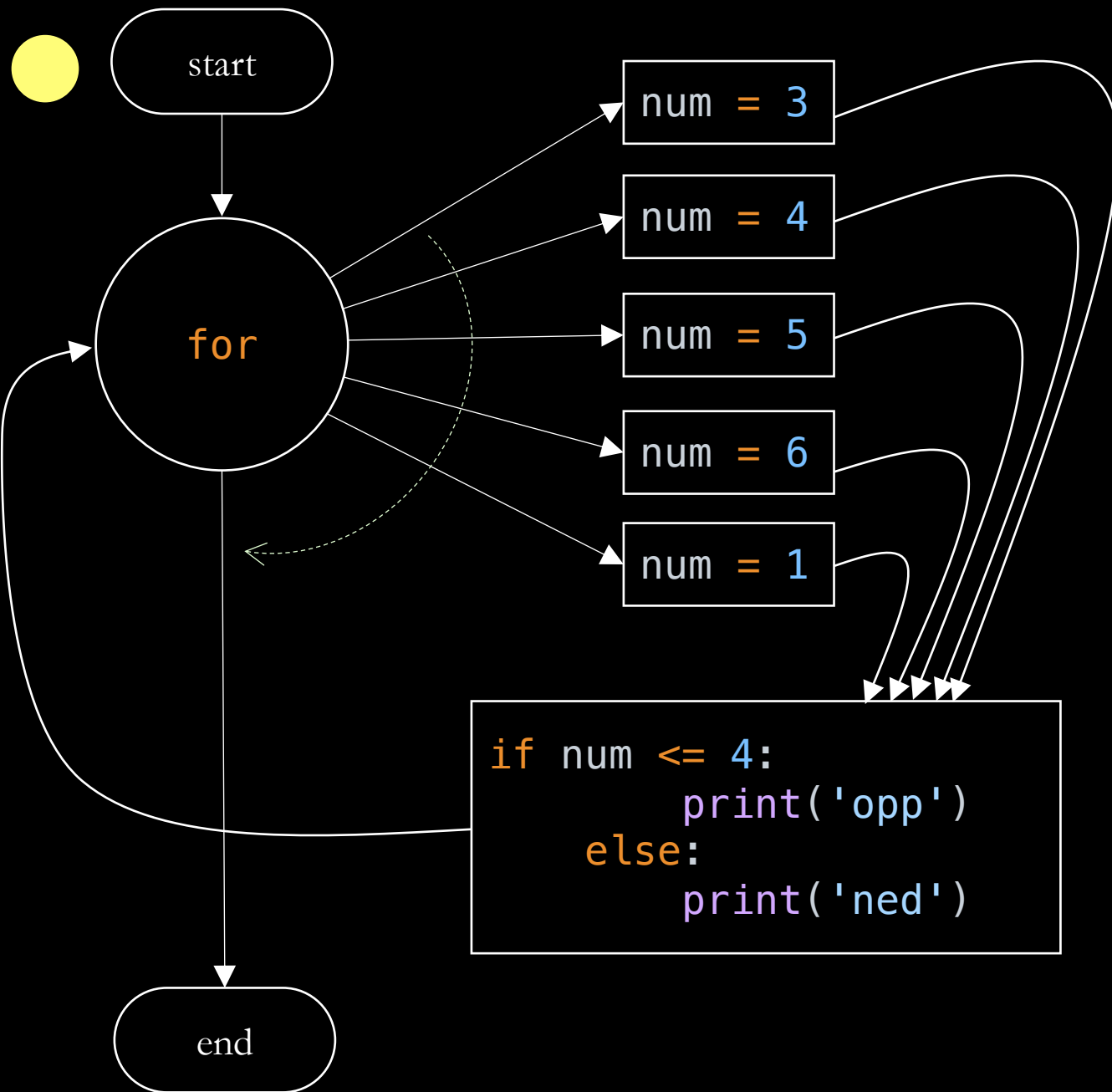
for-løkke. en setning med en løkke kropp som utføres én gang for hvert element i en iterabel.

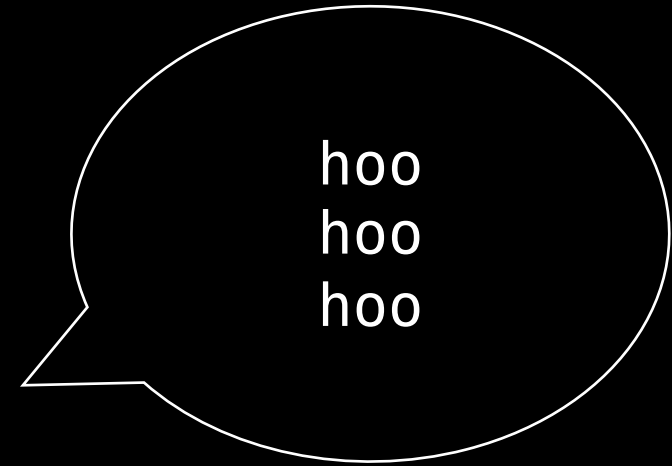
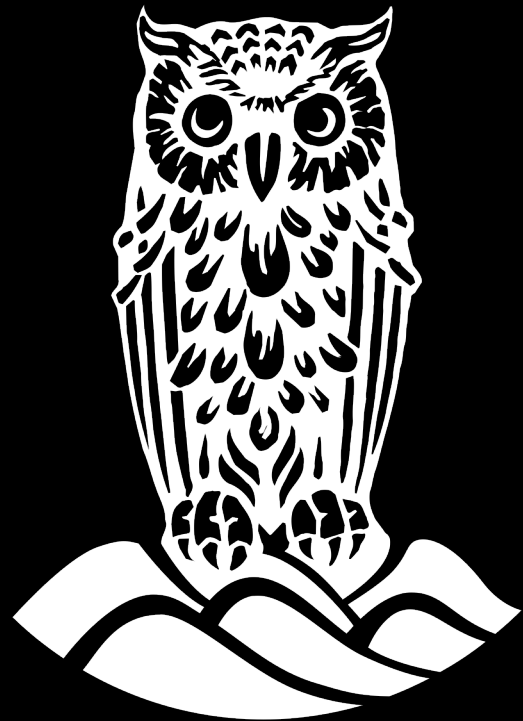
løkke kropp. innrykket blokk av setninger som utføres én gang for hvert element i iterabelen.

```
for num in [3, 4, 5, 6, 1]:  
    if num <= 4:  
        print('opp')  
    else:  
        print('ned')
```



```
for num in [3, 4, 5, 6, 1]:
    if num <= 4:
        print('opp')
    else:
        print('ned')
```





```
print('hoo')  
print('hoo')  
print('hoo')
```

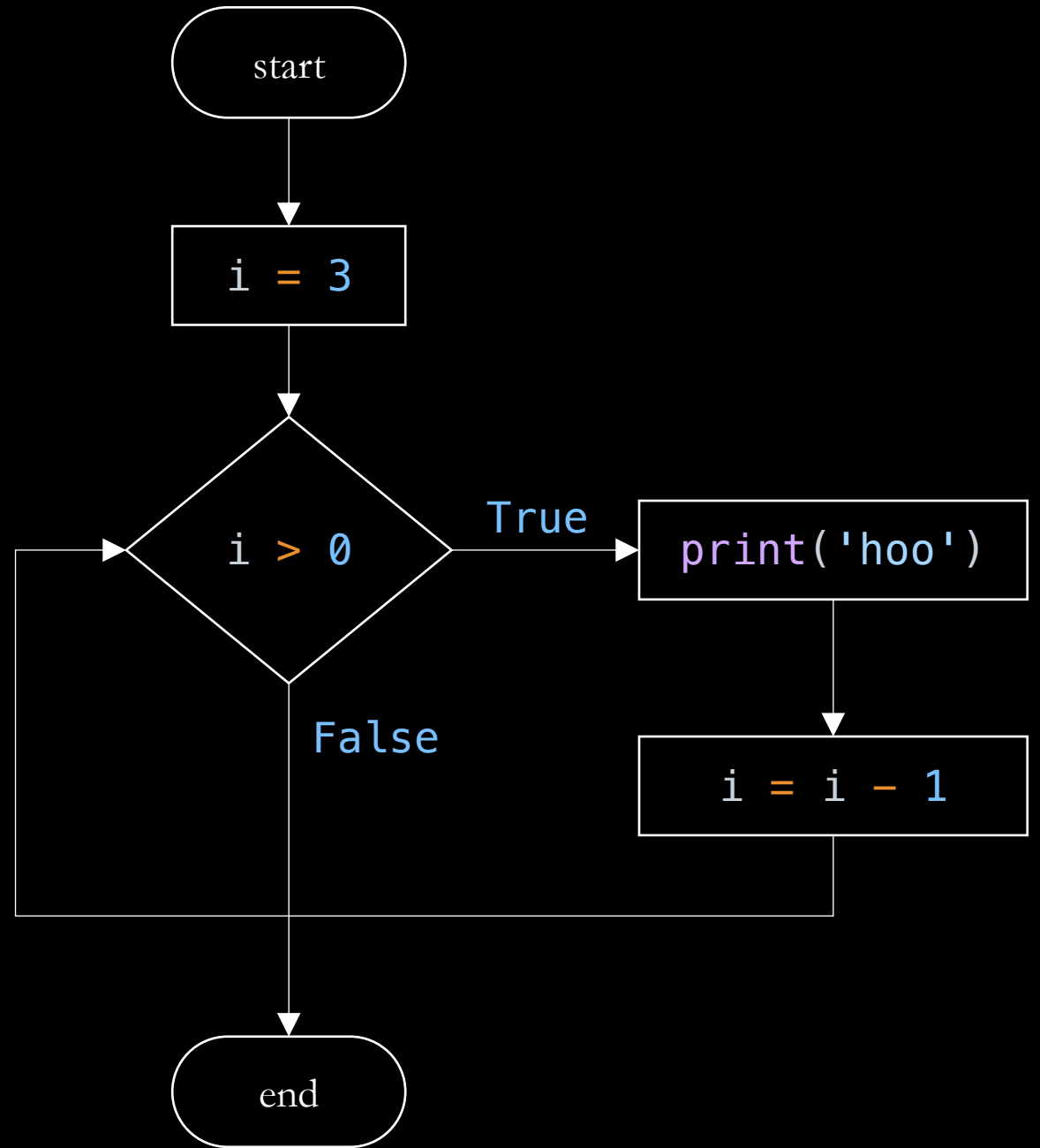
betingelse. et uttrykk som evaluerer til en boolsk verdi (True eller False)

```
i = 3
while i > 0:
    print('hoo')
    i = i - 1
```

} while-setning

løkke kropp. innrykket blokk av setninger som utføres så lenge betingelsen er True.


```
i = 3
while i > 0:
    print('hoo')
    i = i - 1
```

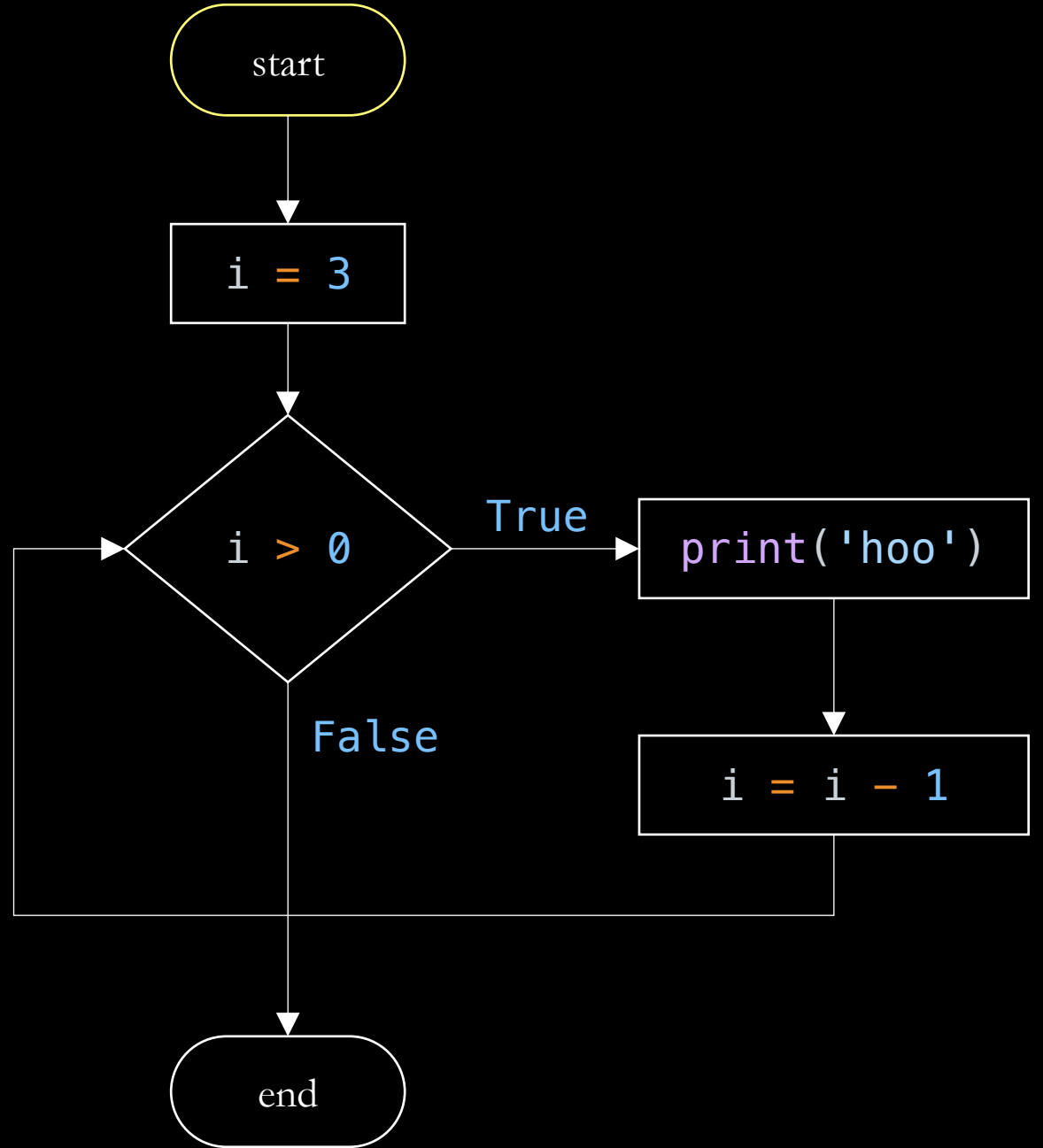
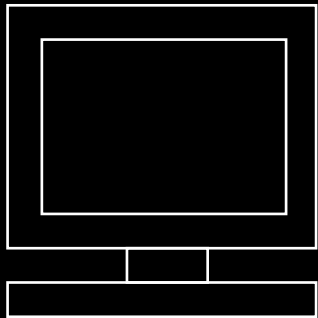




```
i = 3
while i > 0:
    print('hoo')
    i = i - 1
```

Variabler

Objekter

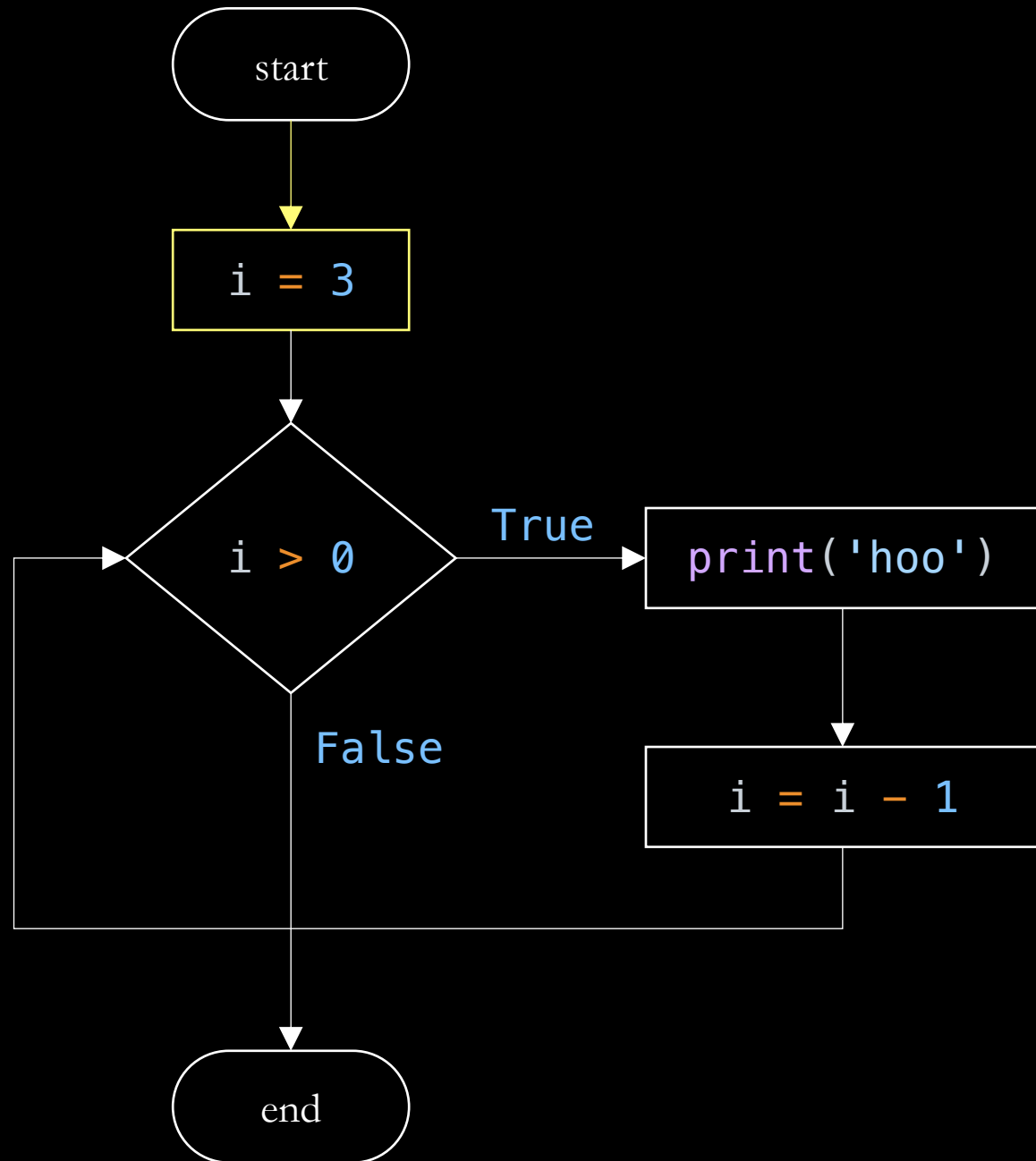
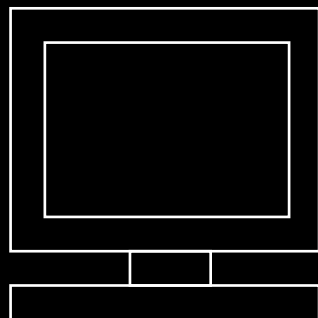




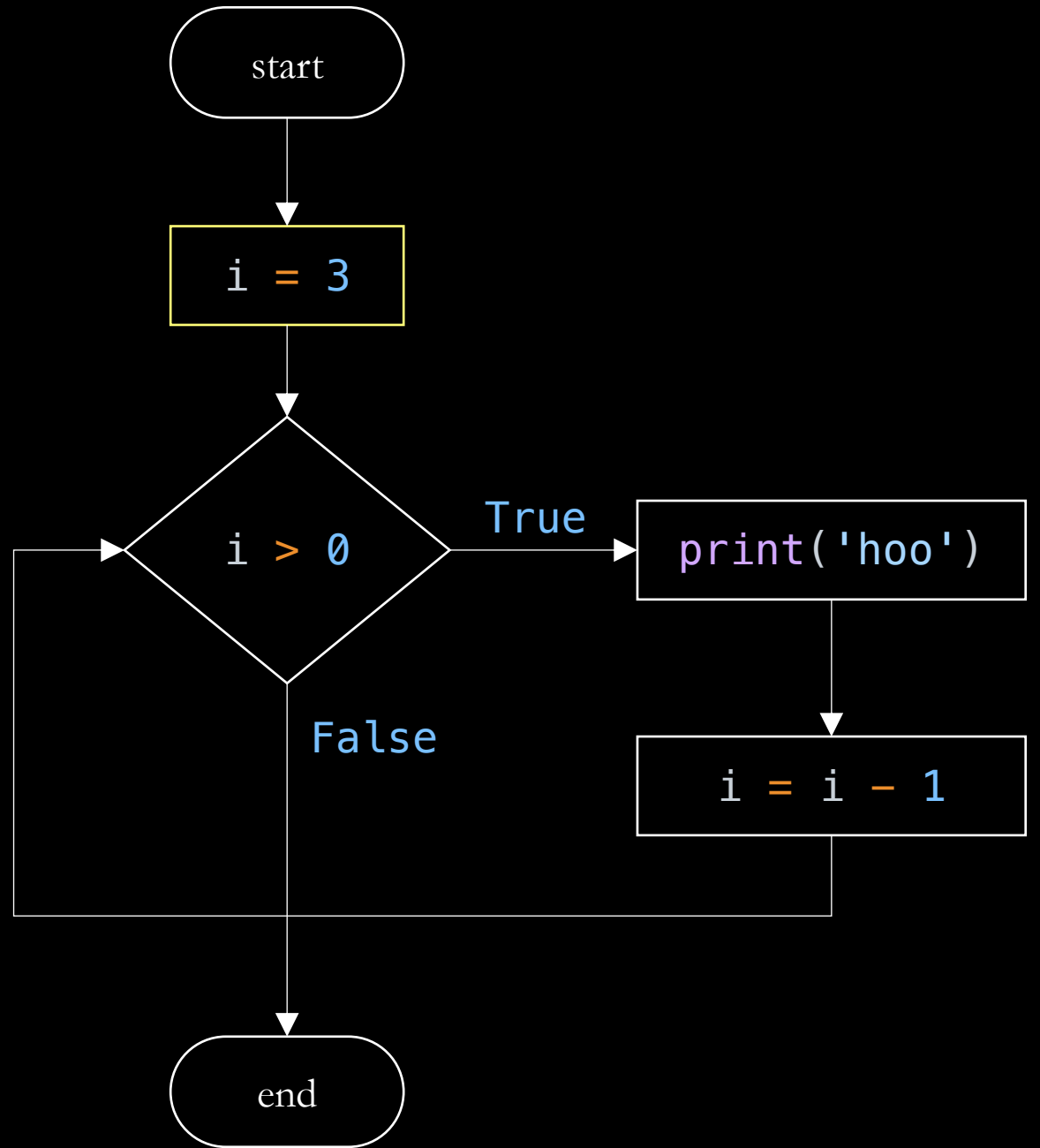
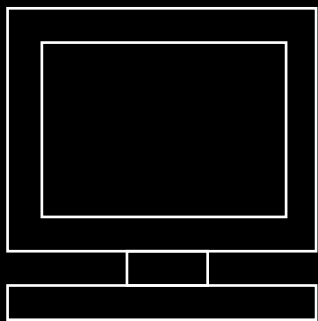
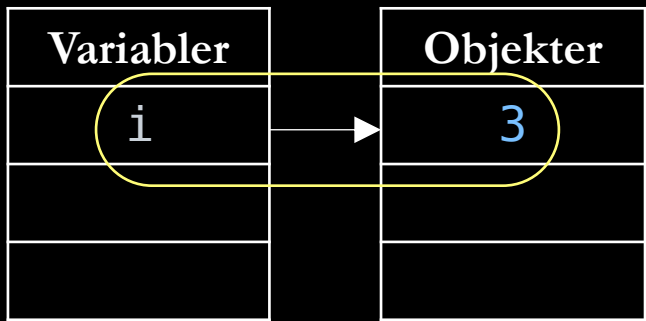
```
i = 3
while i > 0:
    print('hoo')
    i = i - 1
```

Variabler

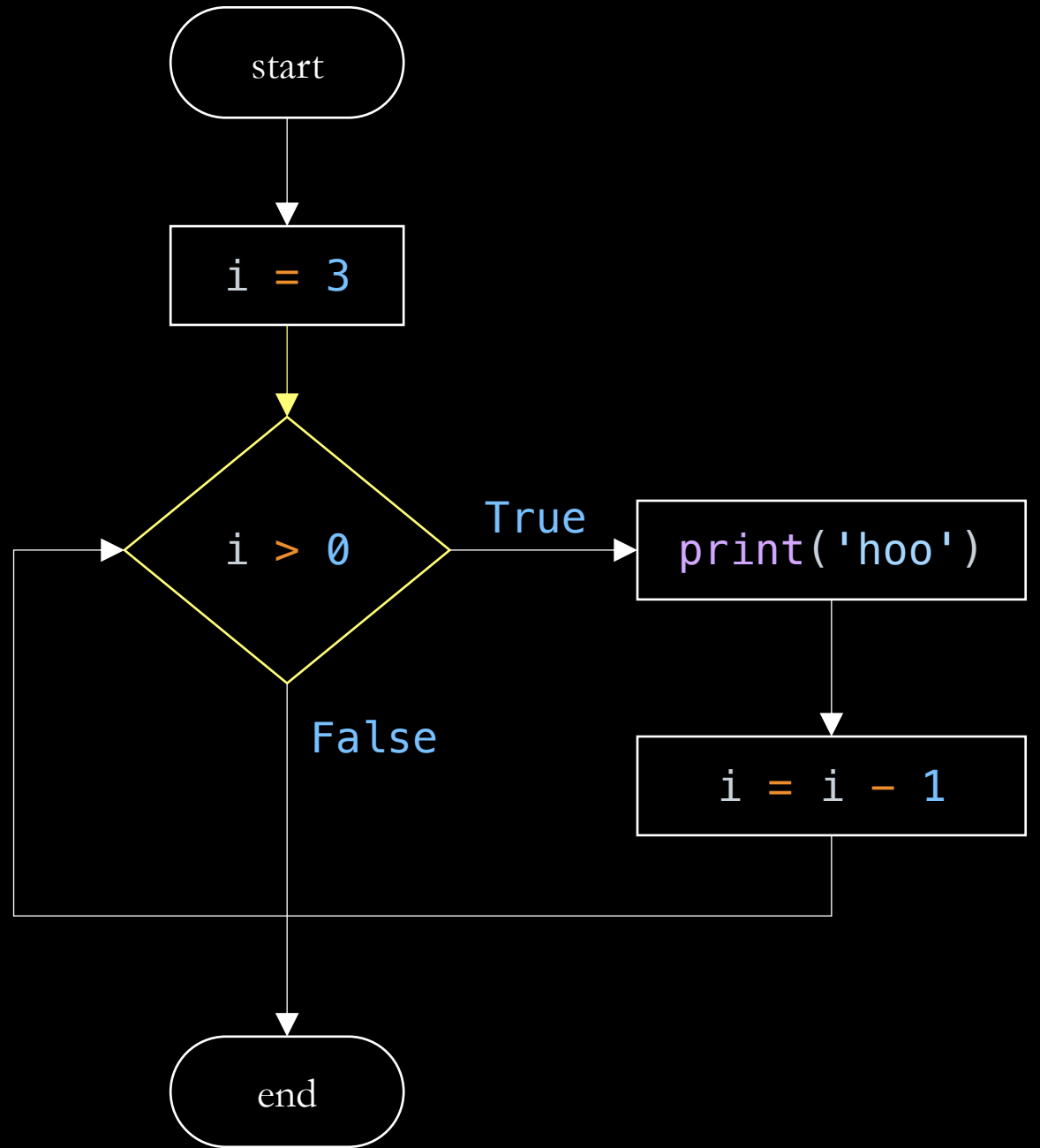
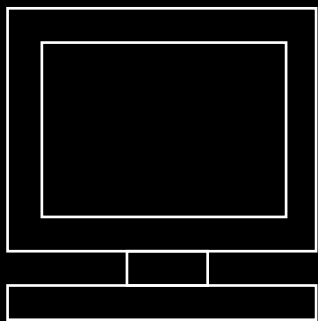
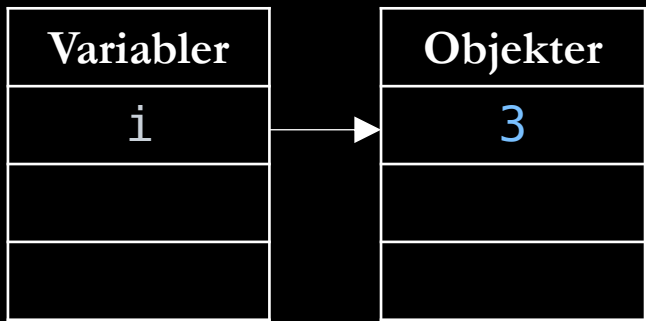
Objekter



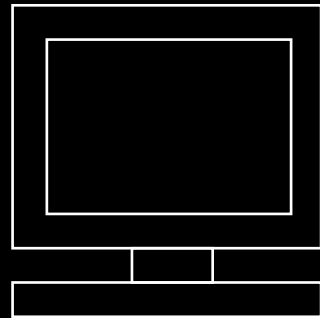
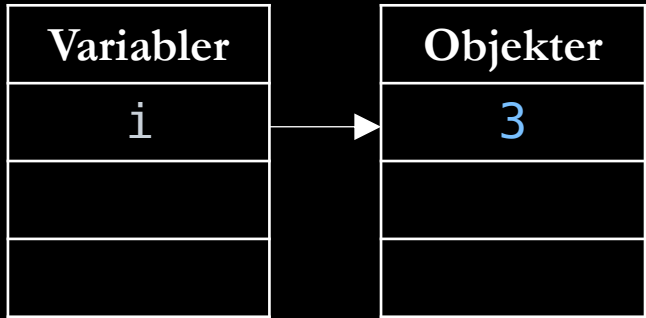
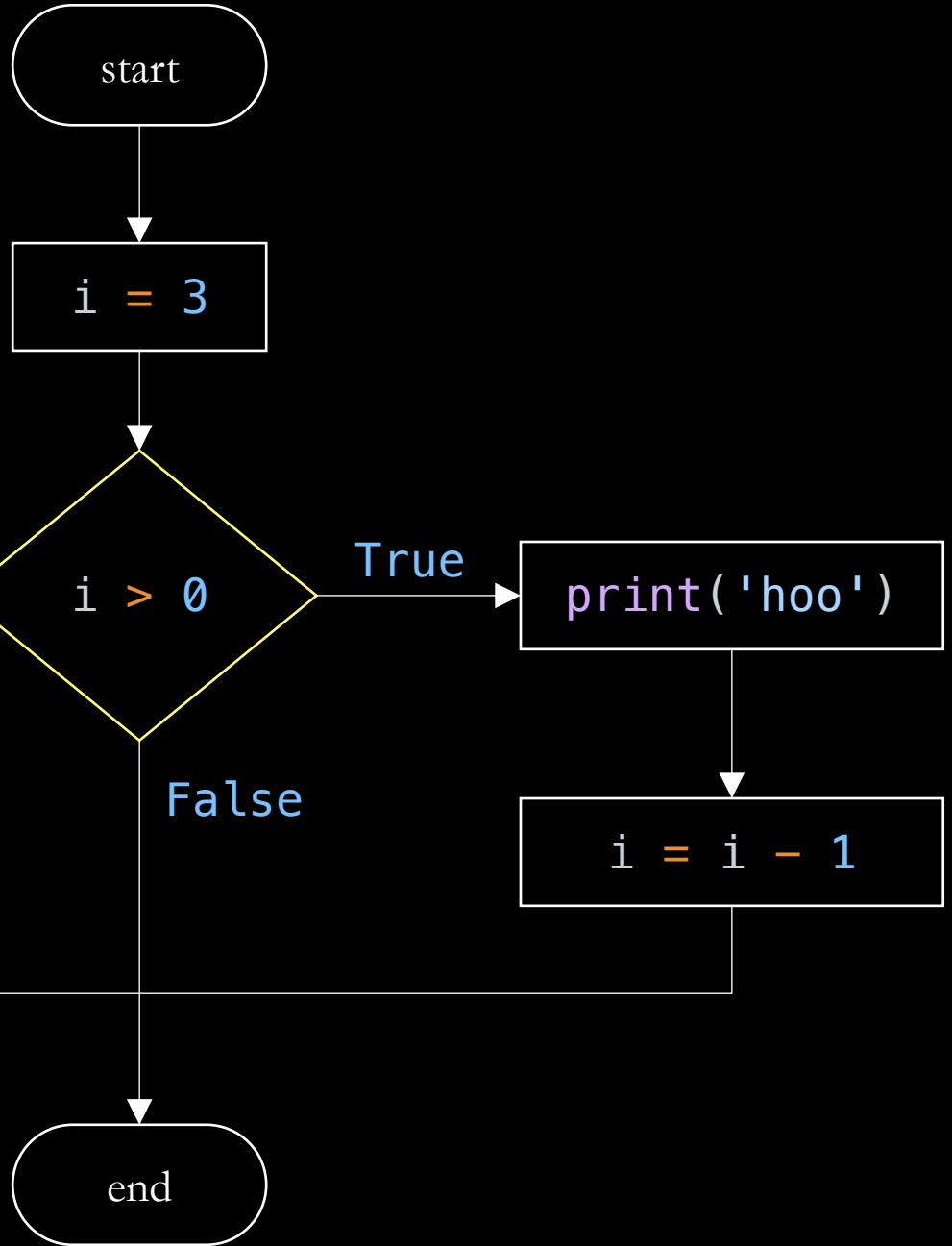
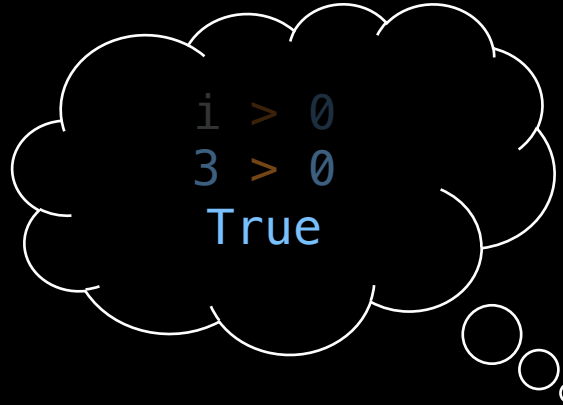
```
→ i = 3
while i > 0:
    print('hoo')
    i = i - 1
```



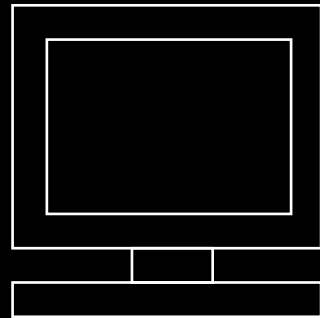
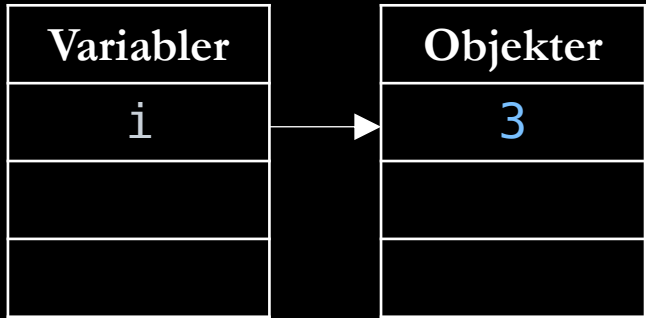
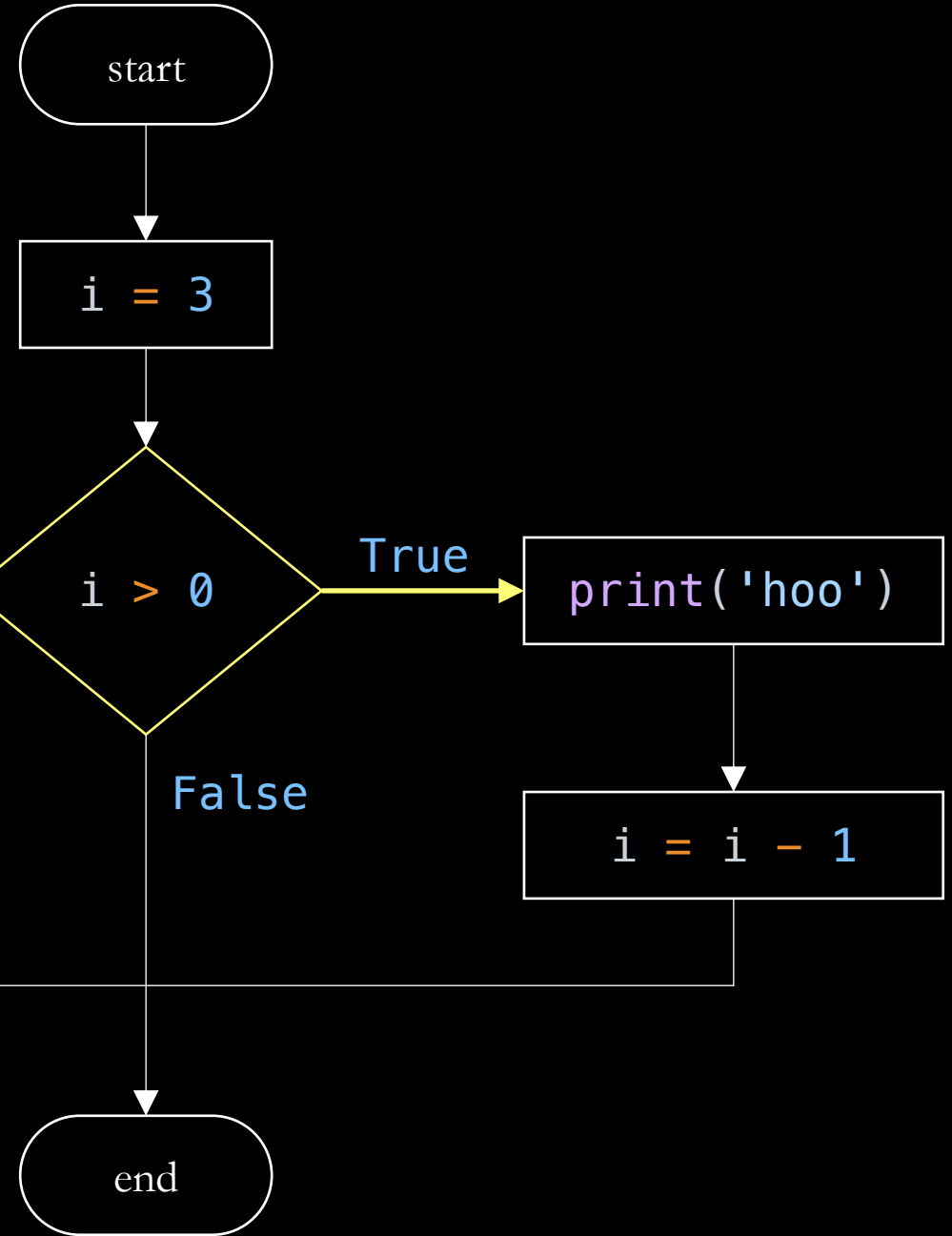
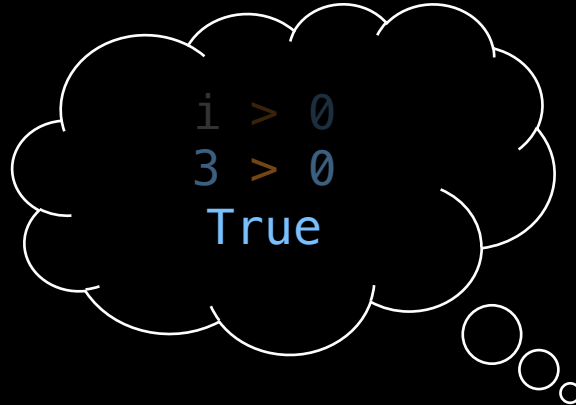
```
i = 3
while i > 0:
    print('hoo')
    i = i - 1
```



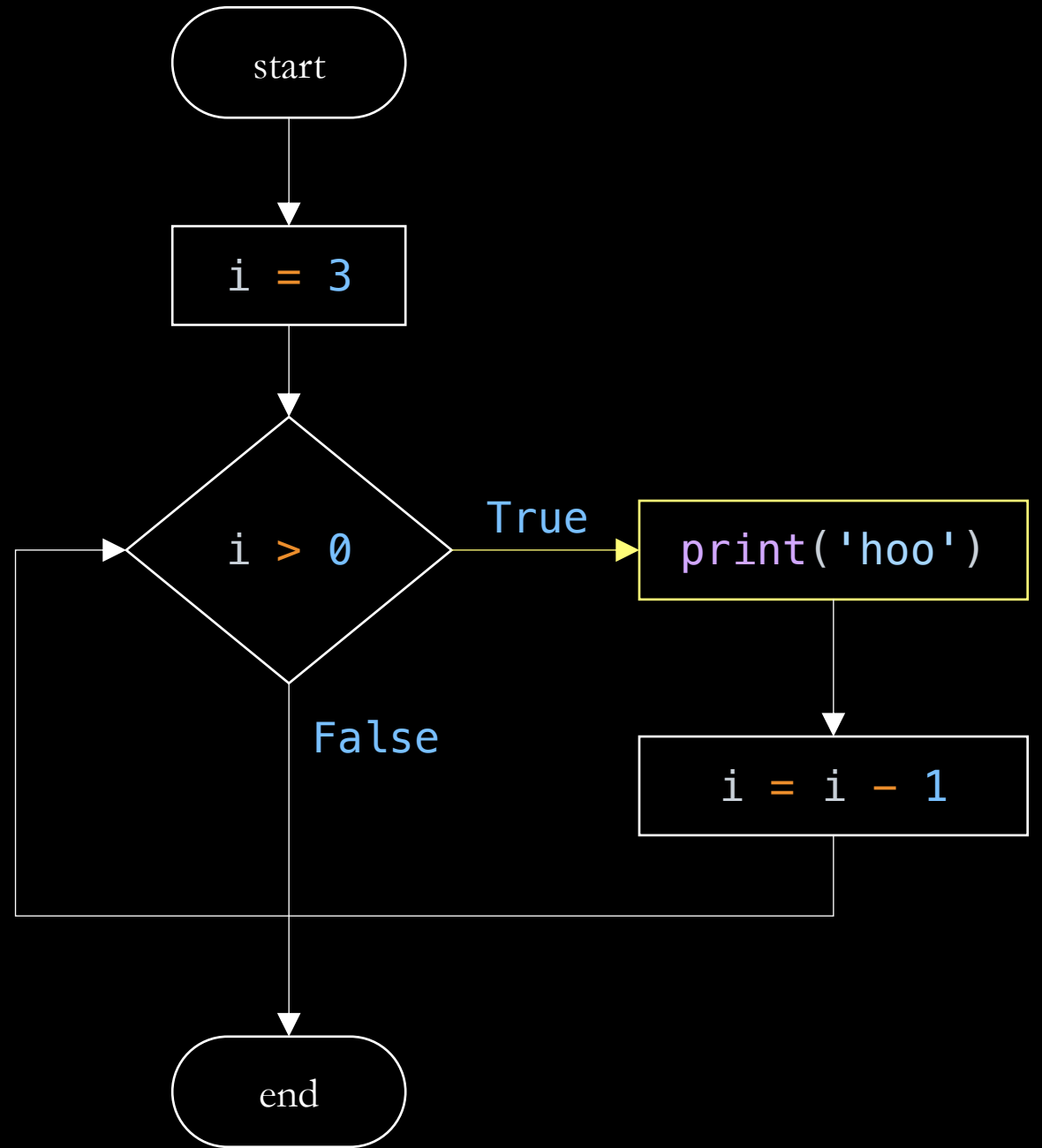
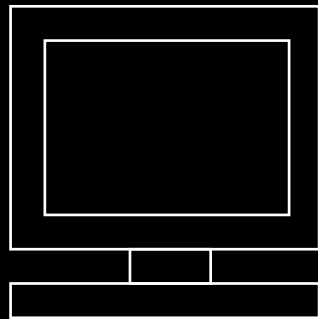
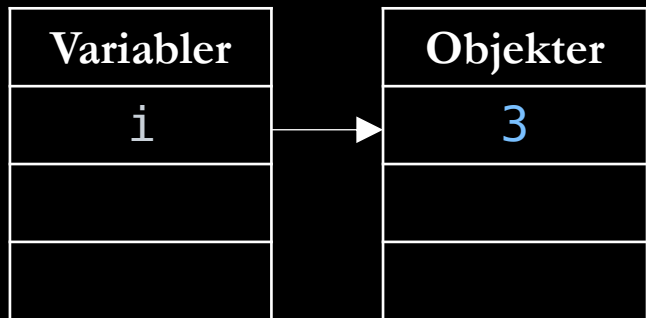
```
i = 3
while i > 0:
    print('hoo')
    i = i - 1
```



```
i = 3
while i > 0:
    print('hoo')
    i = i - 1
```



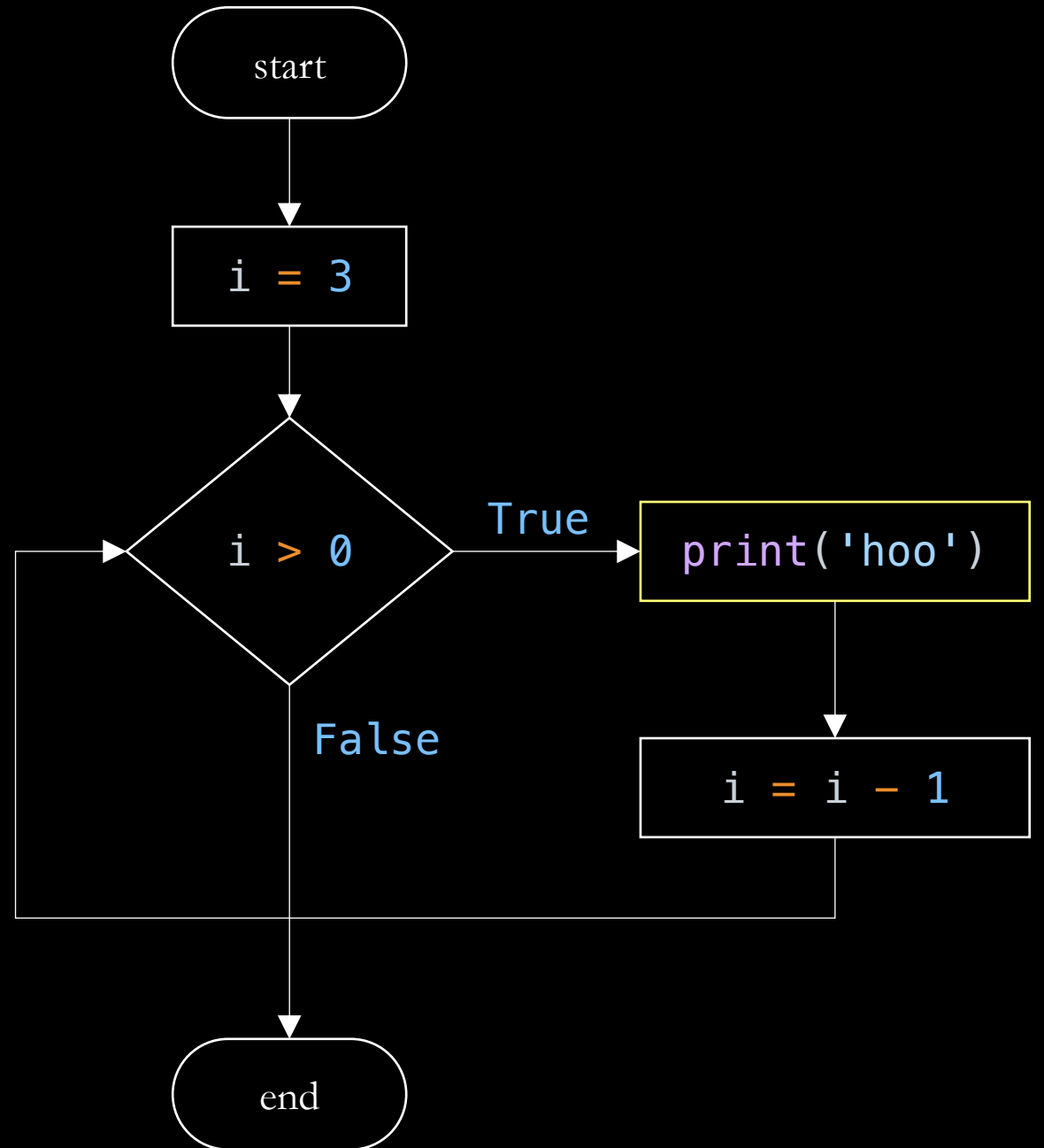
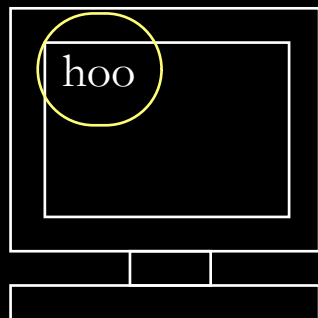
```
i = 3
while i > 0:
    print('hoo')
    i = i - 1
```




```
i = 3
while i > 0:
    print('hoo')
    i = i - 1
```

Variabler
i

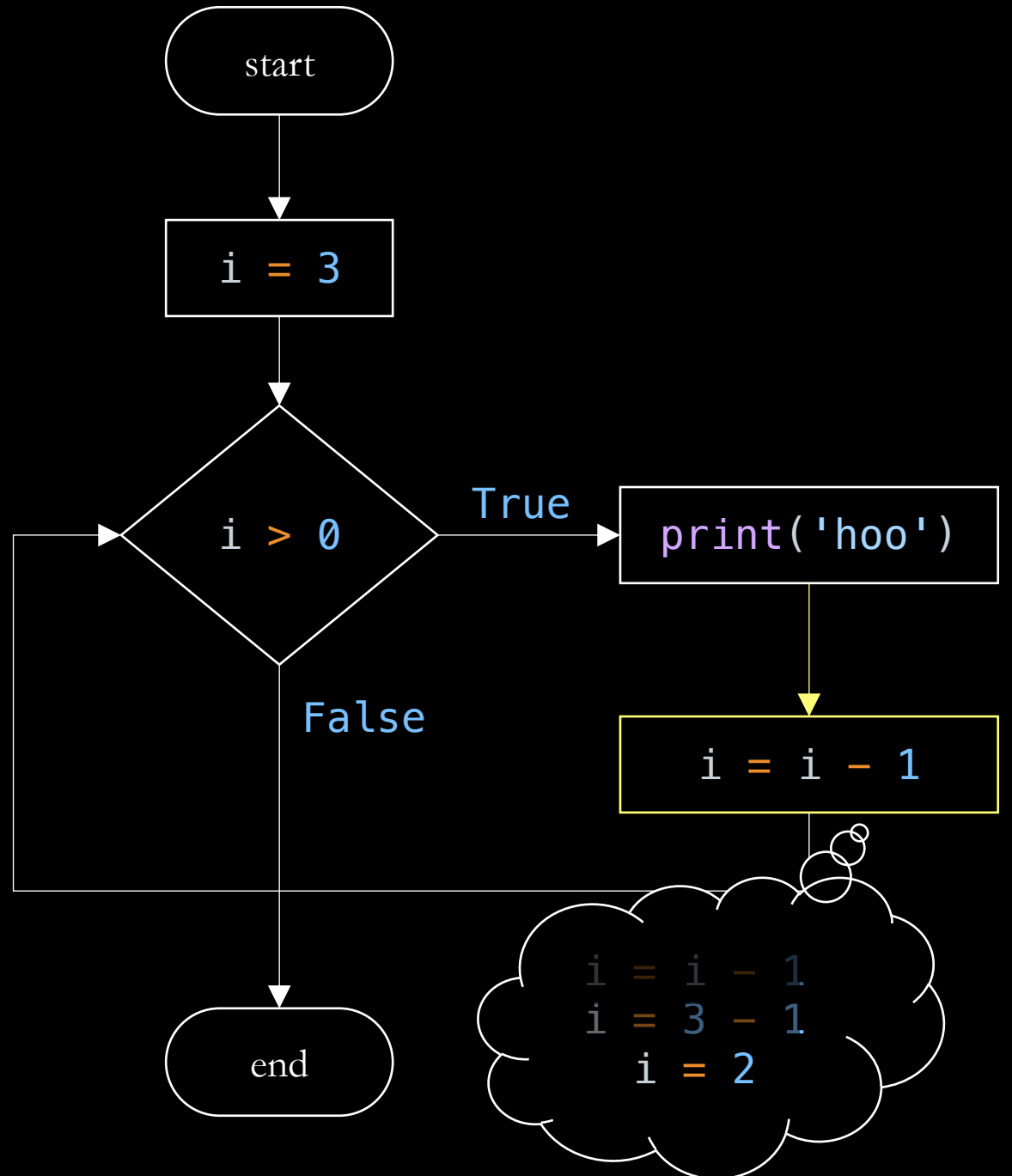
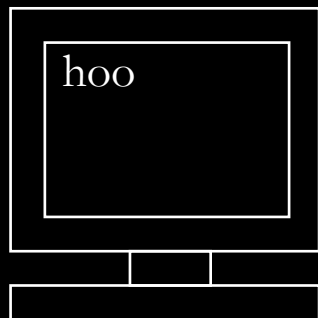
Objekter
3



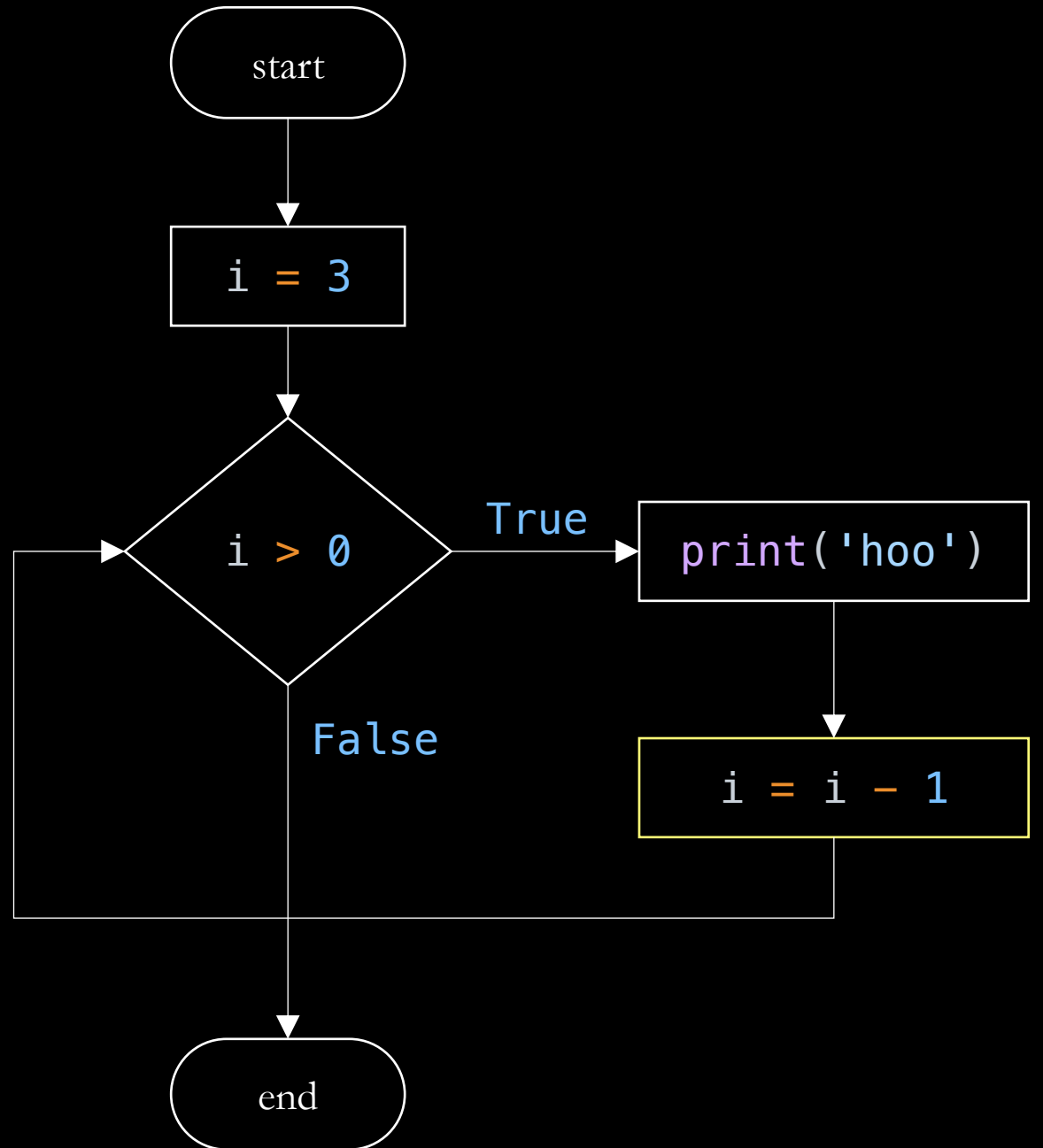
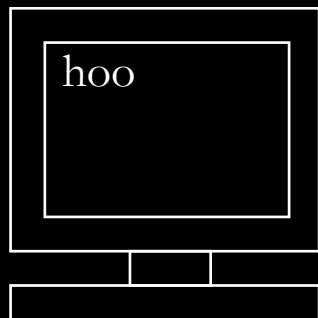
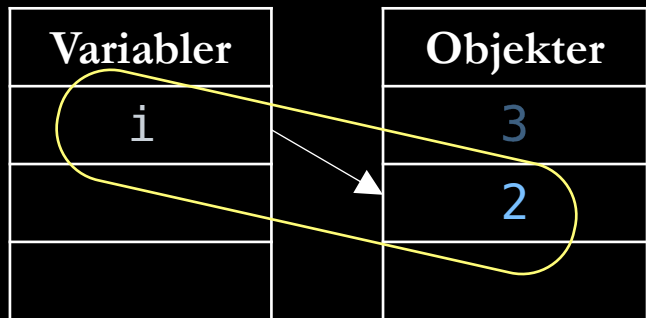
```
i = 3
while i > 0:
    print('hoo')
    i = i - 1
```

Variabler
i

Objekter
3



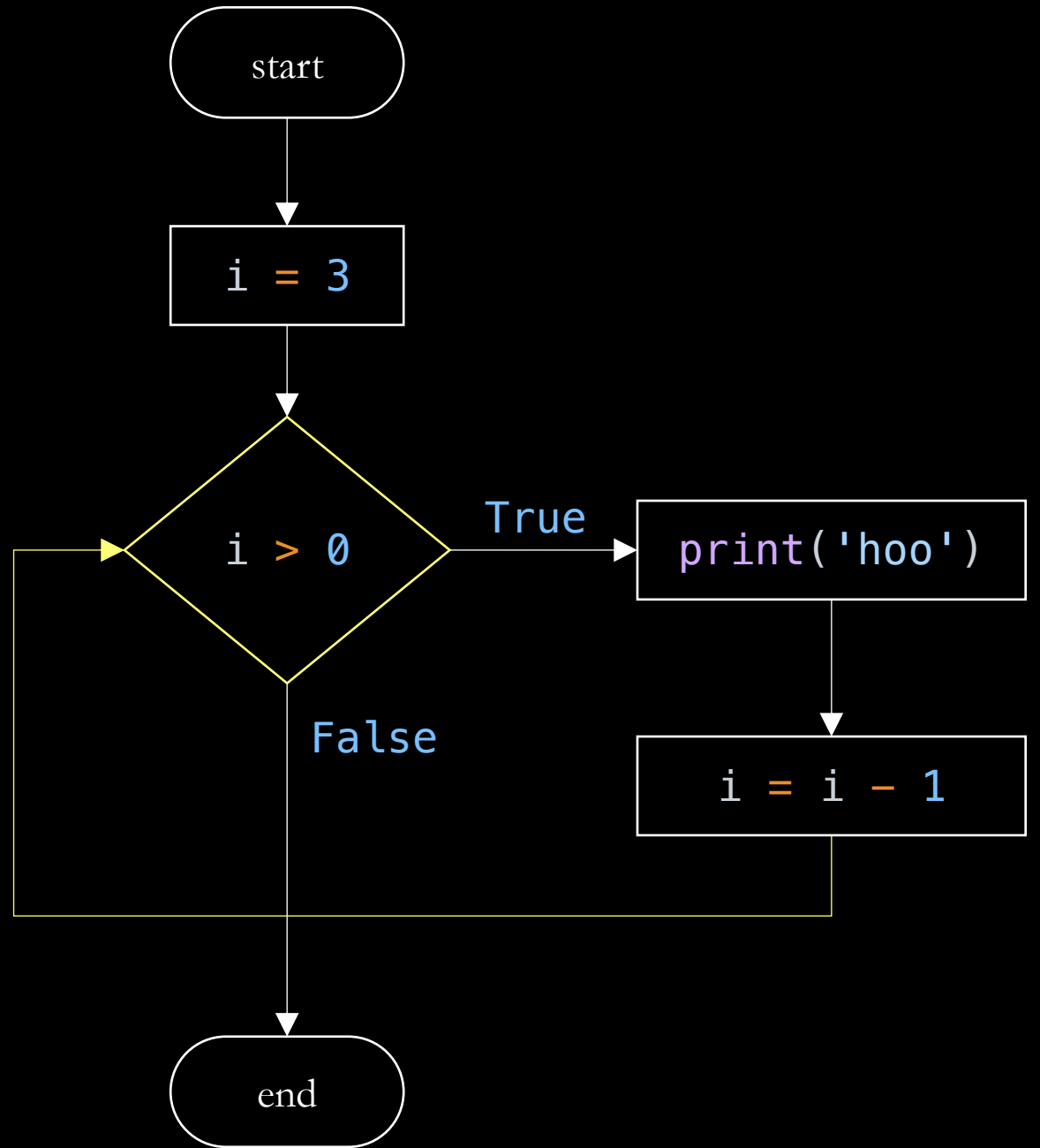
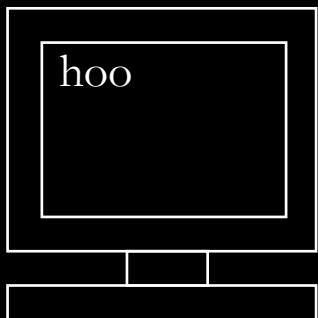
```
i = 3
while i > 0:
    print('hoo')
    i = i - 1
```



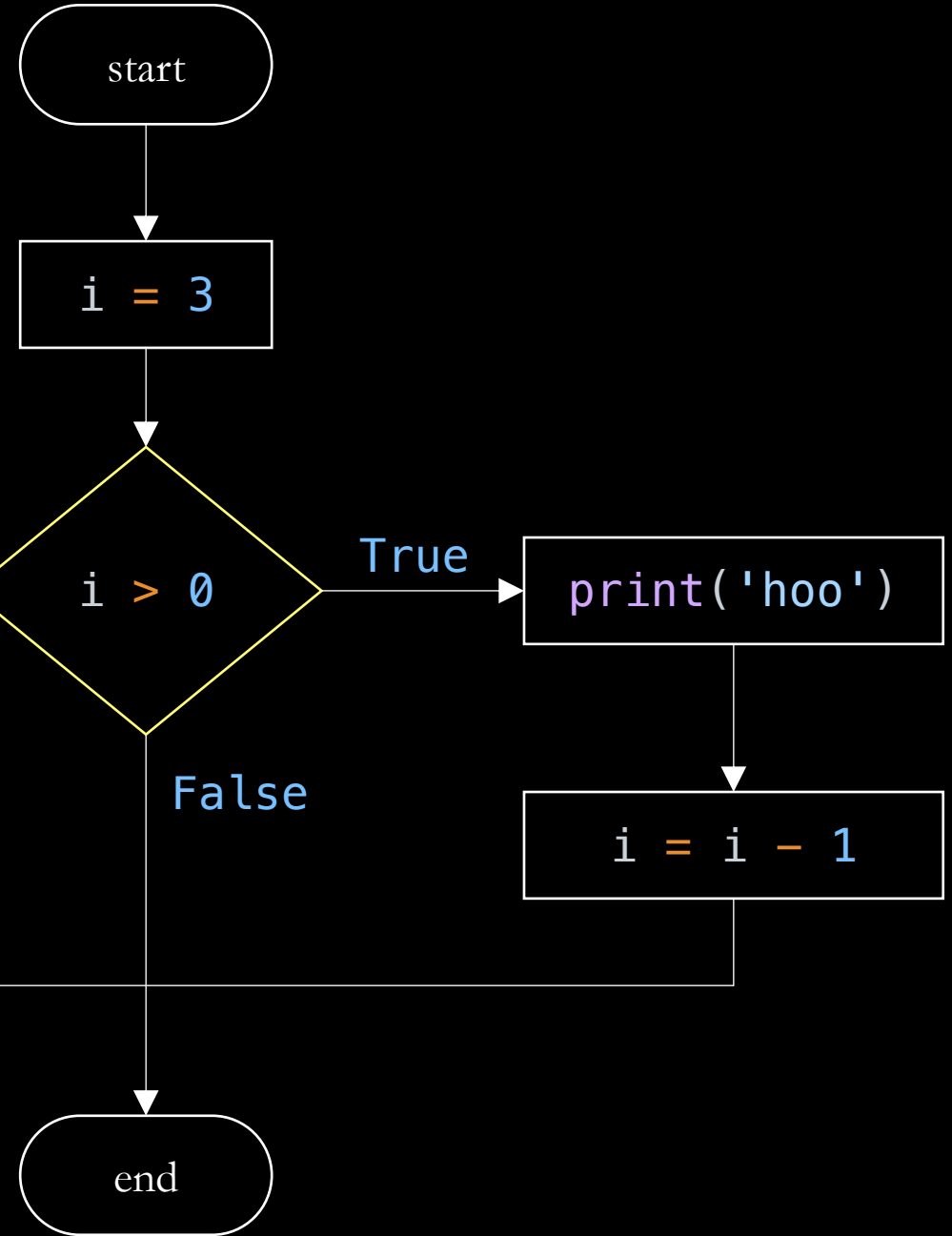
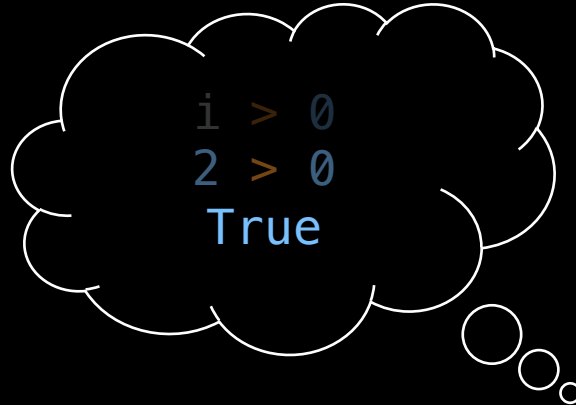
```
→ i = 3
while i > 0:
    print('hoo')
    i = i - 1
```

Variabler
i

Objekter
3
2

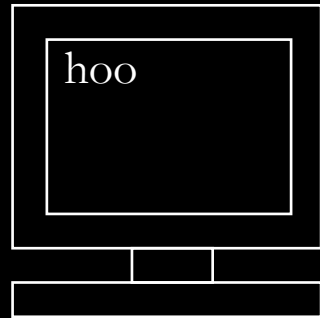


```
i = 3
while i > 0:
    print('hoo')
    i = i - 1
```

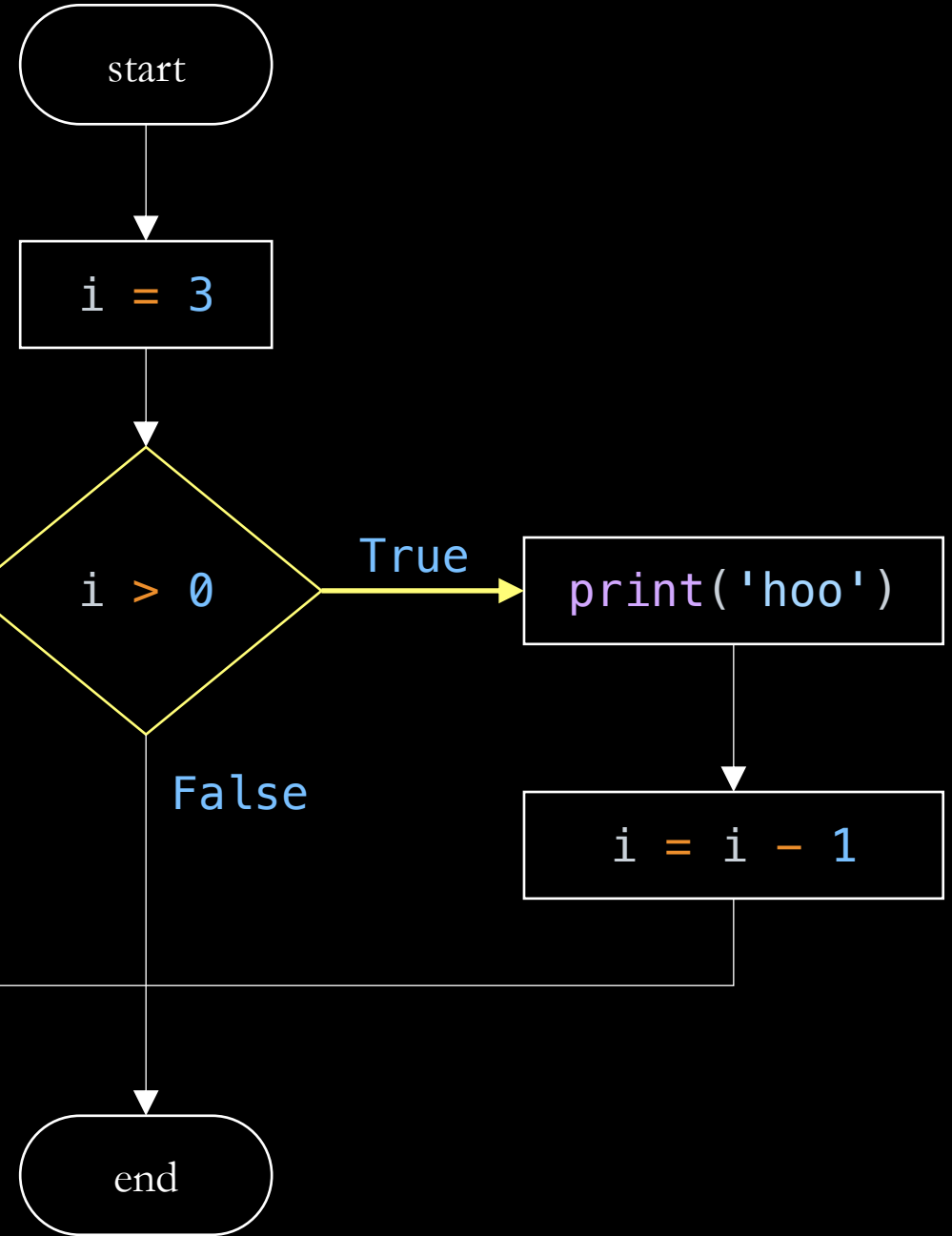
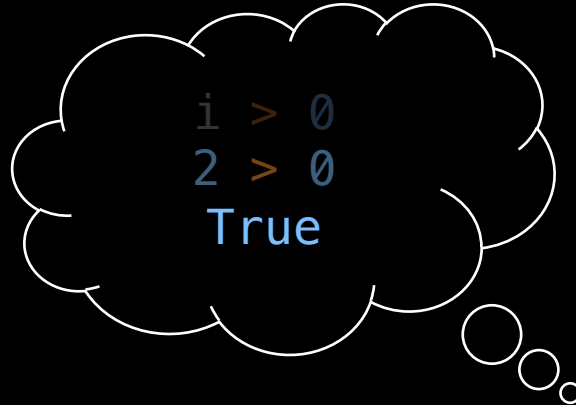


Variabler
i

Objekter
3
2

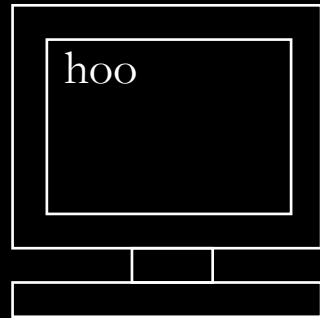


```
i = 3
while i > 0:
    print('hoo')
    i = i - 1
```



Variabler
i

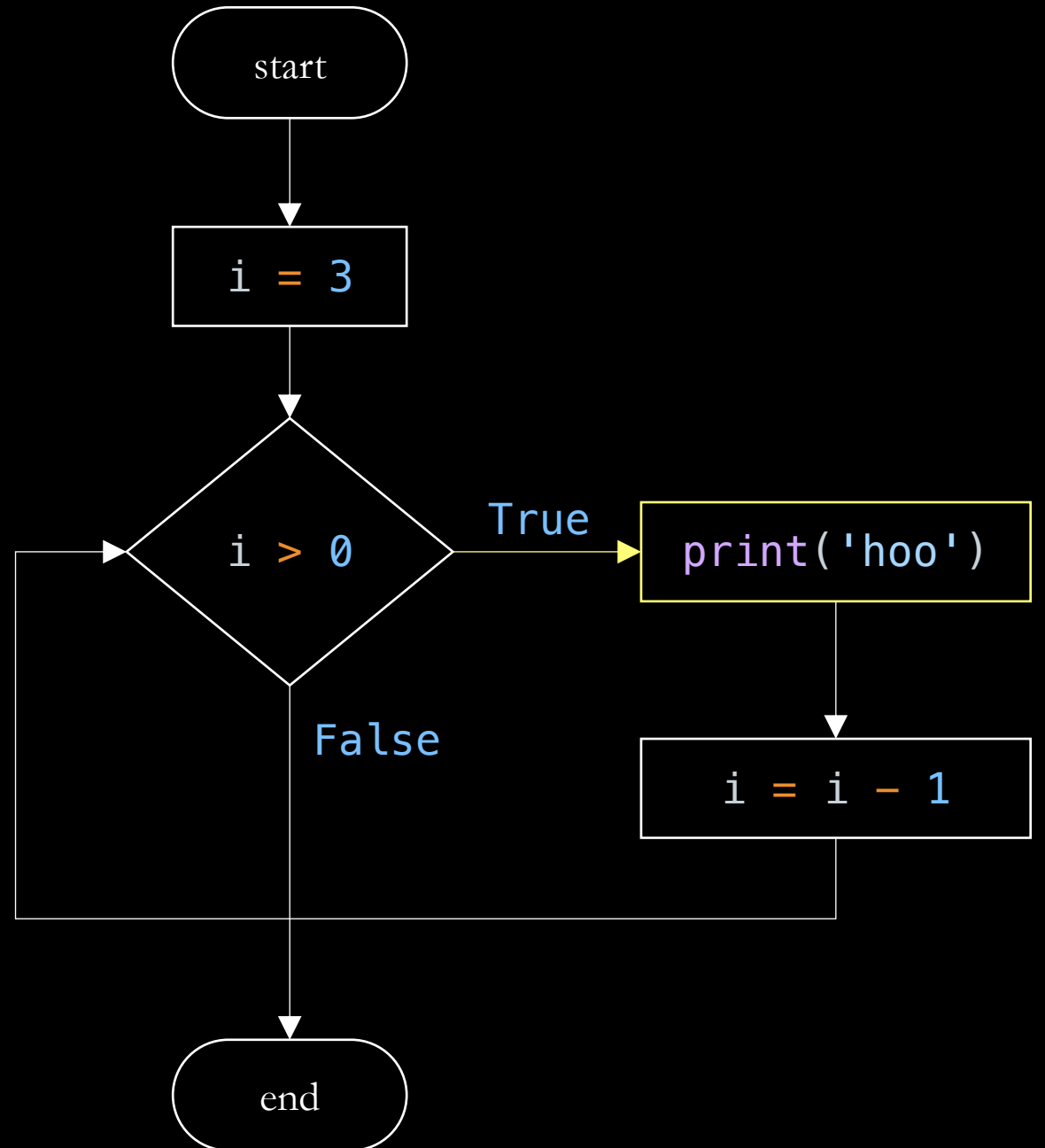
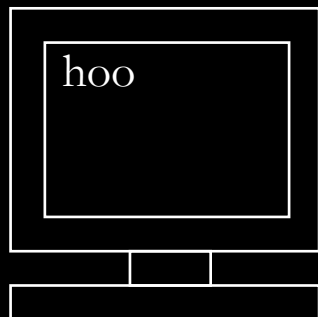
Objekter
3
2



```
i = 3
while i > 0:
    print('hoo')
    i = i - 1
```

Variabler
i

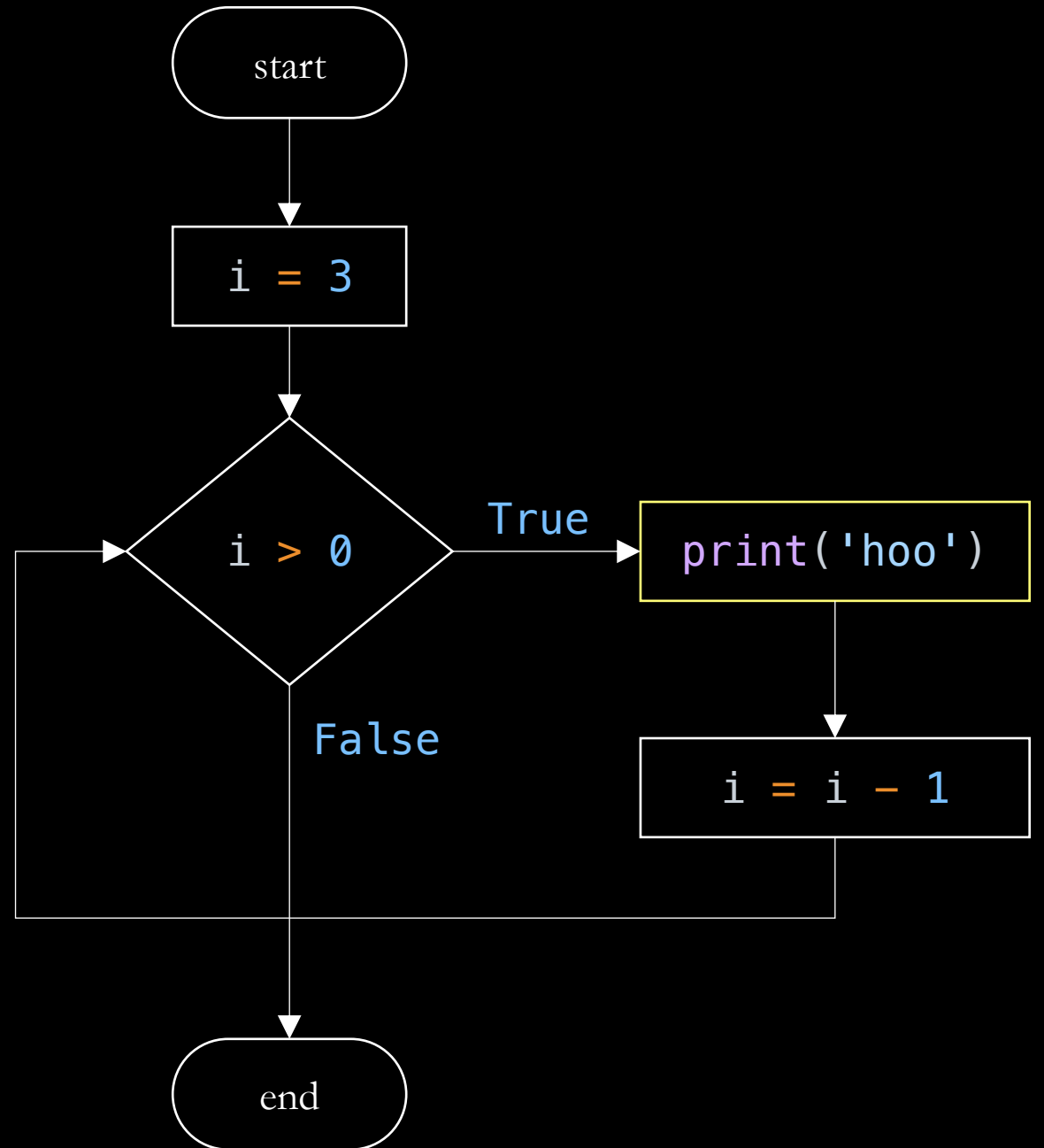
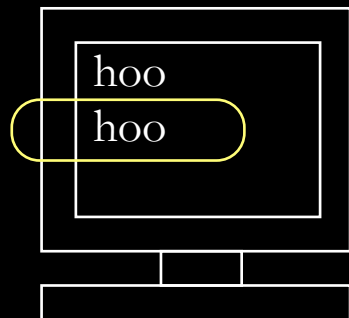
Objekter
3
2



```
i = 3
while i > 0:
    print('hoo')
    i = i - 1
```

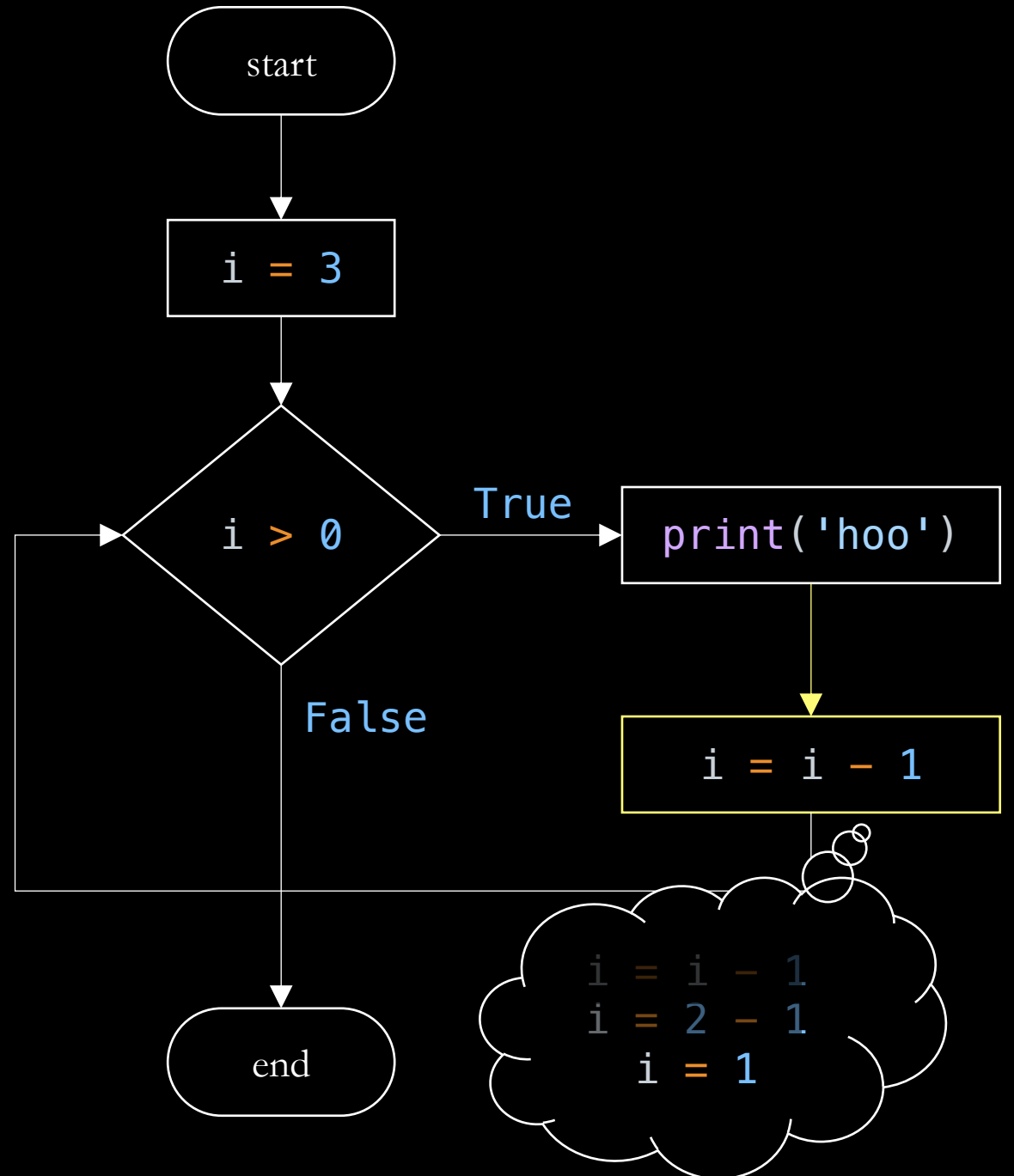
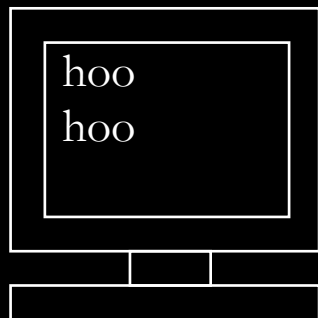
Variabler
i

Objekter
3
2



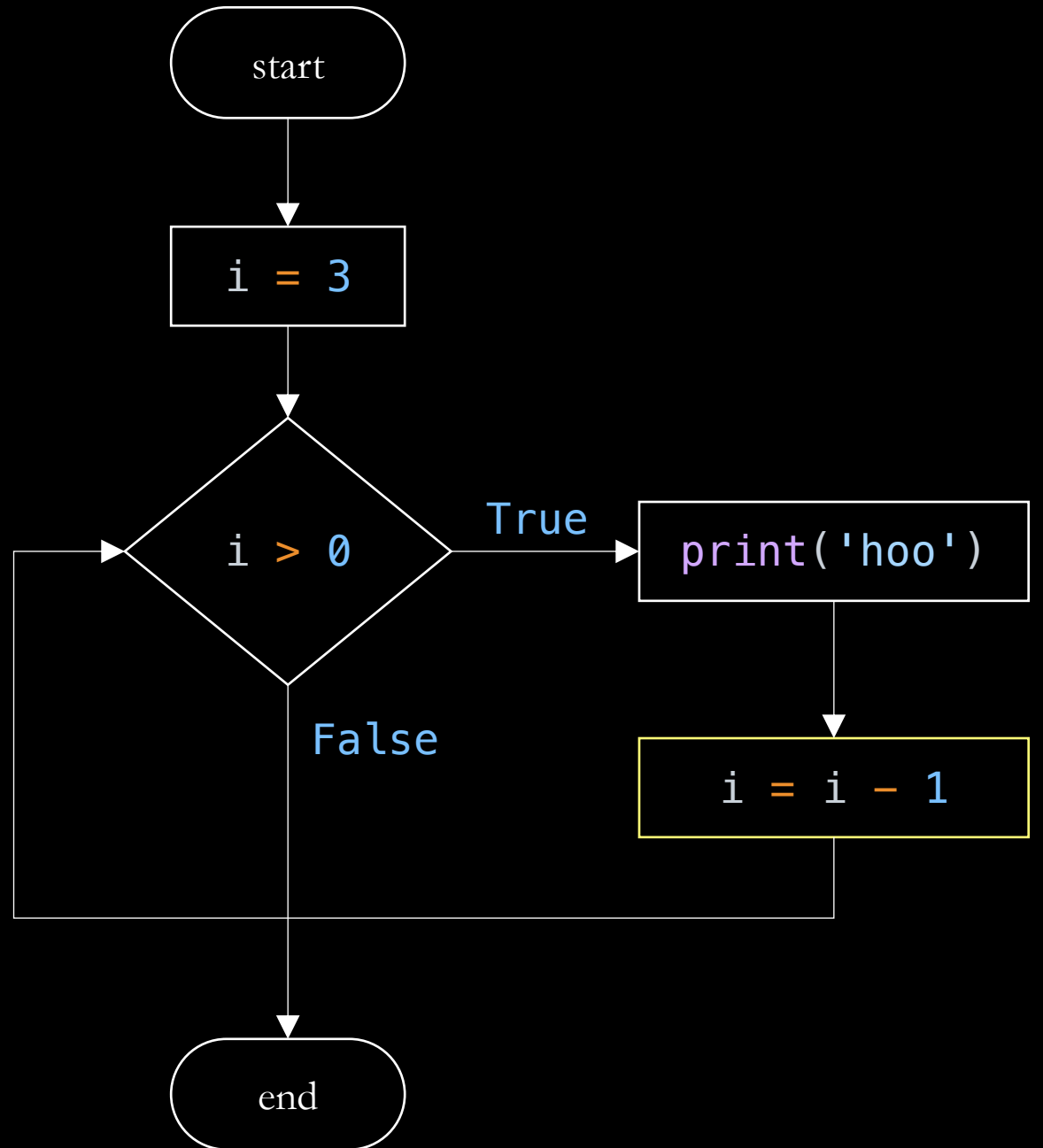
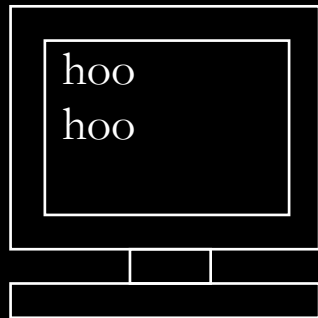

```
i = 3
while i > 0:
    print('hoo')
    i = i - 1
```

Variabler	Objekter
i	3
	2



```
i = 3
while i > 0:
    print('hoo')
    i = i - 1
```

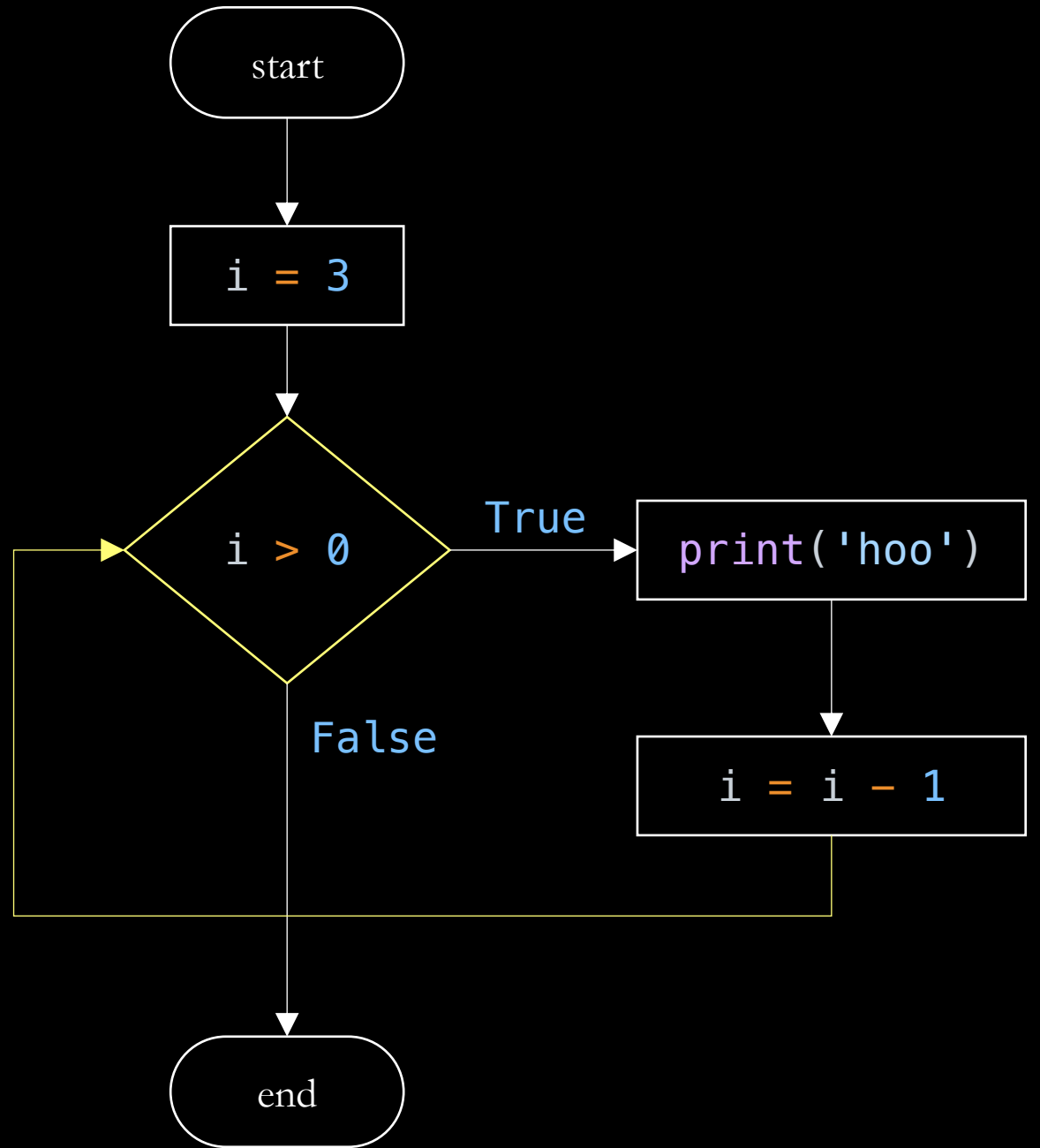
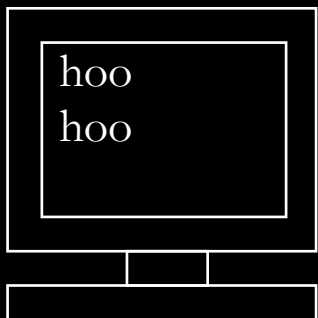
Variabler	Objekter
i	3
	2
	1



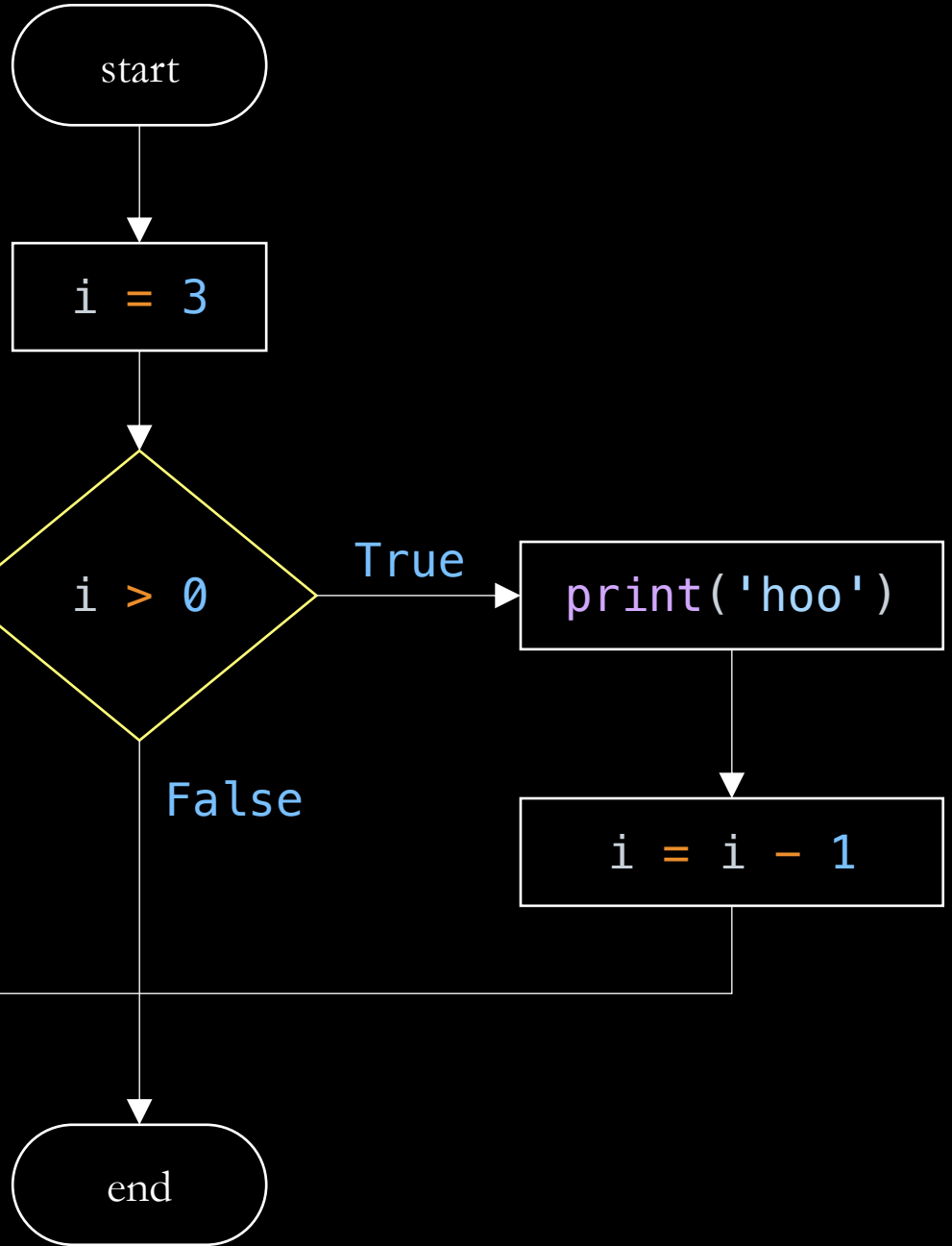
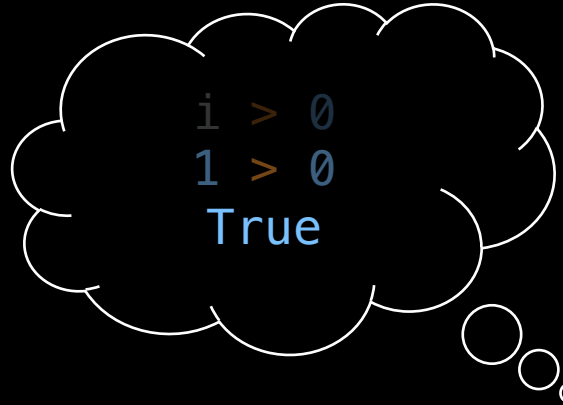
```
→ i = 3
while i > 0:
    print('hoo')
    i = i - 1
```

Variabler
i

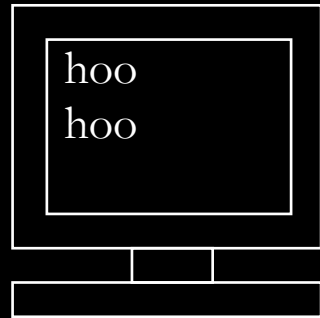
Objekter
3
2
1



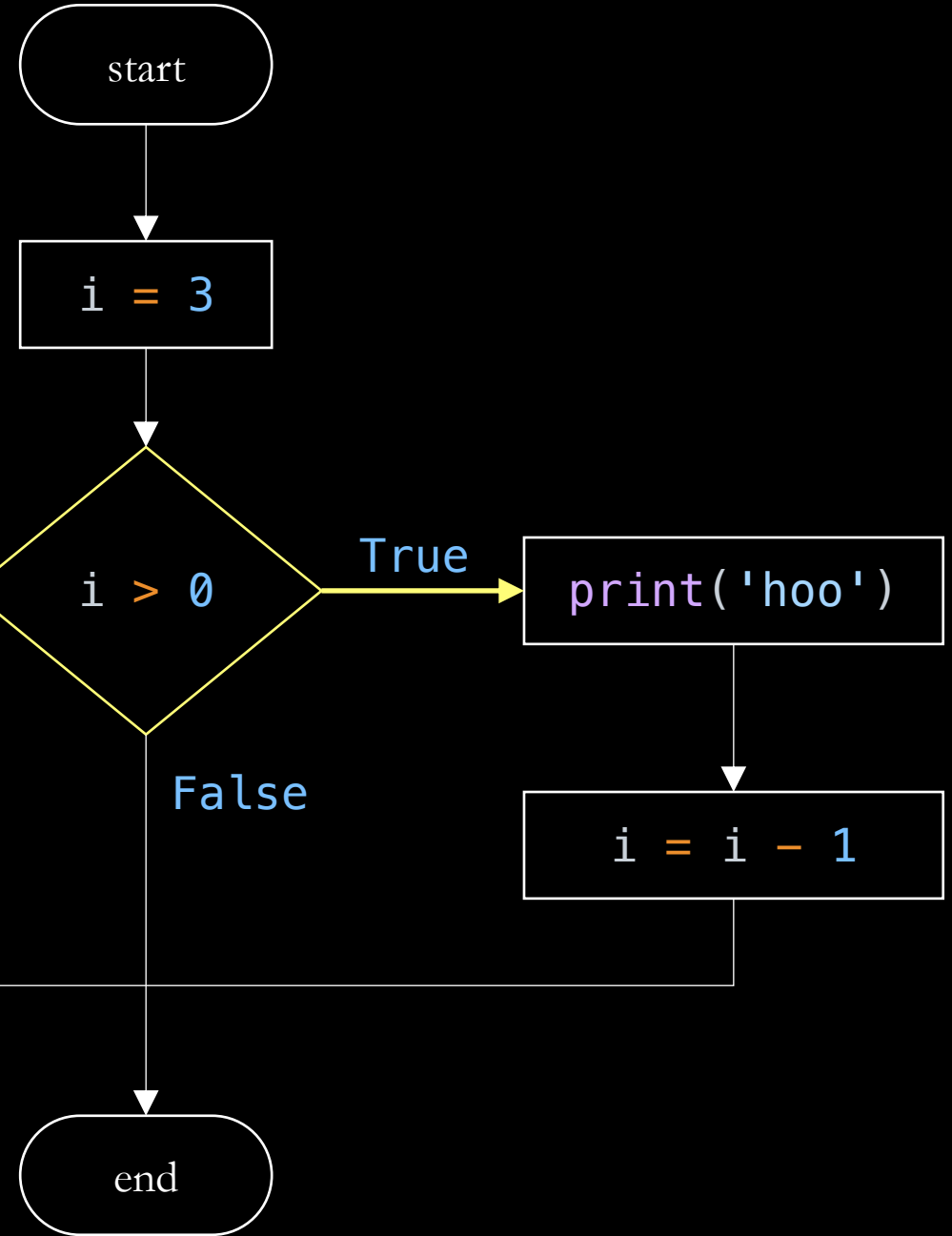
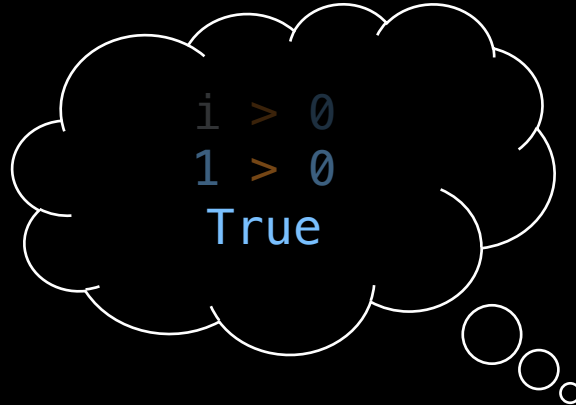
```
i = 3
while i > 0:
    print('hoo')
    i = i - 1
```



Variabler	Objekter
i	3
	2
	1

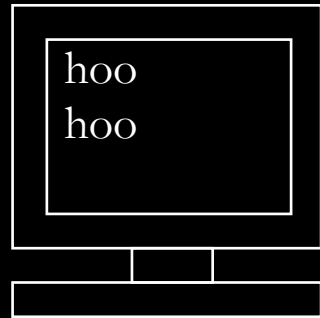


```
i = 3
while i > 0:
    print('hoo')
    i = i - 1
```



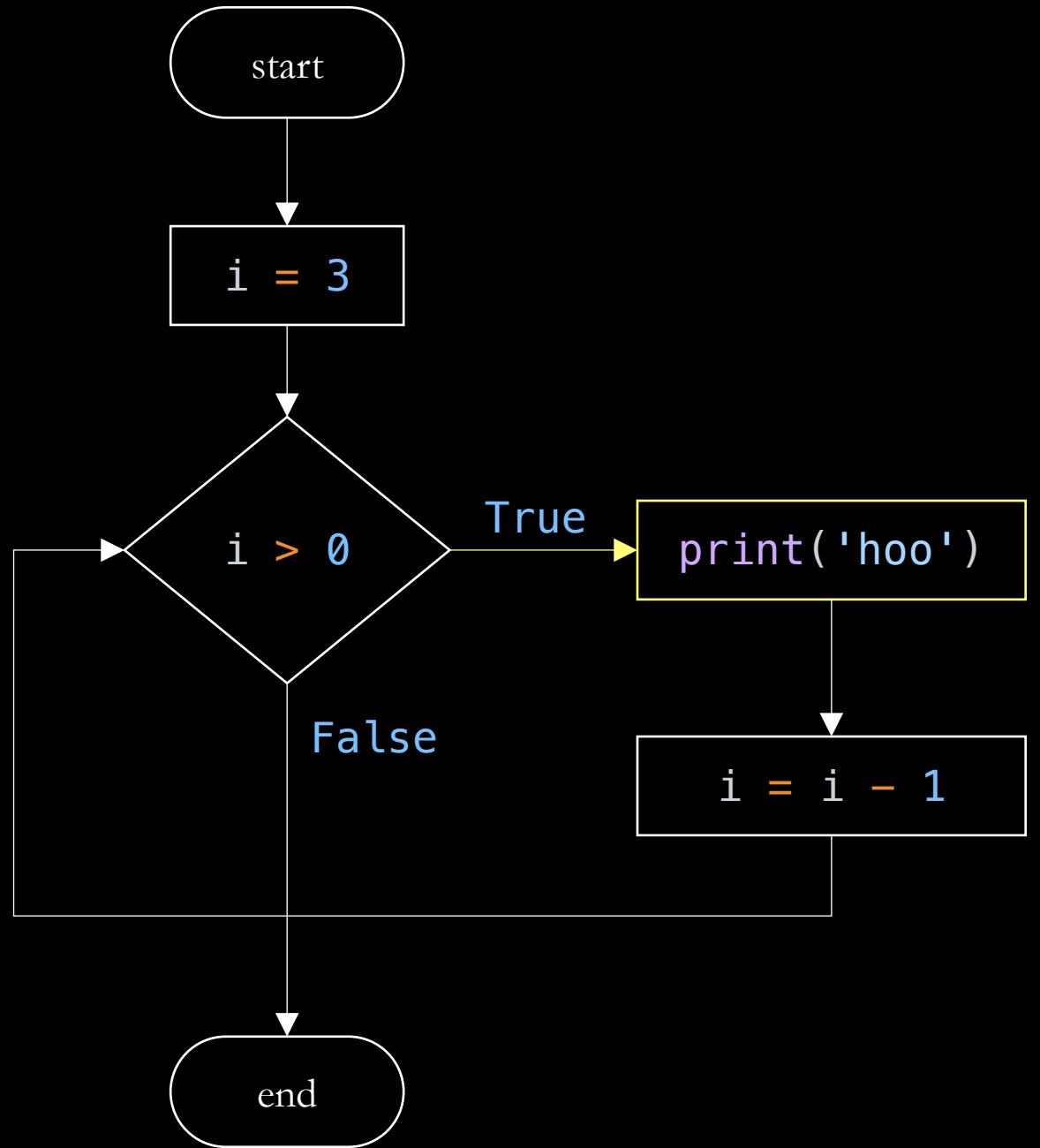
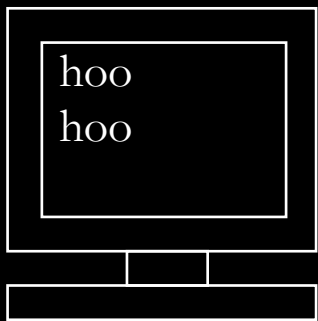
Variabler
i

Objekter
3
2
1



```
i = 3
while i > 0:
    print('hoo')
    i = i - 1
```

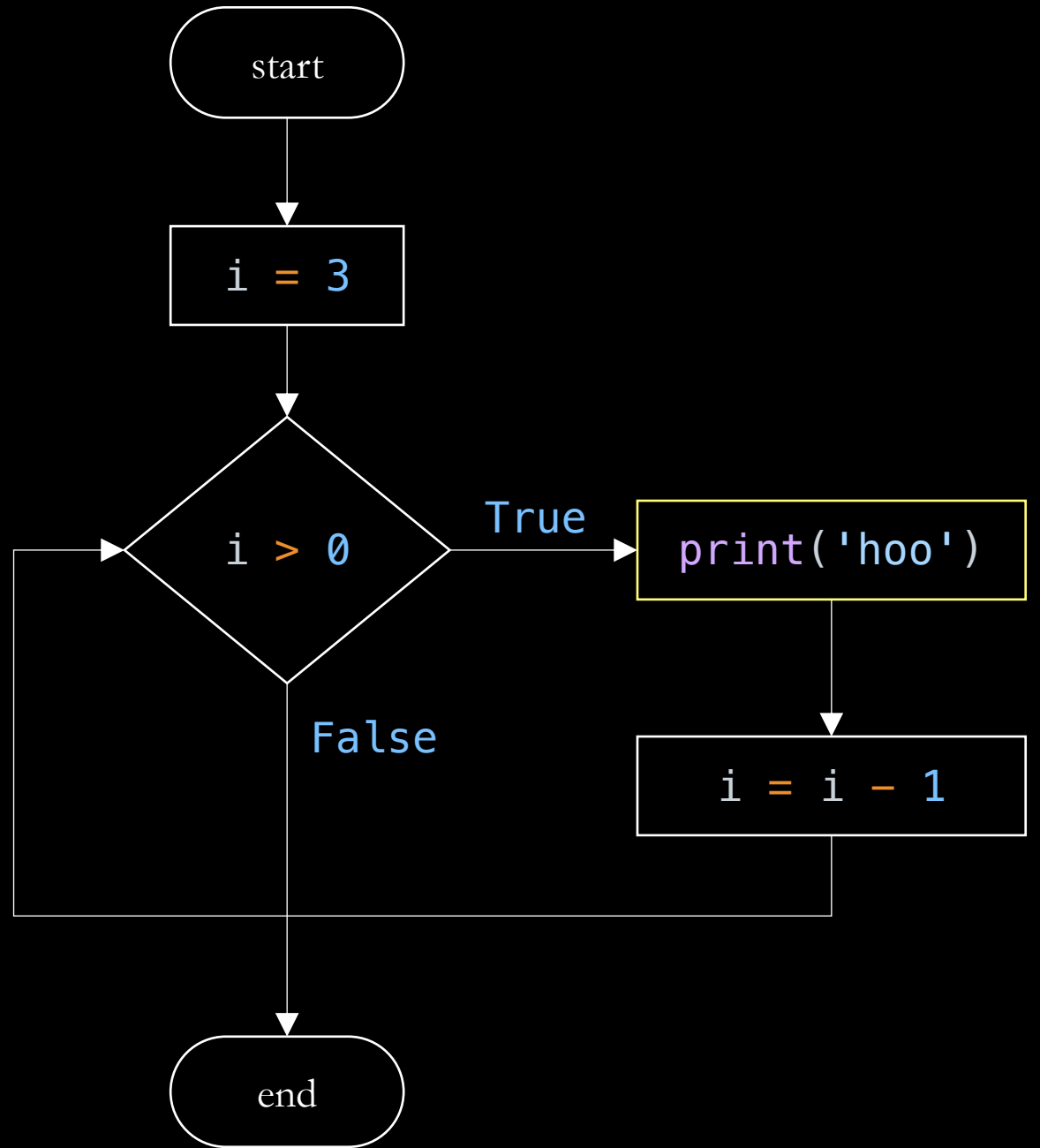
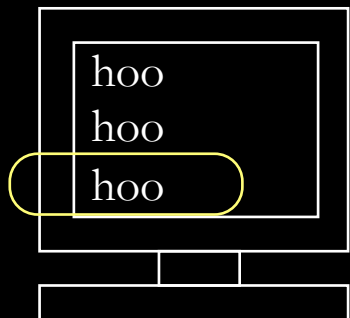
Variabler	Objekter
i	3
	2
	1



```
i = 3
while i > 0:
    print('hoo')
    i = i - 1
```

Variabler
i

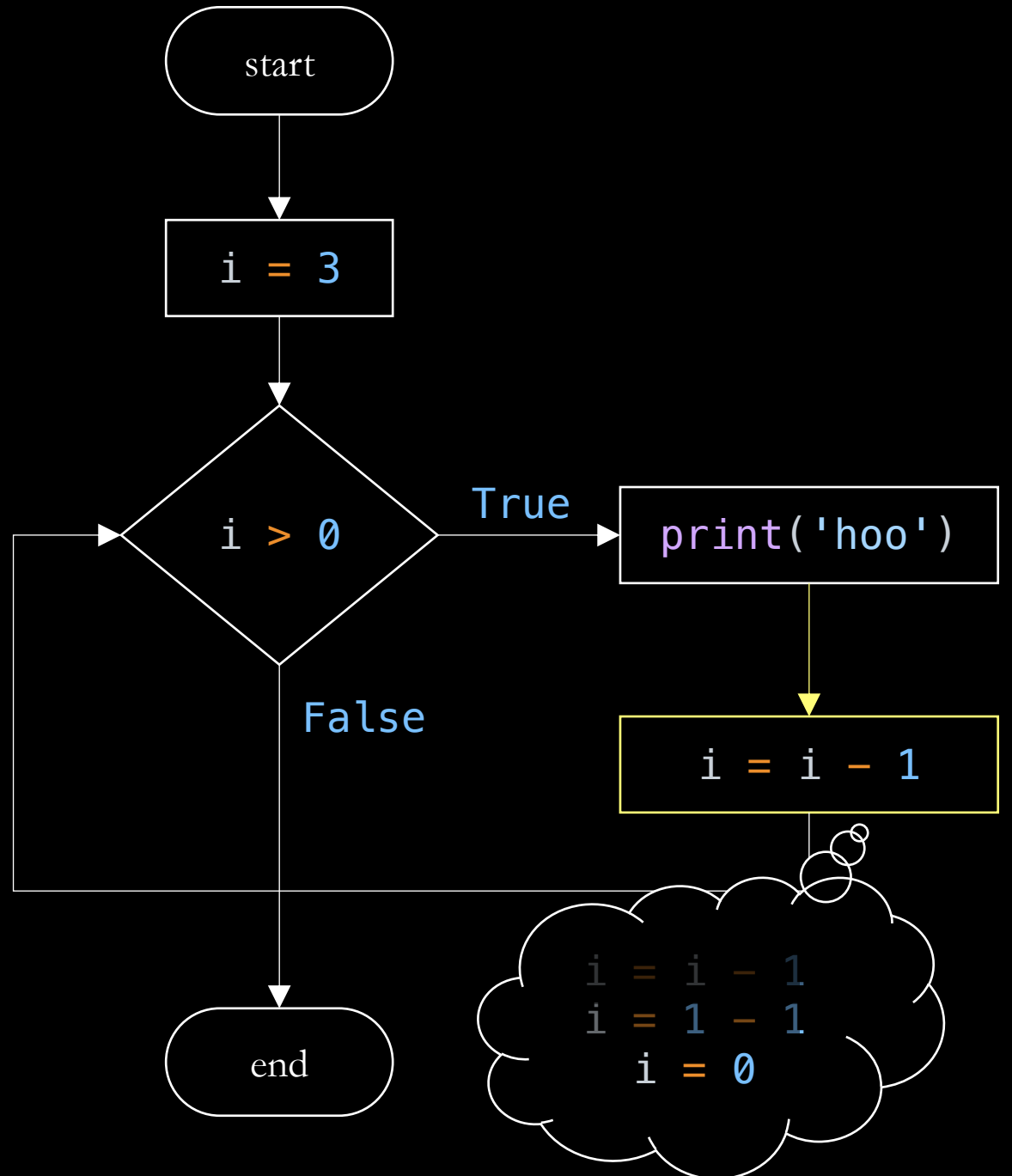
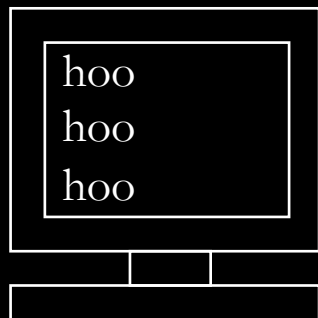
Objekter
3
2
1



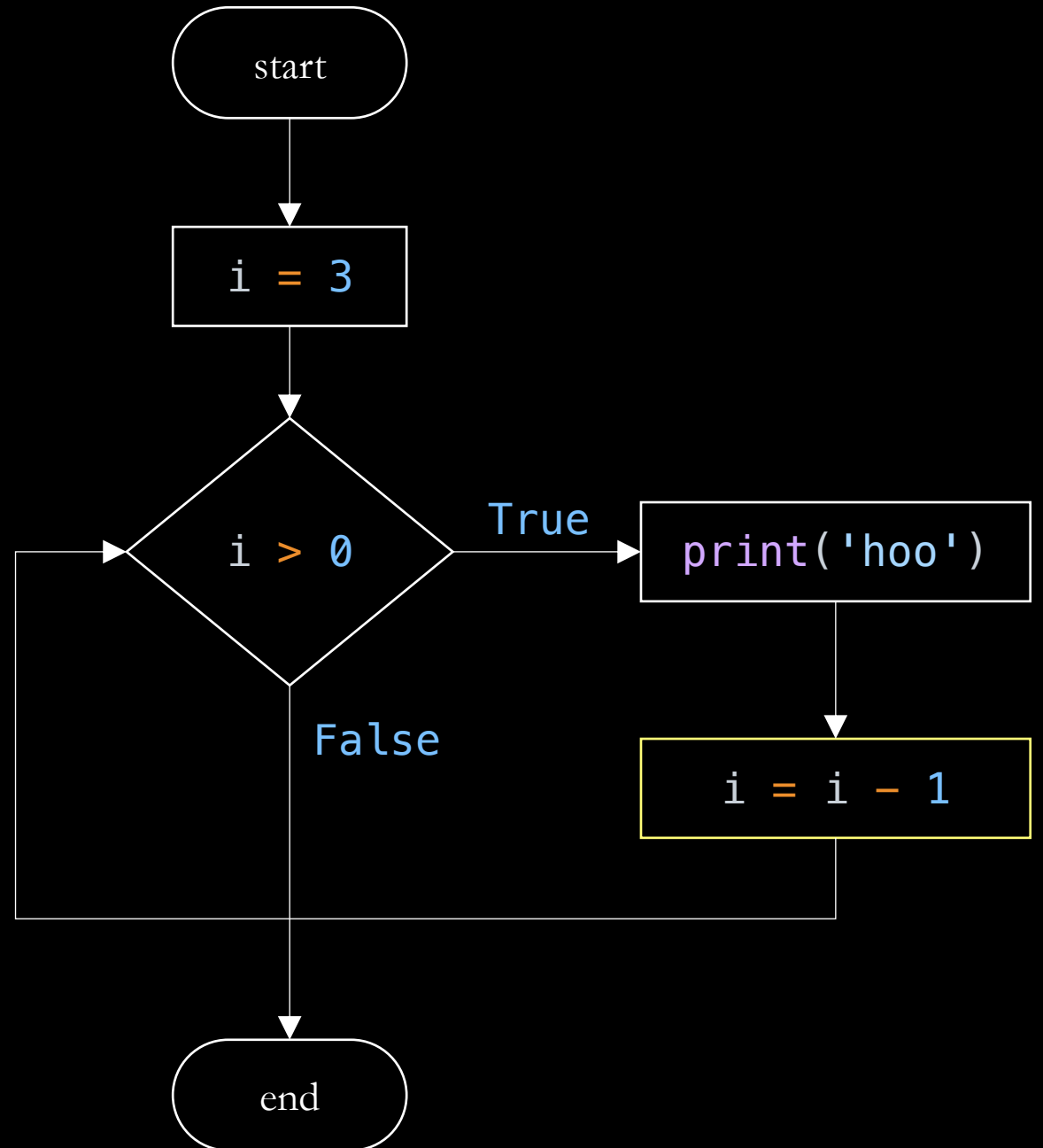
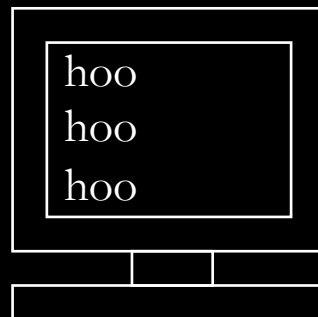
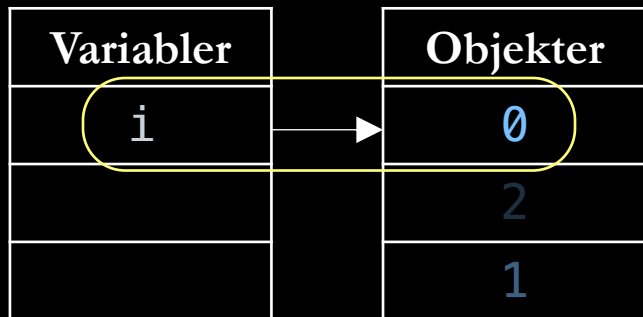
```
i = 3
while i > 0:
    print('hoo')
    i = i - 1
```

Variabler
i

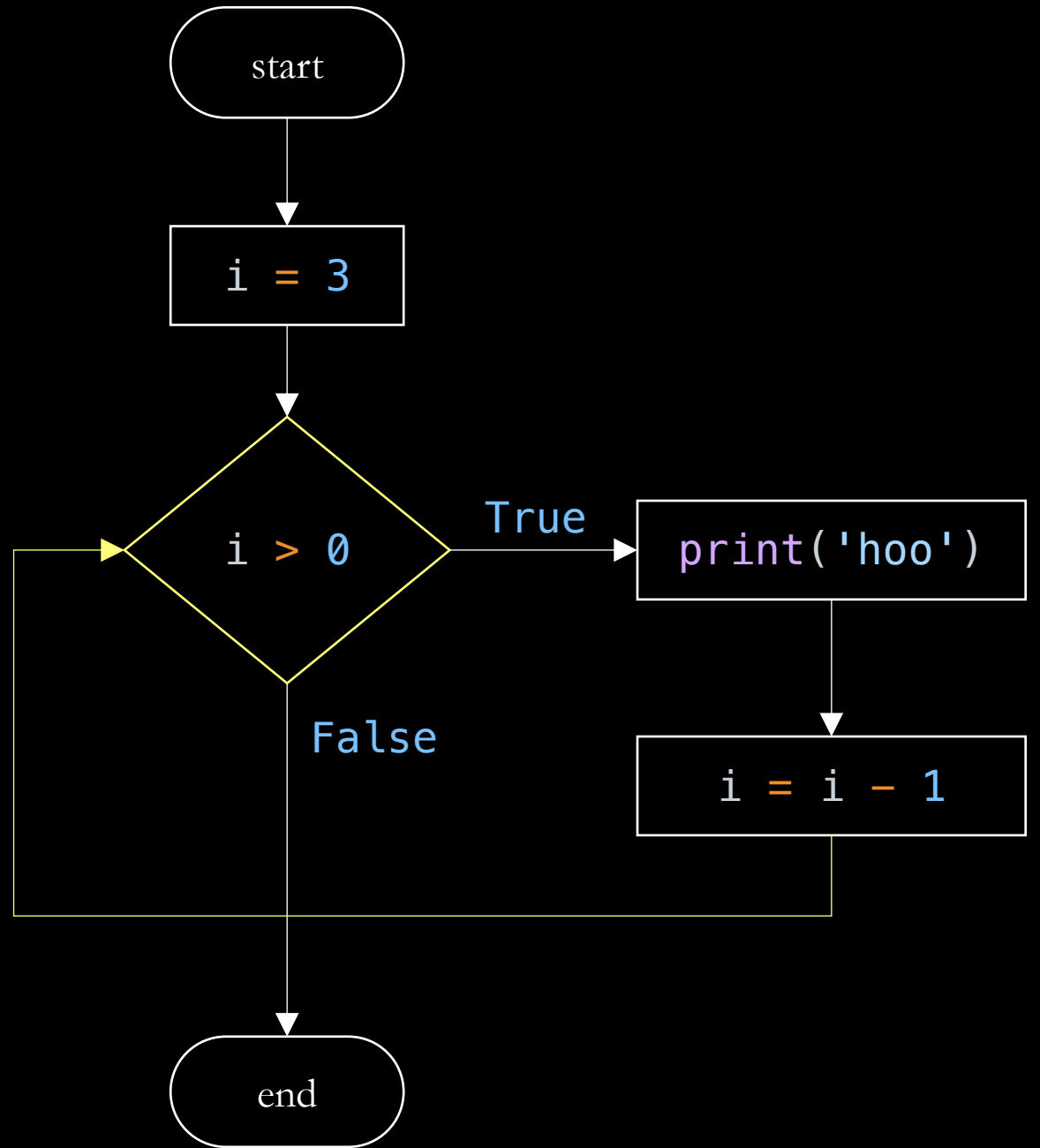
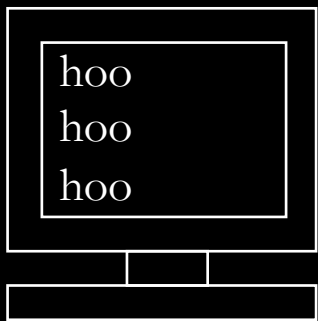
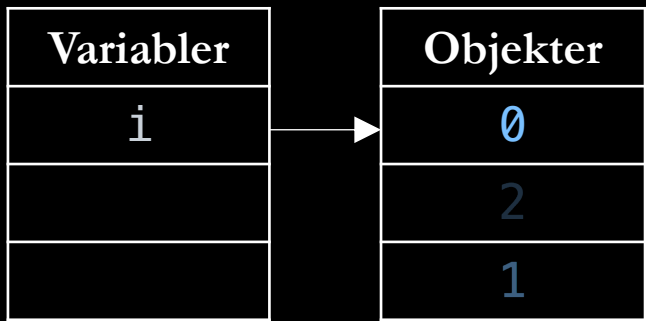
Objekter
3
2
1




```
i = 3
while i > 0:
    print('hoo')
    i = i - 1
```

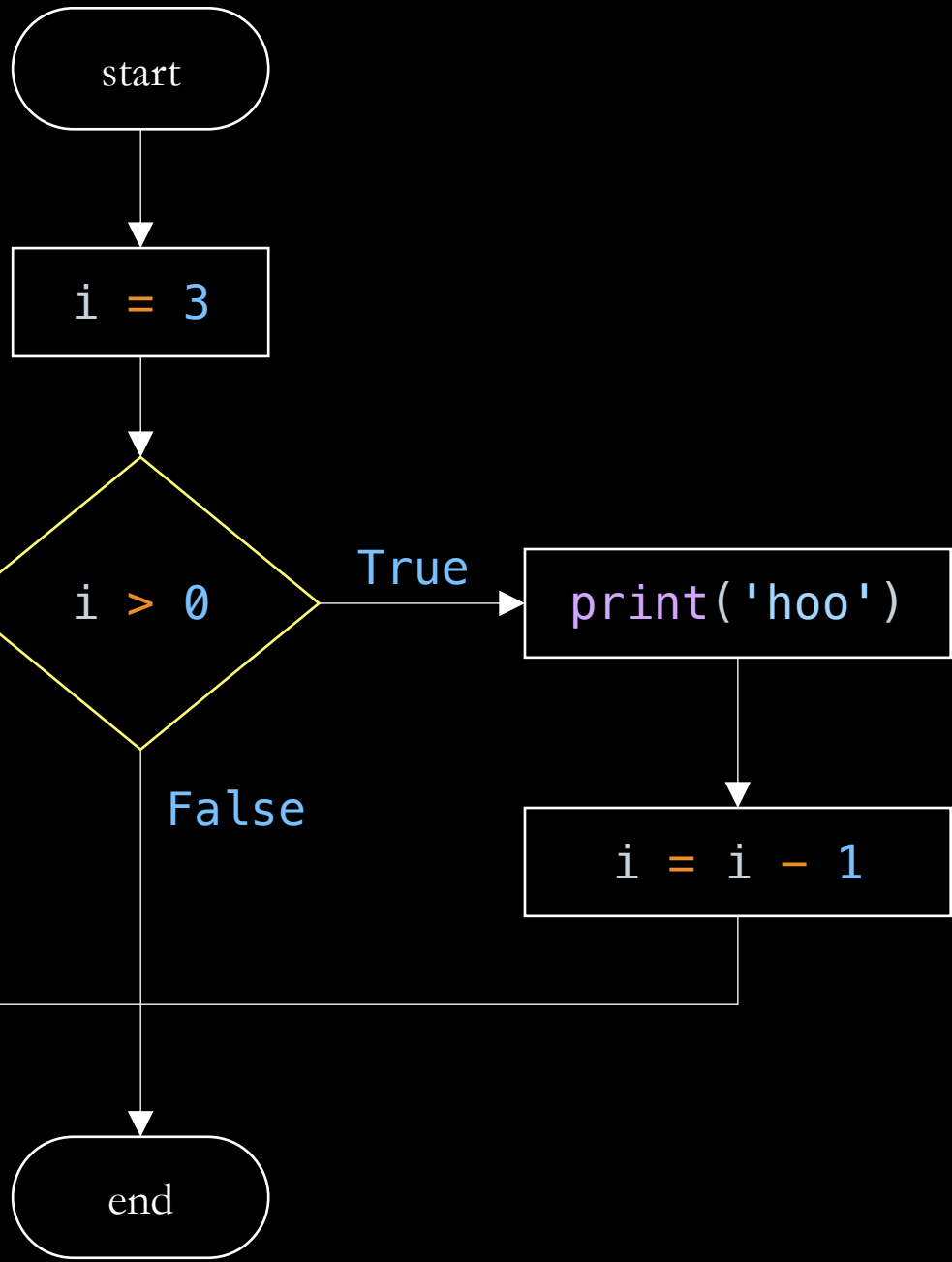


```
→ i = 3
while i > 0:
    print('hoo')
    i = i - 1
```

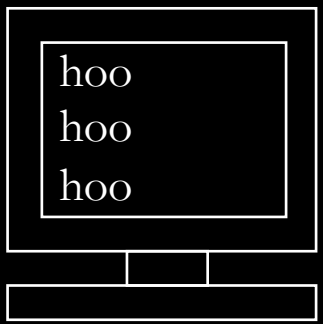


```
i = 3
while i > 0:
    print('hoo')
    i = i - 1
```

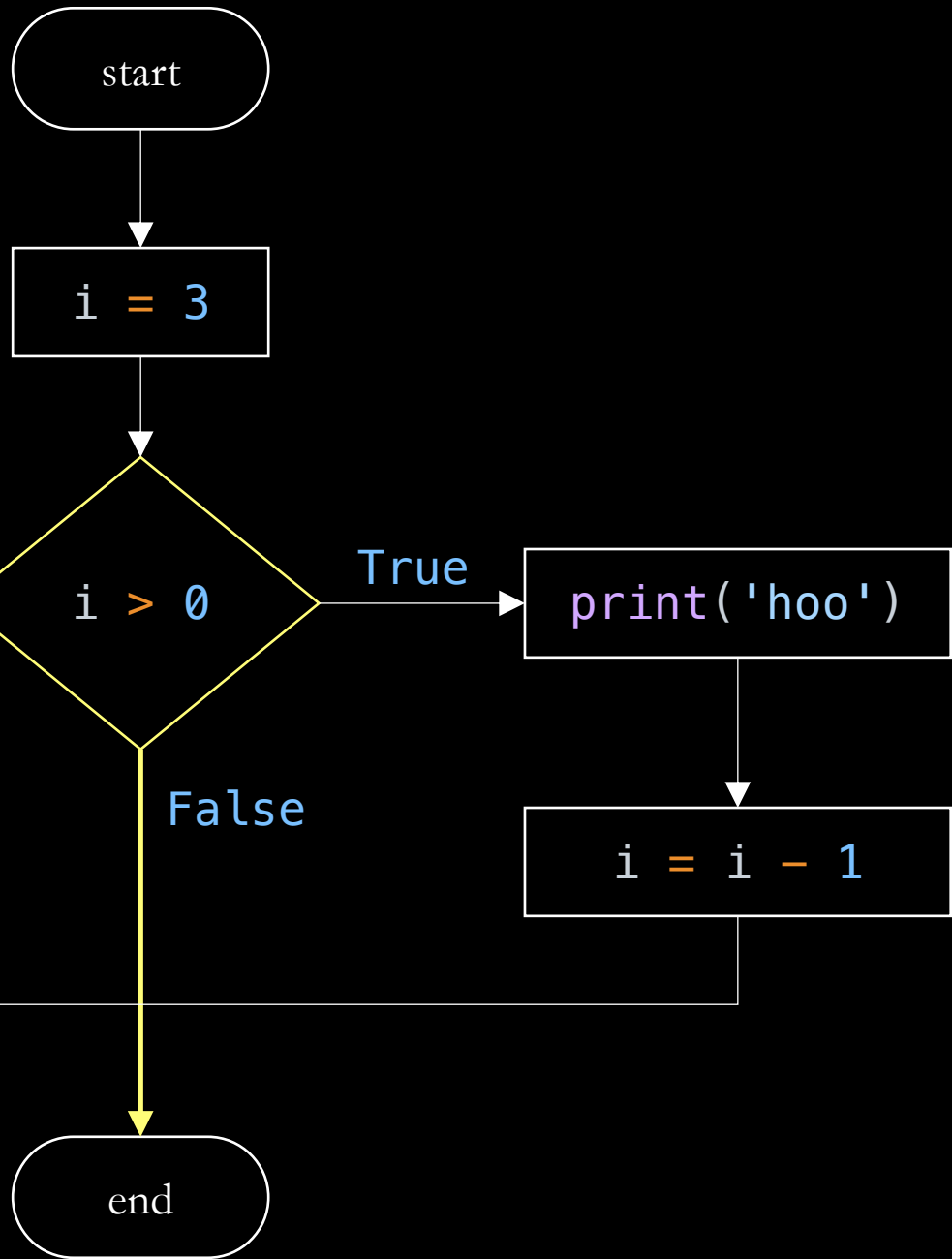
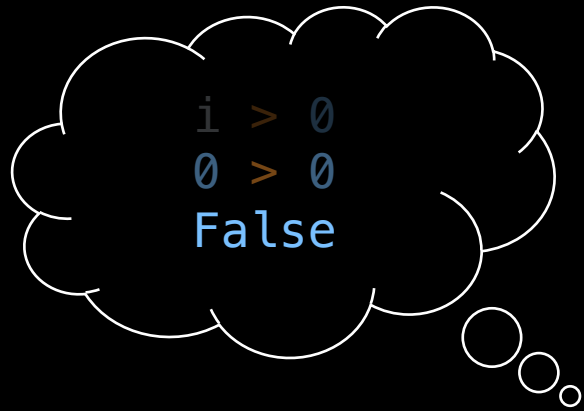
i > 0
0 > 0
False



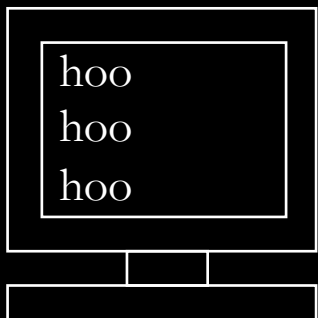
Variabler	Objekter
i	0
	2
	1



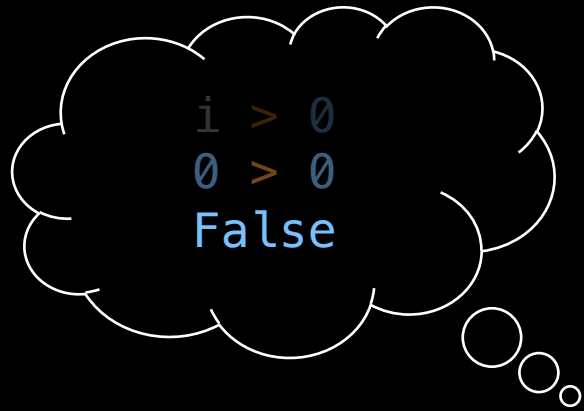
```
i = 3
while i > 0:
    print('hoo')
    i = i - 1
```



Variabler	Objekter
i	0
	2
	1



```
i = 3
while i > 0:
    print('hoo')
    i = i - 1
```



start

`i = 3`

`i > 0`

True

`print('hoo')`

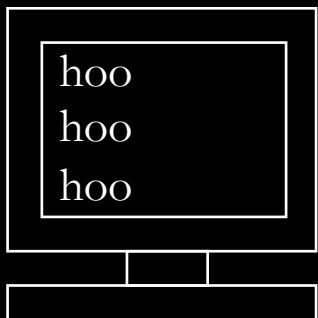
False

`i = i - 1`

end

Variabler
i

Objekter
0
2
1



```
i = 3
while i > 0:
    print('hoo')
    i = i - 1
```



```
i = 0
while i < 3:
    print('hoo')
    i += 1
```

```
i = 0
while i < 3:
    print('hoo')
    i += 1
```



```
for i in [0, 1, 2]:
    print('hoo')
```

```
for i in [0, 1, 2]:  
    print('hoo')
```



```
for i in range(3):  
    print('hoo')
```



```
for i in range(3):  
    print('hoo')
```



```
for _ in range(3):  
    print('hoo')
```

range(5)

0, 1, 2, 3, 4

```
range(3, 7)
```

```
3, 4, 5, 6
```

```
range(10, 20, 2)
```

```
10, 12, 14, 16, 18
```

```
range(20, 10, -2)
```

```
20, 18, 16, 14, 12
```

for

while

kan vare evig



kortere å skrive



mindre bugs

Oppgave:

Les input fra brukeren helt til du får noe du vil ha

break

avbryt løkken umiddelbart

continue

avbryt denne iterasjon av løkken

foo.py

```
from bar import add

num1 = int(input('num1 = '))
num2 = int(input('num2 = '))

ans = add(num1, num2)
print(f'Added nums: {ans}')
```

```
$ python foo.py
Testing add... OK
Testing multiply... OK
num1 = 2
num2 = 2
Added nums: 4
```

bar.py

```
def add(a, b):
    return a + b

print('Testing add... ', end='')
assert 4 == add(2, 2)
print('OK')

def multiply(a, b):
    return a * b

print('Testing multiply... ', end='')
assert 6 == multiply(2, 3)
print('OK')
```

foo.py

```
from bar import add

num1 = int(input('num1 = '))
num2 = int(input('num2 = '))

ans = add(num1, num2)
print(f'Added nums: {ans}')
```

```
$ python foo.py
num1 = 2
num2 = 2
Added nums: 4
```

bar.py

```
def add(a, b):
    return a + b

def test_add():
    print('Testing add... ', end='')
    assert 4 == add(2, 2)
    print('OK')

def multiply(a, b):
    return a * b

def test_multiply():
    print('Testing multiply... ', end='')
    assert 6 == multiply(2, 3)
    print('OK')

if __name__ == '__main__':
    test_add()
    test_multiply()
```

Oppgave:

Gitt et heltall x , hvor mange 2'ere er det i tallet?

Oppgave:

Hva er største faktor i tallet 209414732?

Oppgave:

Er tallet x et primtall?

Oppgave:

Hva er det n 'te primtallet?