



- Print - Lause vai funktio?
 - Print on python2.7:ssa lause, 3:ssa funktio
 - Miten muille funktioille annetaan parametreja? Tulee ensi viikolla

Palaute



- Luennolla liian hidas tahti
- Luennolla käytiin asiat liian nopeasti läpi
 - Tulkitsen tämän niin, että tahti oli sopiva
- Lainausmerkit?
 - Lainausmerkeillä merkitään aina merkkijonoa. Vain tavalliset (shift-2) lainausmerkit ovat syntaksin mukaisia.

Palaute



Lisää haasteita? Digitaalinen esterata.
Löytyy sivulta ville-
mobile.utu.fi/salaseura

Joka viikon luennoissa lisää vihjeitä
etenemiseen.

Kurssin jälkeen esim. Advent of code.
Myös itse voi keksiä itselleen
koodausongelmia

Palaute



Selkeämmät tehtävänannot

- Tehtävänannot tarkoituksella hieman epäselvät. Ohjelmoijan työssä 'tehtävänannot' sitten vieläkin epäselvemmät.
- Lisäksi kannustetaan kokeilemaan (ja sietämään epävarmuutta)

Palaute

Toisto



Turun yliopisto
University of Turku



- Kertaus
- Ohjausrakenne: toisto
 - Määrätty toisto (Definiitti)
 - Määräämätön toisto (Indefiniitti)
- Livekoodaus

Hyvin yksinkertaistettuna: Loogisia lausekkeita

A	B	A and B
1	1	1

Kertaus: Boolean algebra

Lause

Aiheuttaa muutoksen
ohjelman tilassa

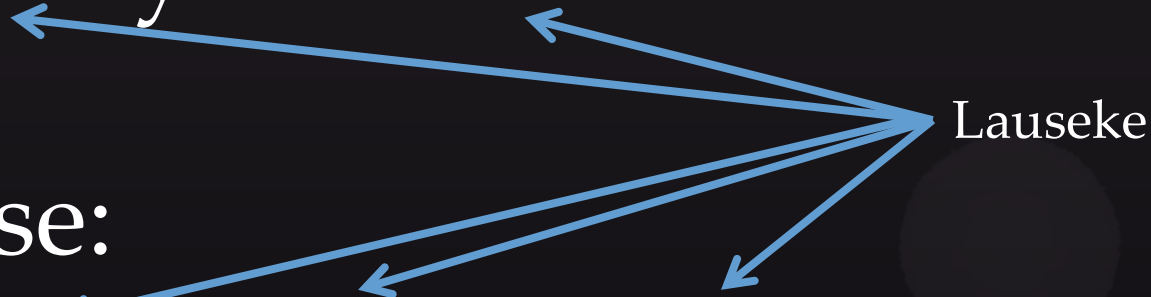
Lauseke

Arvo ohjelman
tilassa

Kertaus: Lause/Lauseke

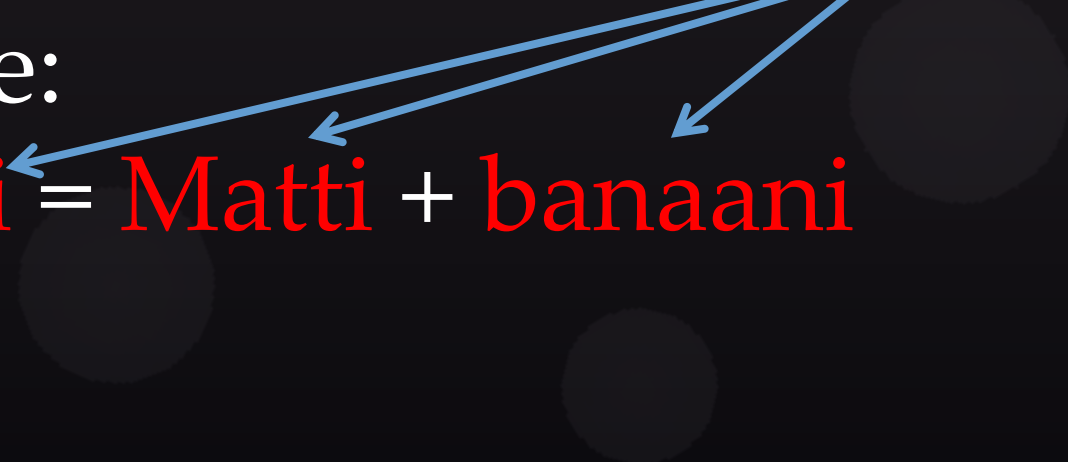
Lause:

Matti syö banaanin



Lause:

Matti = Matti + banaanin



Kertaus: Lause/Lauseke



Toisto

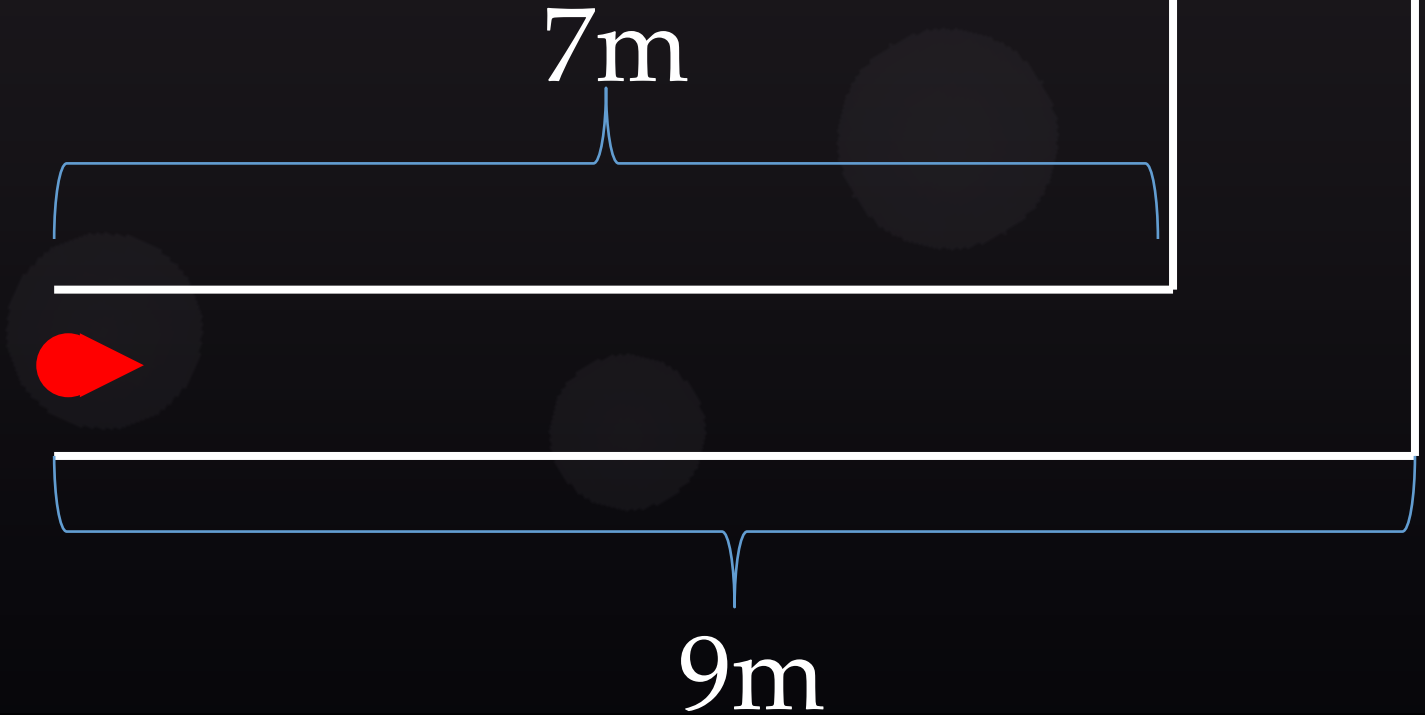
Nimiä

- Toisto(rakenne)
- Silmukka
- Looppi
- Iterointi

Toisto

Voit

- Ottaa askeleen (1m)
- Kääntyy oikealle/vasemmalle
- Katsoa, voitko ottaa askeleen



Toisto

```
if (onkoSeina) :
```

```
    otaAskel
```

```
else:
```

```
    käännyVasemmalle
```

```
if (onkoSeina) :
```

```
    otaAskel
```

```
else:
```

```
    käännyVasemmalle
```

```
#Ylläoleva vielä 4 kertaa kopioituna....
```

Toisto

Toistolla haetaan

- yleisyyttä
- saman asian uudelleenkirjoittamisen välttämistä
- ABSTRAHOINTIA

Toisto

Definiitti toisto
(määrätty)

Indefiniitti toisto
(määräämätön)

Toisto

Definiitin toiston
määrä on vakio ja
tiedetään ennen
ohjelman suoritusta

Toisto: Definiitti

Jokaiselle x :lle, $0 \leq x < 11$, $x \in \mathbb{N}$
`print "2*x = ", 2*x`

Toisto

```
for x in [0,1,2,3,4,5]:  
    print x
```

Toisto: Definiitti

Tulostus:  for x in [0,1,2,3,4,5]:
 print x

0
1
2
3
4
5

Toisto: Definiitti

```
if (onkoSeina) :
```

```
    otaAskel
```

```
else:
```

```
    käännyVasemmalle
```

```
if (onkoSeina) :
```

```
    otaAskel
```

```
else:
```

```
    käännyVasemmalle
```

```
#Ylläoleva vielä 4 kertaa kopioituna....
```

Toisto

```
for x in [1,2,3,4,5,6,7,8,9]:  
    if (onkoSeina) :  
        otaAskel  
    else:  
        käännyVasemmalle
```

Toisto

Indefiniitin toiston määrää ei välttämättä tiedetä ennen silmukan suorittamista. Toistojen määrä riippuu ohjelman tilasta.

Toisto: Indefiniitti

Kaksi mahdollisuutta

- While-loop
- For-loop

Toisto: Indefiniitti

Syntaksi, WHILE

```
while <ehto>:  
    lohko
```

Toisto: Indefiniitti

Syntaksi, FOR

```
for <muuttuja> in <lista>:  
    lohko
```

Toisto: Indefiniitti

Lista?

- Tietorakenne (luento 5)
- Kokoelma järjestyksessä olevia asioita

Toisto: Indefiniitti

Pythonissa lista helppo
generoida funktiolla
(seuraava luento) **Range**

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

`range (3,n)`  `[3,4,5,...,n-1]`

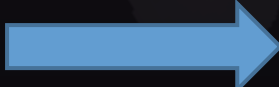
Toisto: Range

Syntaksi, RANGE

range(<alkuarvo>,
<lopetusarvo>, <askel>)

Alkuarvo inklusiivinen (otetaan mukaan)

Loppuarvo eksklusiivinen (ei oteta mukaan)

range(6,13,3)  [6,9,12]

range(20,0,-10)  [20,10]

Toisto: Range

Ongelma:

Löydä ja tulosta kaikki
parilliset luvut välillä $[2,n]$

Toisto: Indefiniitti

```
x = 2
while x <= n:
    if x%2 == 0:
        print x
    x = x+1
```

Toisto: Indefiniitti

```
for x in range(2,n+1):  
    if x%2 == 0:  
        print x
```

Toisto: Indefiniitti

```
| for x in range(2,n+1, 2):  
| | print x
```

Toisto: Indefiniitti

```
for x in {1,2,3,4,5,6,7,8,9}:  
    if (onkoSeina):  
        otaAskel  
    else:  
        käännyVasemmalle
```

Toisto

```
while (True):  
    if (onkoSeina):  
        otaAskel  
    else  
        käännyVasemmalle
```



Toisto



Toisto: sisäkkäinen toisto

Toistoa voi olla myös toiston sisällä

```
for x in range(0,3):  
    for y in range(0,3):  
        print x,y
```

Toisto: sisäkkäinen toisto

Tulostus:

0 0

0 1

1 0

1 1

2 0

2 1



for **x** in range(~~0~~**2**):

for **y** in range(~~0~~**2**):

print **x** , **y**

Program
finished!

Toisto: sisäkkäinen toisto

Kaksi toistorakennetta



Toisto

Peräkkäisyys



Valinta



Toisto



Kertaus

”En mä koskaan irti
susta päästää voi. Sä oot
kuin melodia, joka soi ja
se soi ja soi ja soi.
Luupilla mun korvissa”

- Robin loputtomasta silmukasta

TAUKO

Etsi kaikki 5 ja 7
jaolliset luvut välillä
[10,150]

Käytännön harjoite

```
for i in range(10,151):  
    if(i%5 == 0 and i%7 == 0):  
        print i
```

Käytännön harjoite

```
for i in range(10,151):  
    if(i%5 == 0):  
        if(i%7 == 0):  
            print i
```

Käytännön harjoite

Laske kuinka monta
numeroa 1 luvussa
5915131 on.

Käytännön harjoite

```
luku= 5915131
```

```
maara = 0
```

```
for i in str(luku):
```

```
    if(i=="1"):
```

```
        maara += int(i)
```

Käytännön harjoite

Tee ohjelma, joka lukee
käyttäjältä lukuja ja käyttäjän
syötettyä luku -1, laskee
syötettyjen lukujen
keskiarvon.

Tätä ei voi vielä koodata suoraan; tarkennetaan.

Käytännön harjoite

Laske käyttäjän syöttämiä
lukuja yhteen, KUNNES
käyttäjä syöttää -1. Jaa
summa syötettyjen lukujen
määrällä.

Käytännön harjoite

```
#summan pitää olla yksi, jotta käyttäjän syöttämä -1 kumotaan
summa = 1.0
luku = 0
#määrän pitää olla -1, koska käyttäjän syöttämä -1 lasketaan myös
määrään
maara = -1
```

```
while(luku != "-1"):
    #luetaan syöte merkkijonona
    luku = raw_input()
    #muutetaan merkkijono luvuksi
    summa += int(luku)
    maara += 1
```

```
#kokonaislukujakolasku, toisen operandeista tulee olla desimaaliluku
print summa/maara
```

Käytännön harjoite

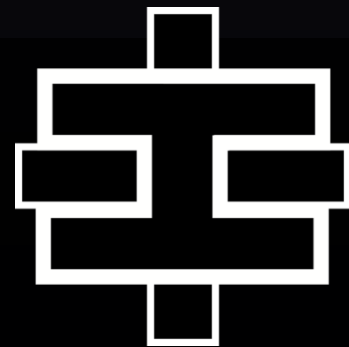
Oletetaan yritys, jolla on 97, 149 ja 827 ihmisen kokoisia ryhmiä.

Tiedämme yrityksestä

- Työntekijöiden määrän numeroiden summa on 41
- Yrityksessä on vähintään 10^7 ja enintään 10^8 ihmistä töissä.

Kuinka monta ihmistä yritys työllistää?

Kotitehtävä



Koodaat tekstieditoria.
Tarvitset etsi-toiminnon, eli
tavan löytää tekstistä kohdan,
josta haluttu merkkijono
alkaa.

(älä käytä Pythonin valmista find-metodia, vaan tee itse sama
toiminnallisuus)

Kotitehtävä

```
mjono = "Tässä jotakin tekstiä"  
etsittava = "ässä"
```

```
alkuindeksi = -1
```

```
for i in range(0, len(mjono)):  
    if(mjono[i] == etsittava[0]):  
        for x in range(0, len(etsittava)):
```

```
            if(i+x >= len(mjono)):  
                break
```

```
            if(mjono[i+x] != etsittava[x]):  
                break
```

```
            if(x == len(etsittava)-1):  
                alkuindeksi = i
```

```
print(alkuindeksi)
```

Käytännön harjoite