

Python vs plokkskeem

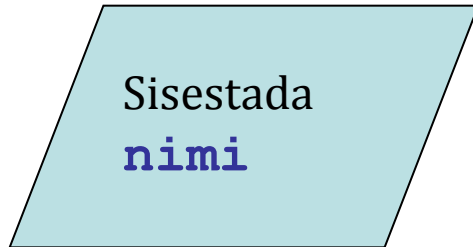
Programmeerimine
7. tahvlipraktikum

Tänases tahvlipraktikumis

- Pythoni konstruktsioonid plokk skeemina

Sisestamine ja väljastamine

```
nimi = input("Sisesta nimi: ")
```

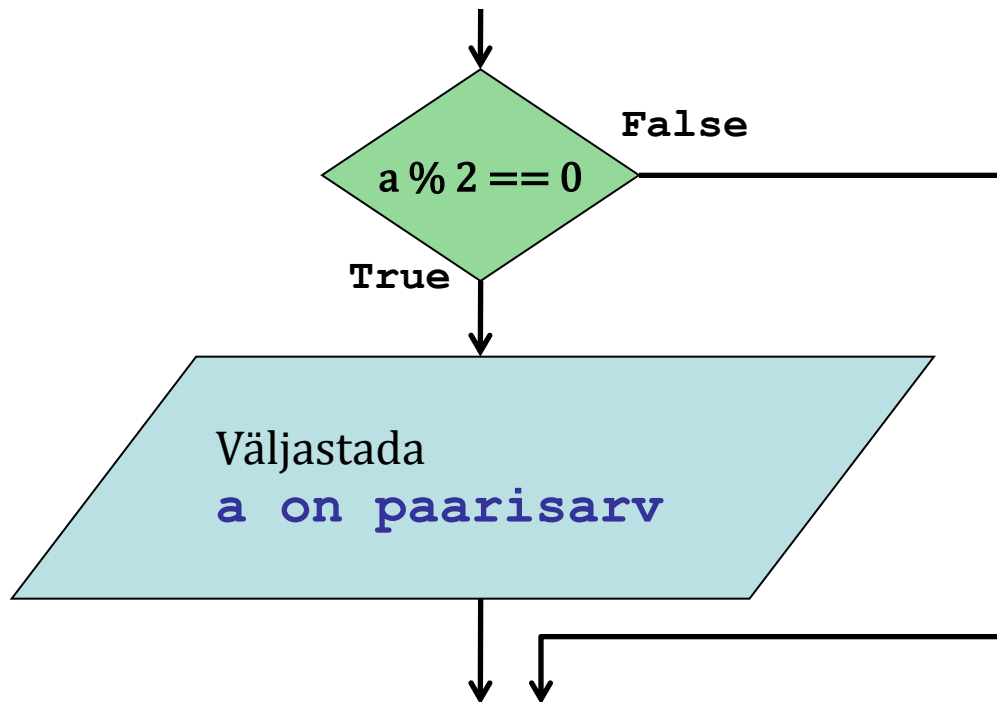


```
print(nimi)
```



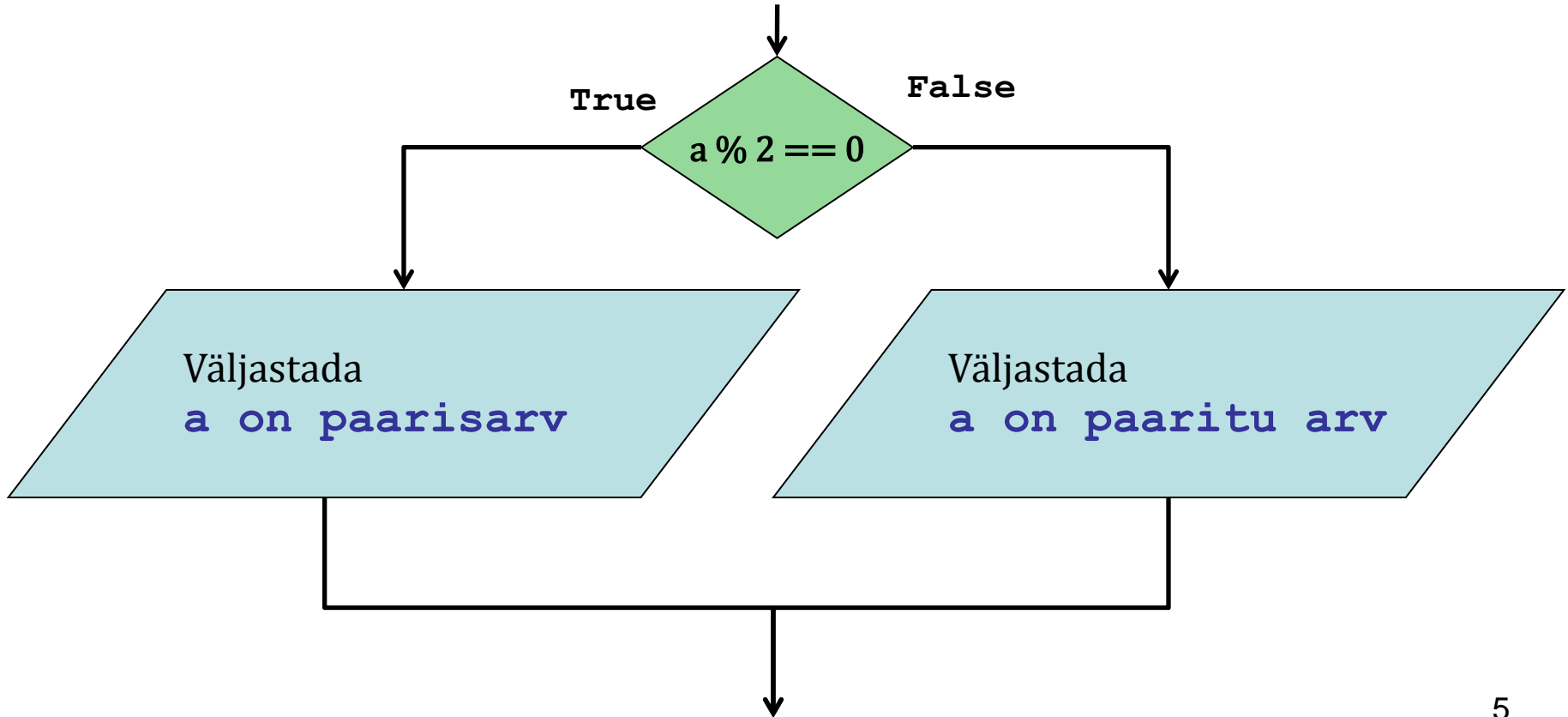
Tingimuslaused

```
if a % 2 == 0:  
    print(str(a) + " on paarisarv")
```



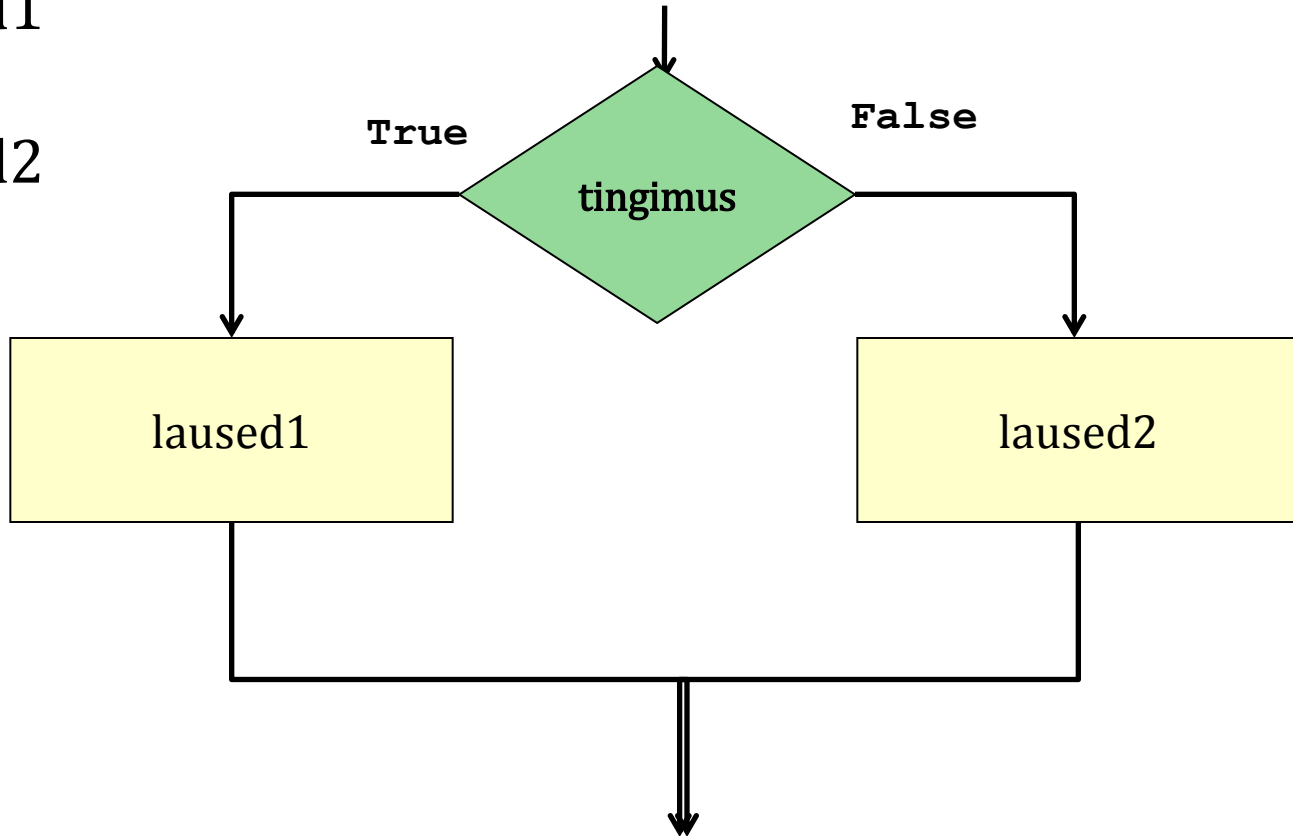
Tingimuslaused

```
if a % 2 == 0:  
    print(str(a) + " on paarisarv")  
else:  
    print(str(a) + " on paaritu arv")
```



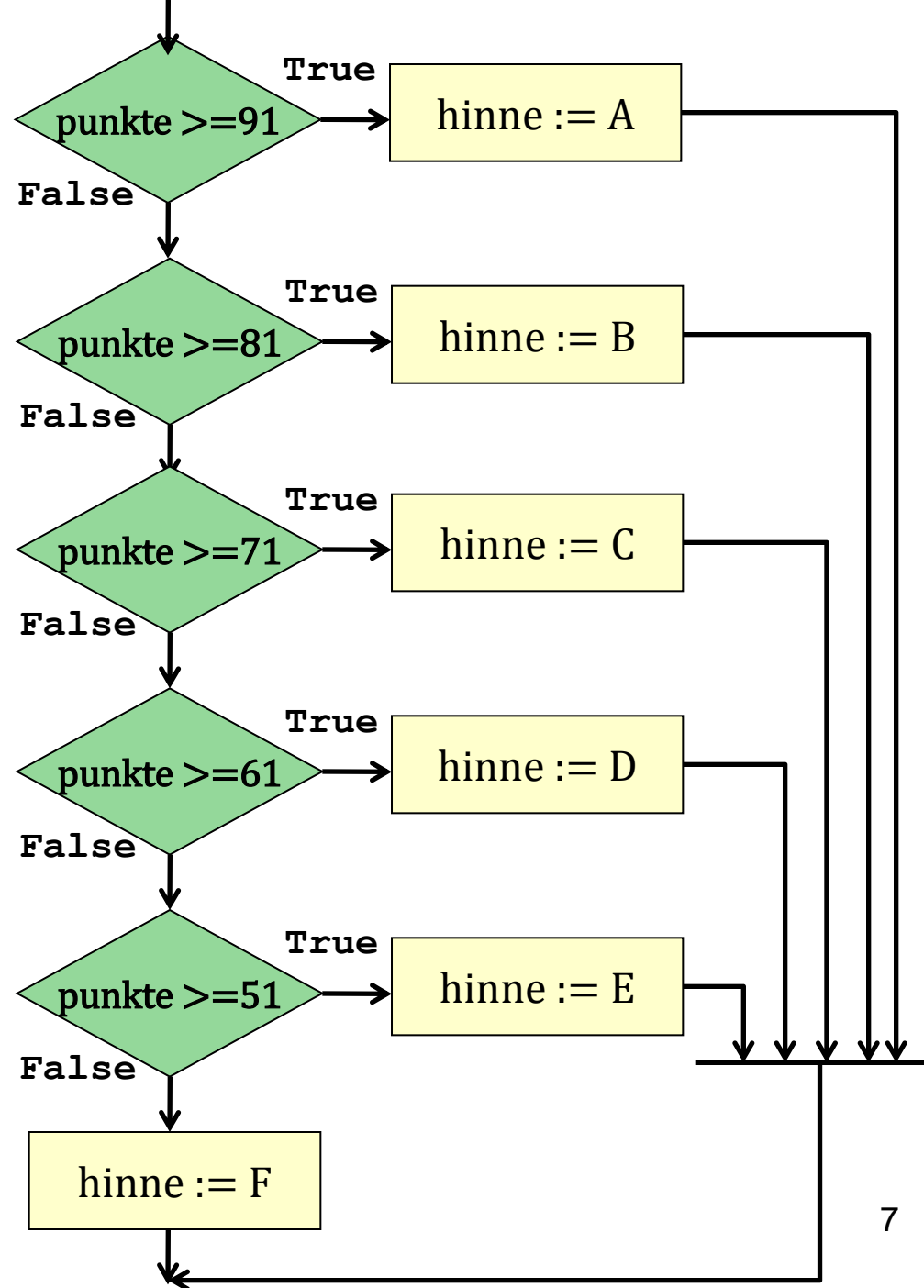
Tingimuslaused

```
if tingimus:  
    laused1  
else:  
    laused2
```



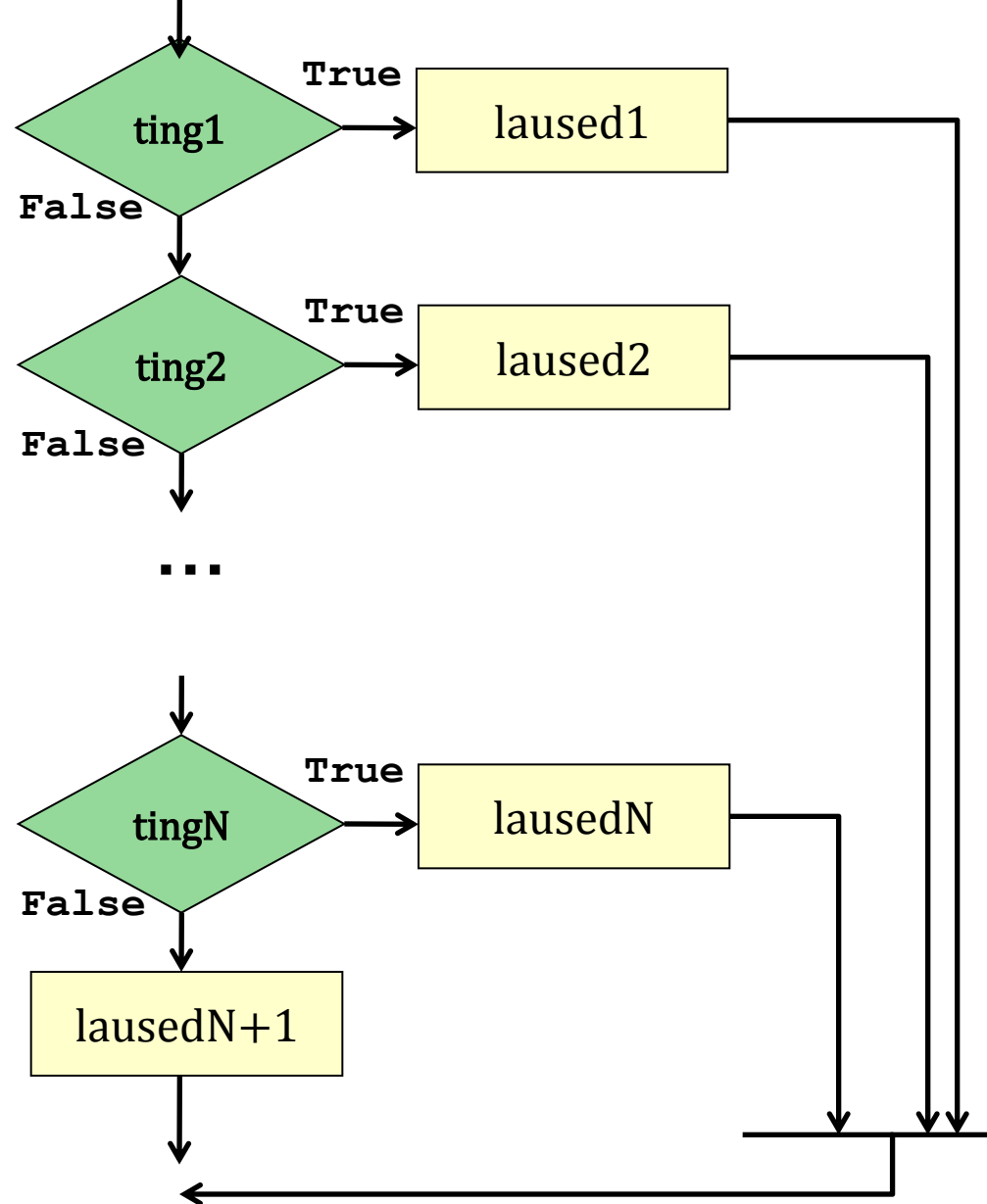
Tingimuslaused

```
if punkte >= 91:  
    hinne = 'A'  
elif punkte >= 81:  
    hinne = 'B'  
elif punkte >= 71:  
    hinne = 'C'  
elif punkte >= 61:  
    hinne = 'D'  
elif punkte >= 51:  
    hinne = 'E'  
else:  
    hinne = 'F'
```



Tingimuslaused

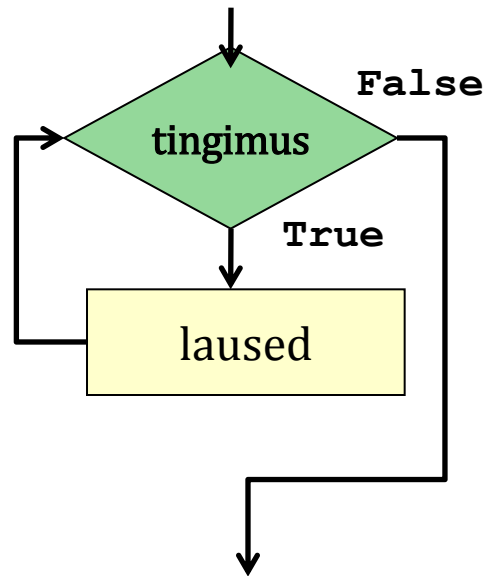
```
if ting1:  
    laused1  
elif ting2:  
    laused2  
...  
elif tingN:  
    lausedN  
else:  
    lausedN+1
```



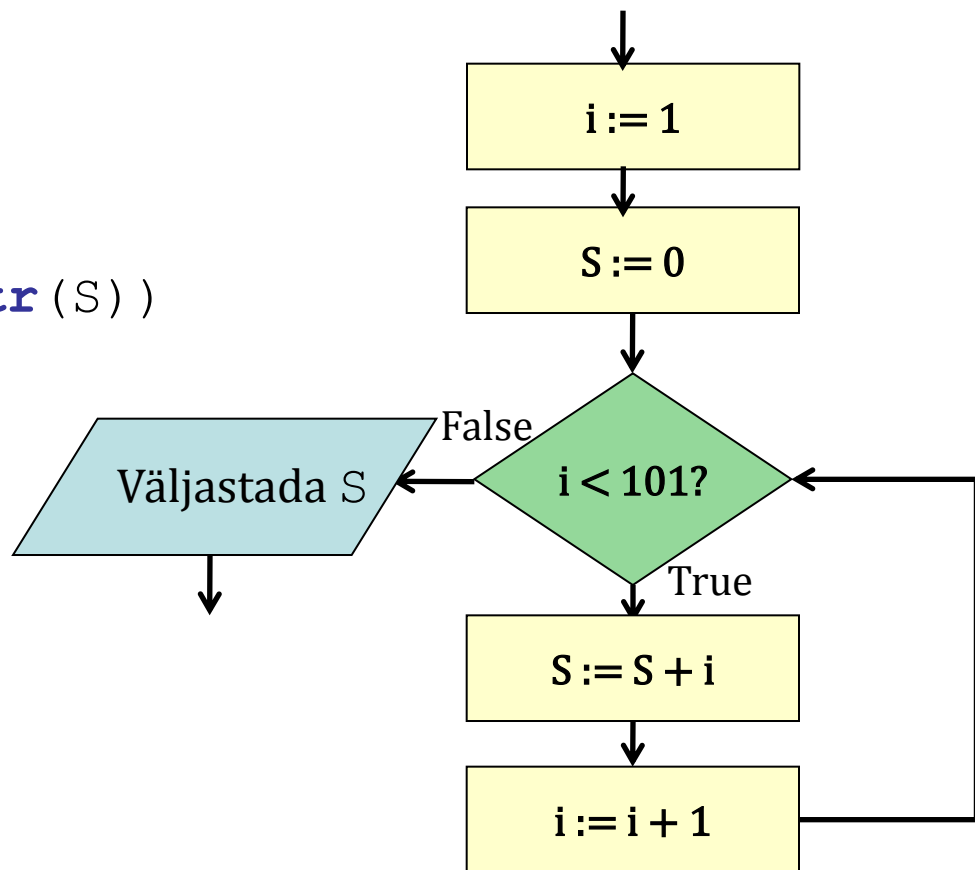
Tsükliid

- Eelkontrolliga tsükkel

```
while tingimus:  
    laused
```

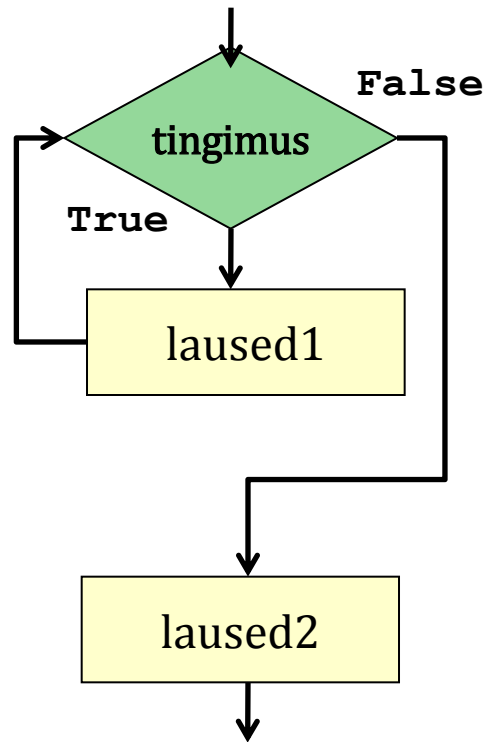


```
i = 1
S = 0
while i < 101:
    S = S + i
    i = i + 1
print("Summa= " + str(S))
```



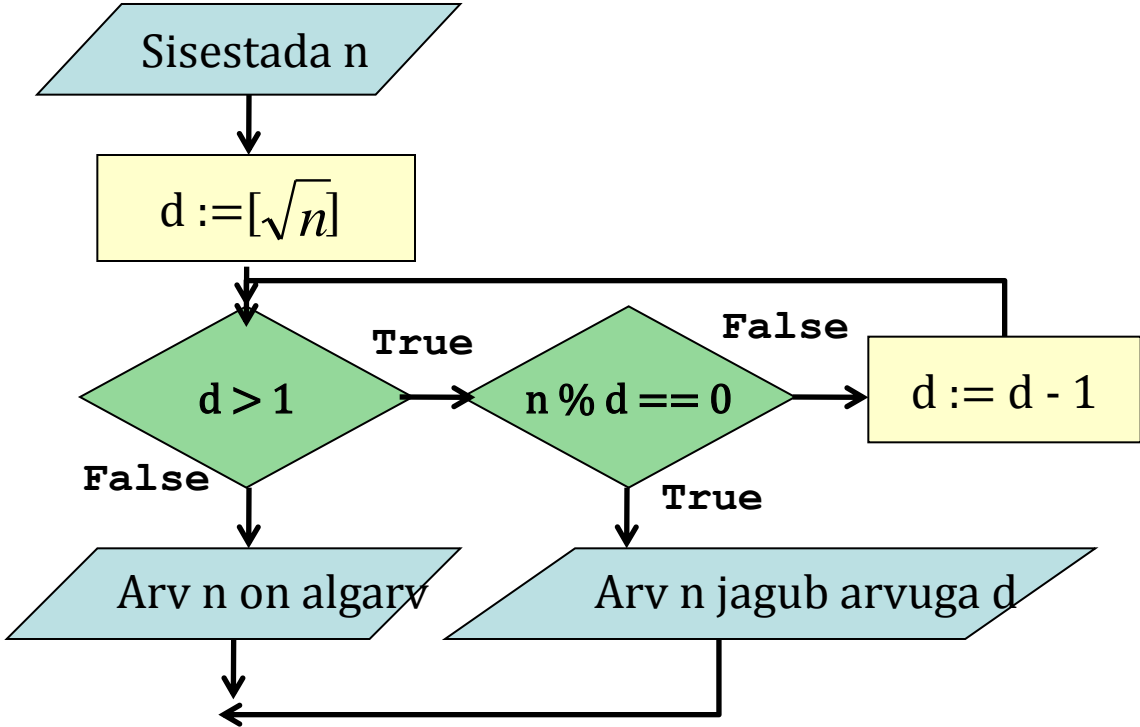
■ Eelkontrolliga tsükkel - üldkuju

```
while tingimus:  
    laused1  
else:  
    laused2
```



Algarvu kontroll

```
import math
n = int(input("Sisesta arv: "))
d = int(math.sqrt(n))
while d > 1:
    if n % d == 0:
        print("Arv", n, "jagub arvuga", d)
        break
    d = d - 1
else:
    print("Arv", n, "on algarv!")
```



Demo Debug

- Määratud tsükkel

Jada genereerimine: `range(start, stop, step)`,

`range(6) => [0, 1, 2, 3, 4, 5]`

`range(2, 5) => [2, 3, 4]`

`range(3, 7, 2) => [3, 5]`

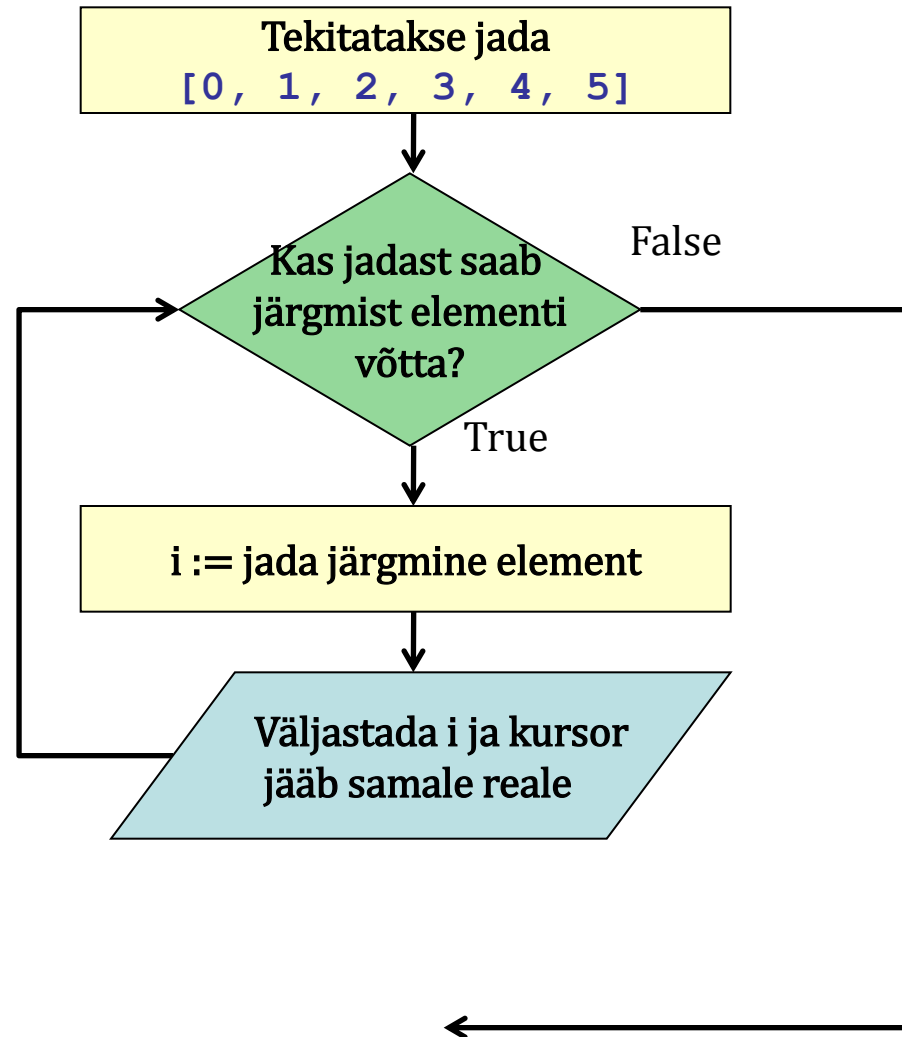
`range(15, 3, -4) => [15, 11, 7]`

```
for i in range(6):  
    print(i, end = " ")
```

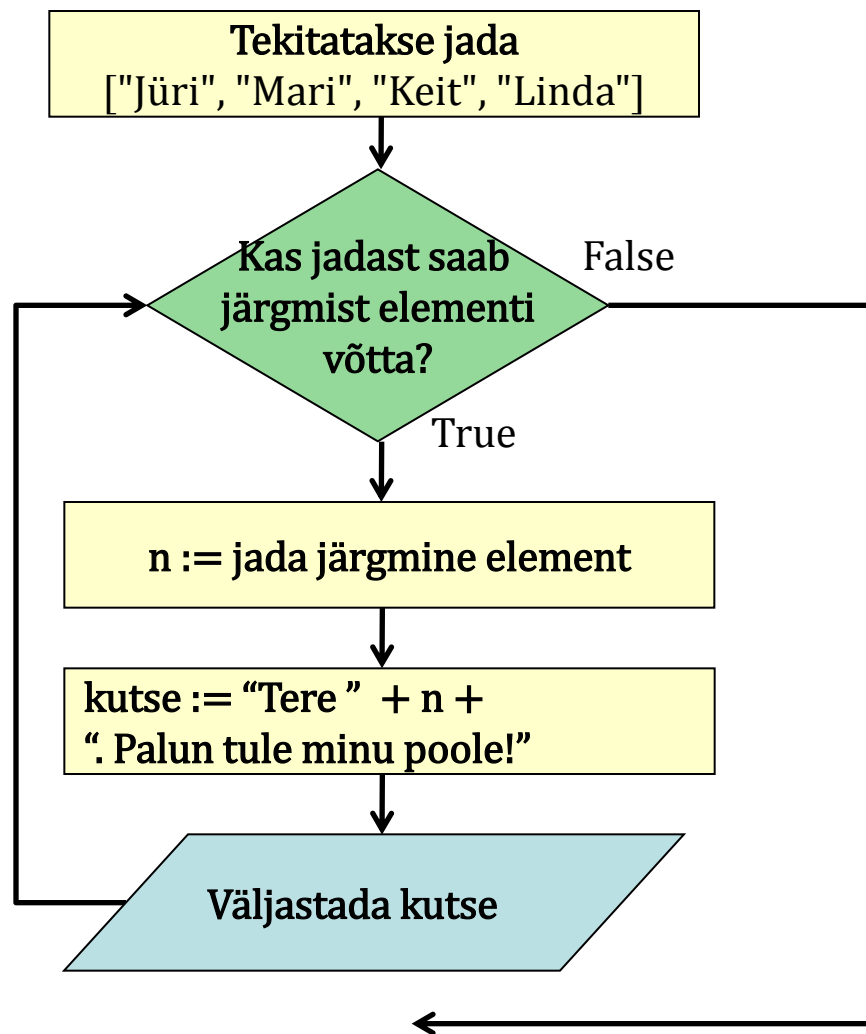
vt. loeng 4 sl 10-24

```
for i in range(2, -2):  
    print(i, end = " ")
```

`range(2, -2) => []`

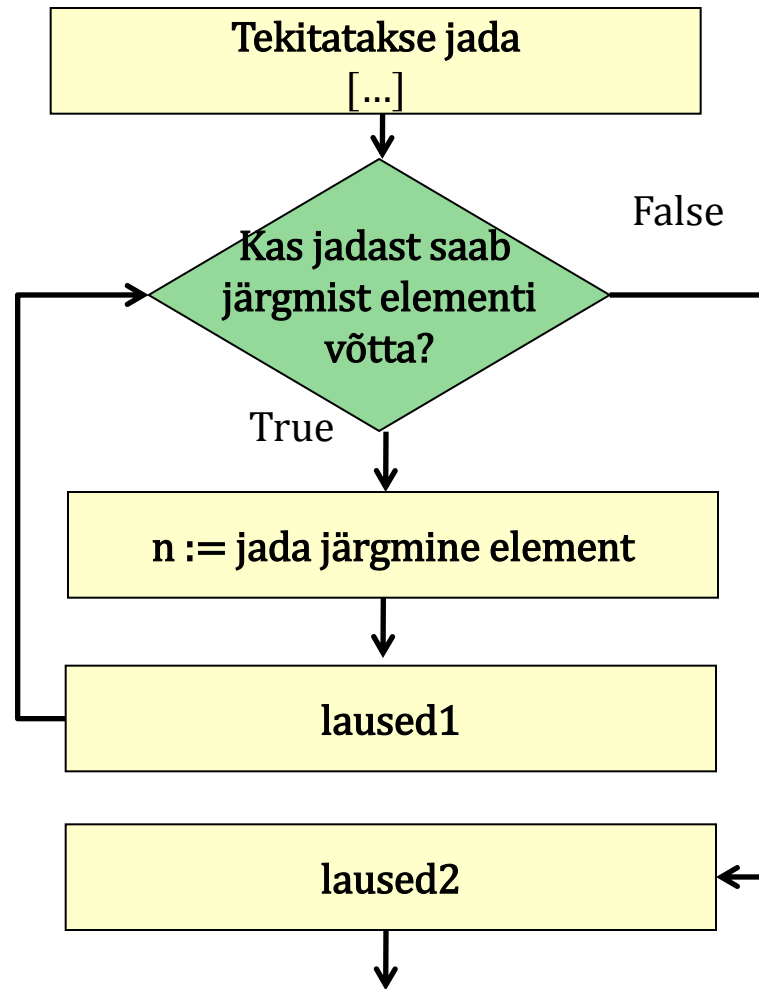


```
for n in ["Jüri", "Mari", "Keit", "Linda"]:  
    kutse = "Tere " + n + ". Palun tule minu poole!"  
    print(kutse)
```



- Määratud tsükkel
- - üldkuju

```
for i in <jada>:  
    laused1  
else:  
    laused2
```




```
ucmd = input("Sisesta käsk:")  
for cmd in ["insert", "delete", "update"]:  
    if cmd == ucmd:  
        valid = 1  
        break  
else:  
    valid = 0  
if valid == 1:  
    print(ucmd + " on kehtiv käsk")  
else:  
    print(ucmd + " ei ole kehtiv käsk")
```

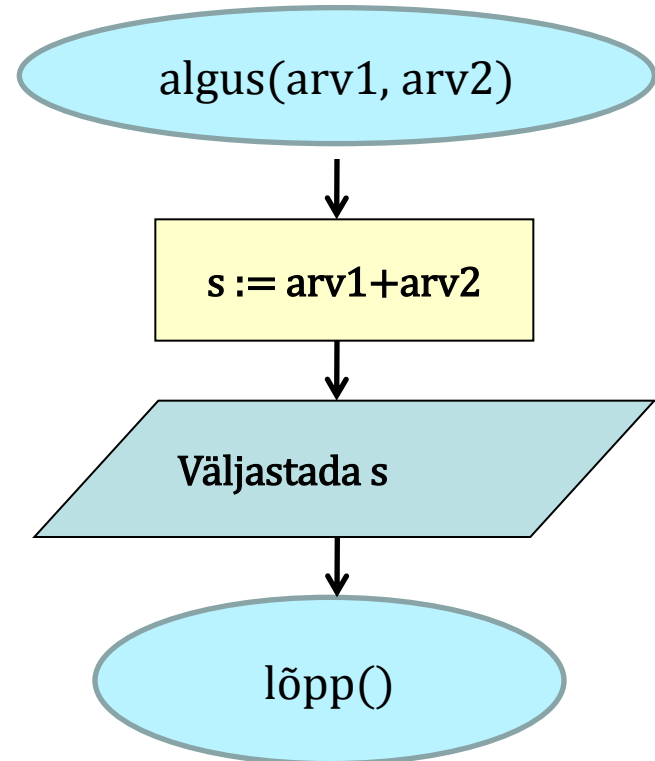
Funktsioonid - I

```
def väljastaSumma(arv1, arv2):  
    s = arv1 + arv2  
    print("Summa= " + str(s))
```

Pöördumine:

väljastaSumma(3, 8)

väljastaSumma(arv1, arv2)

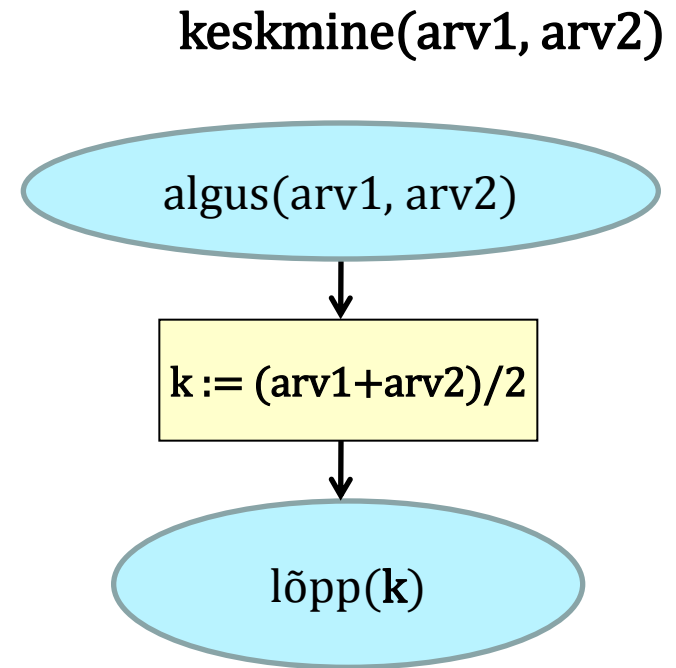


Funktsioonid - II

```
def keskmine(arv1, arv2):  
    k = (arv1 + arv2) / 2  
    return k
```

Pöördumine:

```
kesk = keskmine(3, 8)
```



Vt. loeng 3 sl 27-34

Ülesanded

1. Esitada järgmine programmilõik plokk skeemina.

```
j = 6
while j != 0 :
    print(j)
    if j % 2 == 0 :
        j = j - 1
    else :
        j = j + 1
j = -j
```

Mida väljastab ekraanile see programmilõik?

2. Esitada järgmine programmilõik plokk skeemina.

```
i = 15
j = 6
while j != 0:
    print(i)
    if j % 2 == 0:
        j = j - 1
    else :
        j = j + 1
    j = -j
    i = i + 2 * j
```

Mida väljastab ekraanile see programmilõik?

3. Collatzi hüpotees

Antud on naturaalarv $n > 1$. Kui n on paarisarv, siis jagatakse ta 2-ga. Kui n on paaritu arv (ja ei võrdu ühega), siis asendatakse ta arvuga $3n+1$.

Näide. Valime arvu 7:

7, 22, 11, 34, 17, 52, 26, 13, 40, 20, 10, 5, 16, 8, 4, 2, 1

Seni tõestamata hüpoteesi kohaselt lõpeb see protsess ükskõik millisest arvust lähtudes alati arvuga 1.

Koostada plokk skeem funktsiooni jaoks, mis kontrollib seda hüpoteesi etteantud naturaalarvu n puhul. Väljastada arv n ja iga sammu vahetulemused. Koostada ka vastav funktsioon Pythonis.

4. Lohe

Kangelane võitleb lohega, kellel on n pead.

Enne lööki viskab kangelane täringut ning raiub lohel maha nii mitu pead, kui palju silmi täringul tuli. Seejärel viskab lohe täringut ning vastavalt silmade arvule kasvab tal nii palju uusi päid asemele. Järgmise löögi eel viskab kangelane uuesti täringut jne. Võitlus kestab hetkeni, mil kangelasel õnnestub ühekorraga maha lüüa lohe kõik järelejäänud pead, pärast mida enam uusi päid juurde ei kasva.

Koostada plokskeem funktsiooni jaoks, mis leiab, kui mitme löögiga saab kangelane lohest jagu. Koostada ka vastav funktsioon Pythonis.

5. Kaardid

Kaardipakis on n kaarti. Need jagatakse kolme mängija vahel järgmisel viisil: esimesele üks, teisele kaks, kolmandale üks, esimesele kaks, teisele üks, kolmandale kaks jne, st igal jagamisel antakse vaheldumisi üks kaart või kaks kaarti. Koostada plokk skeem funktsiooni jaoks, mis leiab, mitu kaarti iga mängija saab. Koostada ka vastav funktsioon Pythonis.

Näide. Olgu $n = 11$.

1. mängija	1			2			1		4
2. mängija		2			1			1	4
3. mängija			1			2			3
	1	2	3	4	5	6	7	8	

Numeroloogia

Arv on positiivne täisarv.

Vra on arvu numbrite järjekorra vahetamisel saadud arv.

Näiteks arvu 2376 vra on 6732.

Vraeerimine on arvu liitmine oma vraga.

palindroom

$$\begin{array}{r} 2376 \\ + 6732 \\ \hline 9108 \\ + 8019 \\ \hline 17127 \\ + 72171 \\ \hline \underline{89298} \end{array}$$

6. Vraerimine

- Koostada plokk skeem funktsiooni jaoks, mis muudab etteantud naturaalarvu n numbrite järjestuse vastupidiseks. Koostada ka vastav funktsioon Pythonis.
- Koostada plokk skeem funktsioonile, mis kontrollib, kas etteantud arv on palindroom. Koostada ka vastav funktsioon Pythonis.
- Koostada plokk skeem funktsiooni jaoks, mis etteantud naturaalarvu n korral leiab vraerimiste arvu, mille järel arv muutub palindroomiks. Koostada ka vastav funktsioon Pythonis.