

INF100 – 05

funksjoner

Crystal Chang Din
David Grellscheid

Funksjoner / functions

Expression / Uttrykk	Value / Verdi
<code>len("Hei")</code>	3
<code>math.sqrt(2)</code>	1,4142
<code>abs(-3.2)</code>	3,2
<code>max(3,7)</code>	7
<code>float("55")</code>	55,0

$$f : x \longrightarrow \sqrt{x}$$

$$f : x, y \longrightarrow 7x^3y^2$$

navn

input

output

Funksjoner / functions

$$f : x, y, z \longrightarrow 2 (x + y + z)$$

$$\text{doublesum} : x, y, z \longrightarrow 2 (x + y + z)$$

```
def doublesum(x,y,z):  
    return 2*(x+y+z)
```

```
abc = doublesum(5,7,1)  
print(abc)
```

Funksjoner / functions

```
def doublesum(x,y,z):  
    return 2*(x+y+z)
```

Funksjonsdefinisjon

```
abc = doublesum(5,7,1)  
print(abc)
```

Funksjonskall

```
xyz = doublesum(3,2,1)  
print(xyz)
```

Funksjonskall

dynamic typing

```
def doublesum(x,y,z):  
    sum = x+y+z  
    return 2*sum
```

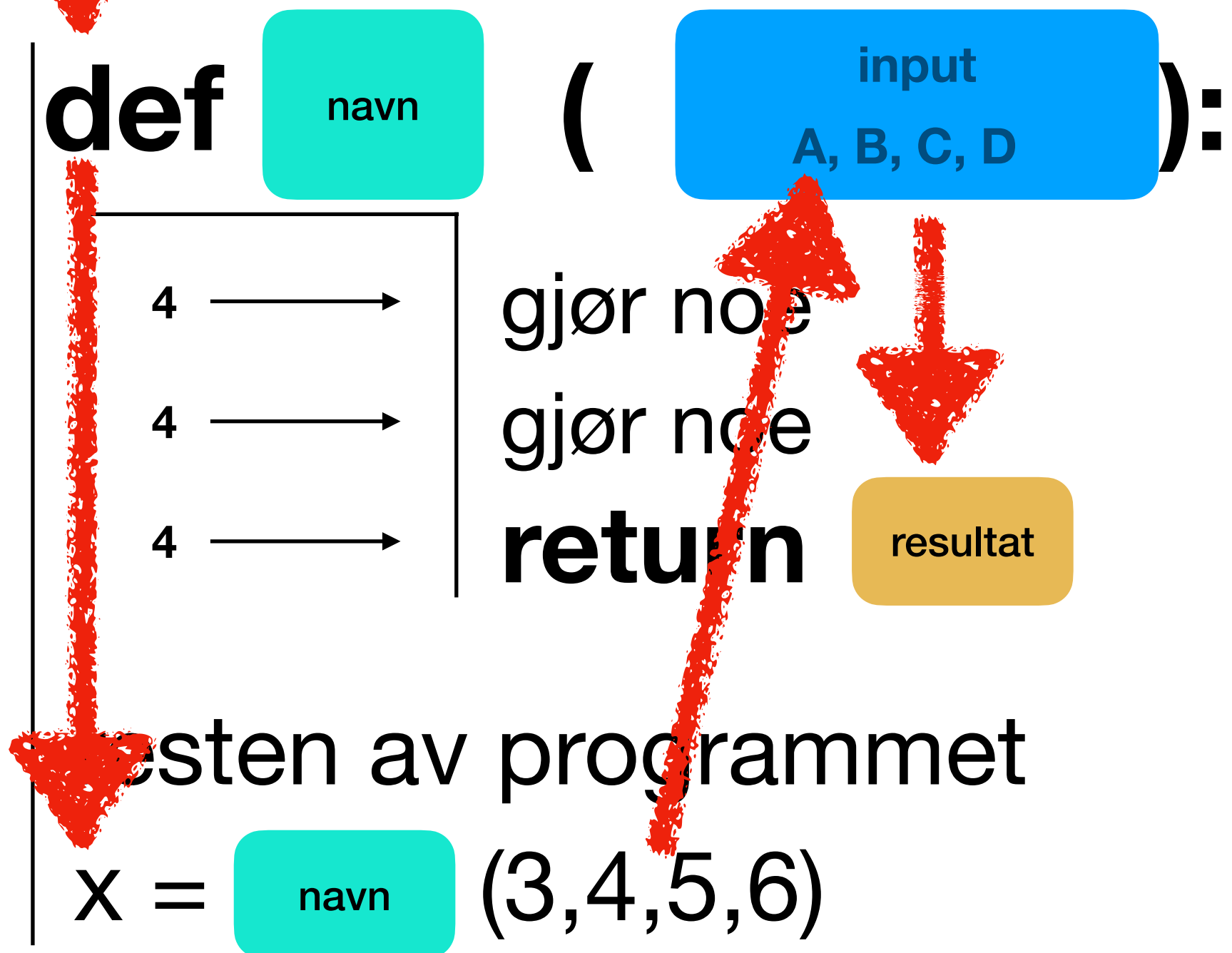
```
abc = doublesum(5,7,1)  
print(abc)
```

26

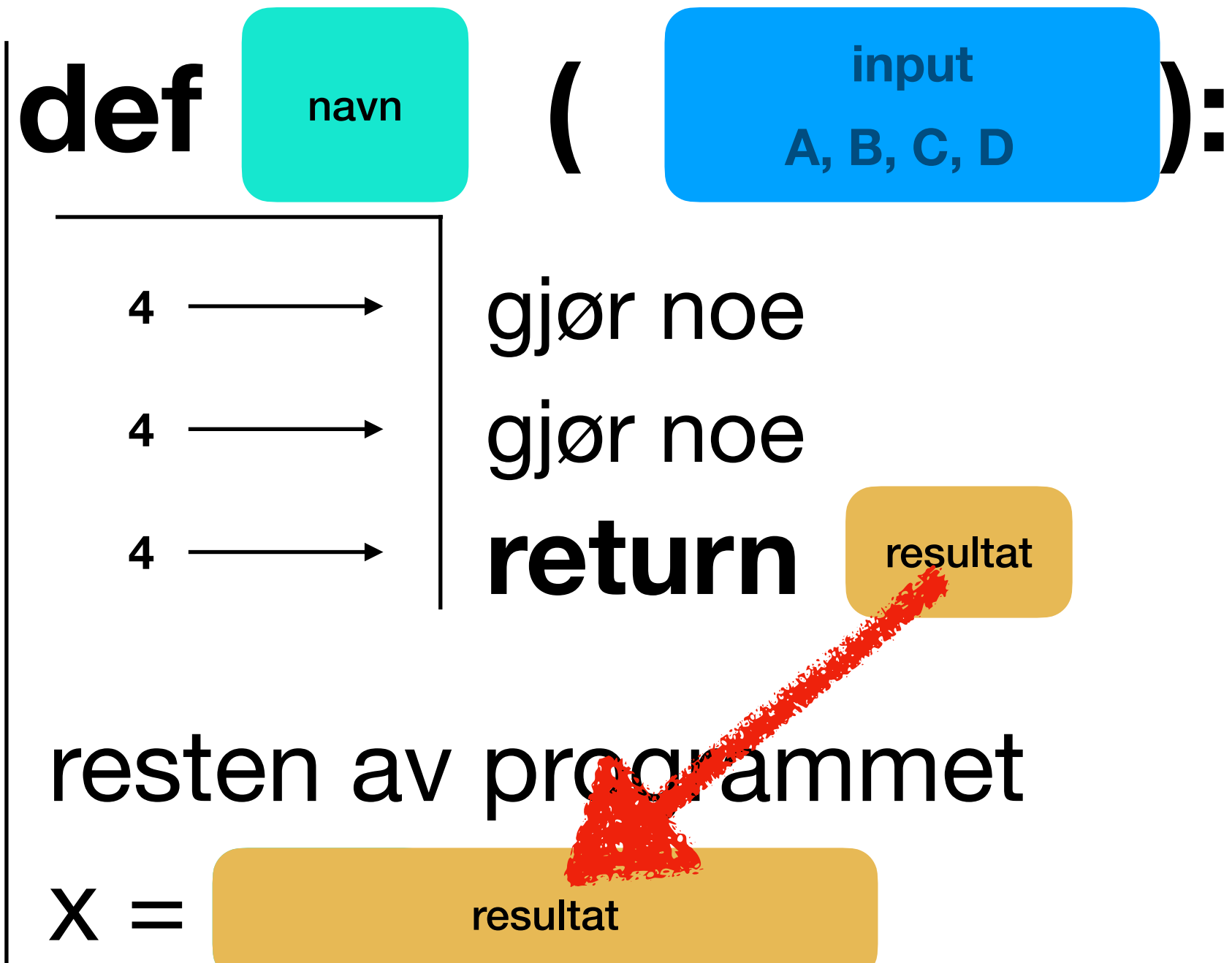
```
foo = doublesum('ab','xz','d')  
print(foo)
```

abxzdabxzd

programmflyt



def

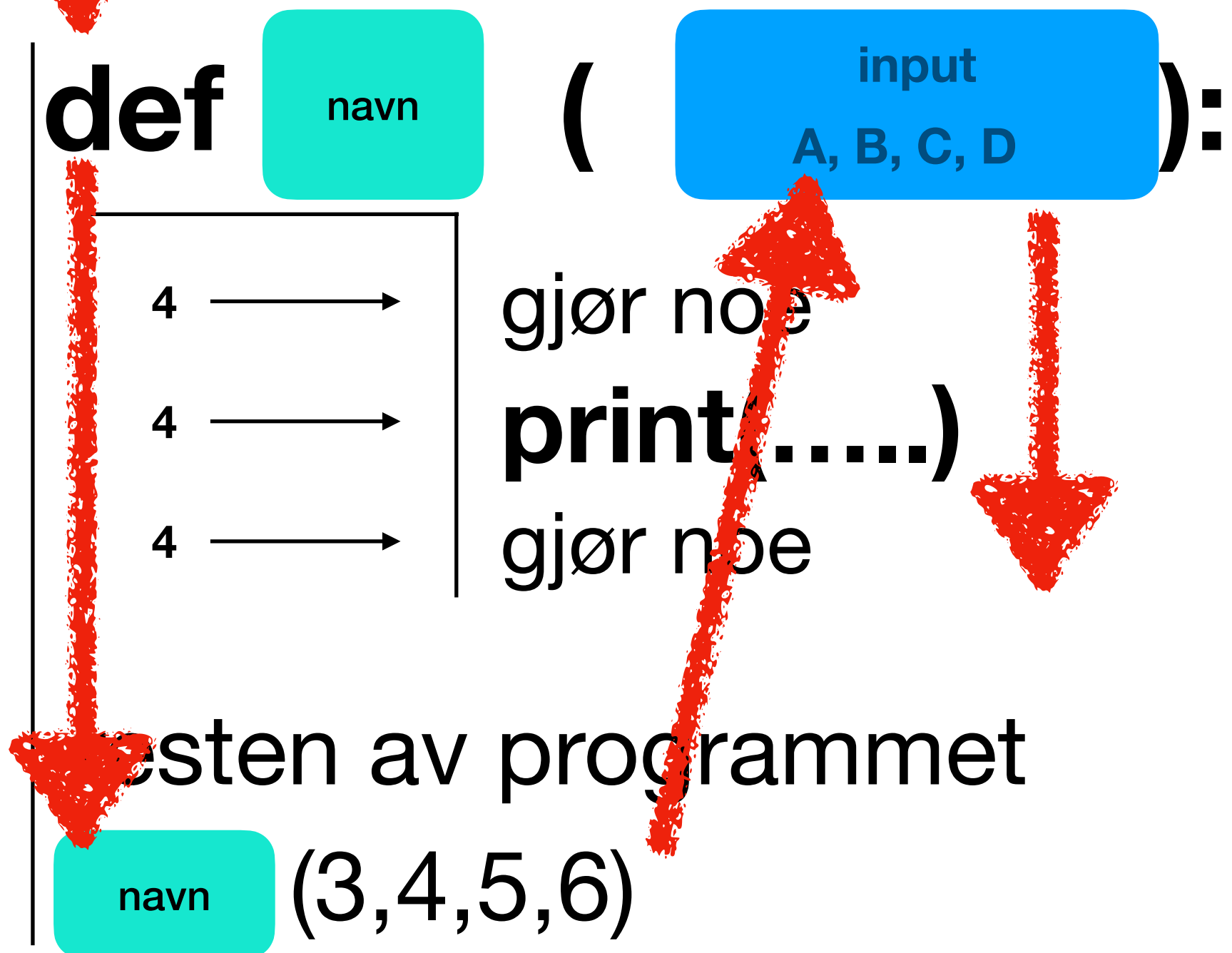


Vi kan bare bruke **return**-setningen i en funksjon.

I Python returnerer hver funksjon noe.

- Hvis **return**-setningen inneholder et uttrykk, evalueres den først og deretter returneres verdien.
- Hvis det ikke er noen **return**-setninger, returnerer funksjonen **None**.

programmflyt



def

def navn (input
A, B, C, D):

4 →

gjør noe

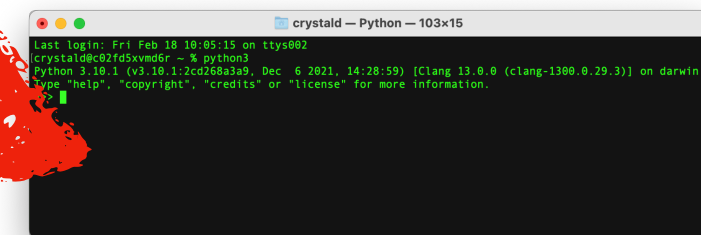
4 →

print(.....)

4 →

gjør noe

bivirkninger



resten av programmet

navn (3,4,5,6)
None



Vanligvis tilordner vi ikke utførelsesresultatet av en funksjon til en variabel når funksjonen ikke har en eksplisitt **return** setning.

bivirkninger / side effects

Expression / Uttrykk	side effect	Value / Verdi
len("Hei")		3
math.sqrt(2)		1,4142
abs(-3.2)		3,2
max(3,7)		7
float("55")		55,0
print(text)	skriv text til skjermen	None
input(qn_text)	skriv qn_text, vent for svar	string fra bruker

print eller return?

```
def doublesum(x,y,z):  
    sum = x+y+z  
    print(sum)  
    return 2*sum
```

```
x = doublesum(3,6,7)
```

Expression / Uttrykk	side effect	Value / Verdi
doublesum(3,6,7)	skriv 16 til skjermen	32
x		32

Scope

```
def doublesum(x,y,z):  
    tmp = x+y+z  
    return 2*tmp
```

```
x = 5  
tmp = 7
```

```
ds = doublesum(1,1,1)
```

```
print(x,tmp,ds)
```

5 7 6

lokale variabler, kun i funksjonen

globale variabler

```
def doublesum(x,y,z):  
    global tmp  
    tmp = x+y+z  
    return 2*tmp
```

```
x = 5  
tmp = 7
```

```
ds = doublesum(1,1,1)
```

```
print(x,tmp,ds)
```

5 3 6

```
def add_up(first, last):  
    if first > last:  
        return -1  
    else:  
        sum = 0  
        for x in range(first, last+1):  
            sum += x  
        return sum
```

positional argument

```
abc = add_up(1, 5)  
print(abc)
```

15

```
abc = add_up(5, 1)  
print(abc)
```

-1

keyword argument

```
abc = add_up(first=1, last=5)  
print(abc)
```

15

```
abc = add_up(last=5, first=1)  
print(abc)
```

15

positional argument keyword argument

```
abc = add_up(1, last=5)  
print(abc)
```

15

default argument



```
def add_up(first, last=5):  
    if first > last:  
        return -1  
    else:  
        sum = 0  
        for x in range(first, last+1):  
            sum += x  
        return sum
```

```
abc = add_up(1)  
print(abc)
```

15

```
abc = add_up(1, 6)  
print(abc)
```

21

```
abc = add_up(1, last=6)  
print(abc)
```

21