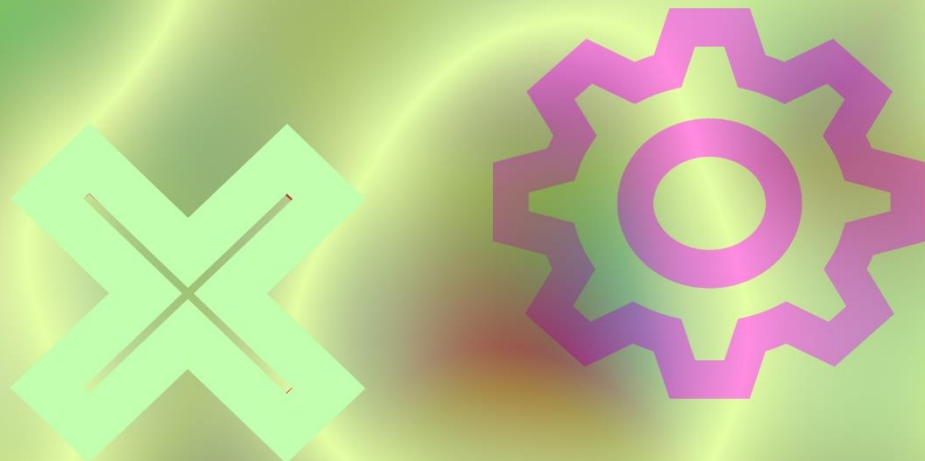




Perusmatematiikkaa ohjelmoimalla

MALLIRATKAISUT

Yläkoulun matematiikan lisämateriaali

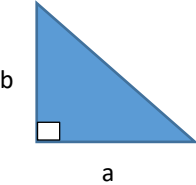
Jouko Järvenpää

PERUSMATEMATIIKKA OHJELMOIMALLA

matemaattisen ohjelmoinnin
harjoituskirja 7-luokalle

MALLIRATKAISUT

 TEHTÄVIÄ

Tehtävä 1. Tiedonhakua ohjelmointikielistä Etsi internetistä tietoa ja kerro lyhyesti, mitkä ovat suosituimmat ohjelmointikielet. Suosituimmat ohjelmointikielet 2020 Python, Java, C, C++, JavaScript, R, Arduino, Go. Mikä on Python-kielen sijoitus suosituimpien ohjelmointikielten listassa? Tuossa listassa ensimmäinen	Tehtävä 2. Tiedonhakua ohjelmoinnista Etsi internetistä vastaukset seuraaviin kysymyksiin. Vastaa lyhyesti yhdellä lauseella. Mikä on ohjelmointikielen tulkki? Ohjelmointikielen tulkki on tietokoneohjelma, joka käsittelee ohjelmointikielisiä lauseita ja suorittaa ne. Onko Python tulkattava vai käännettävä ohjelmointikieli? tulkattava
Tehtävä 3. Ohjelmitavissa laitteissa on prosessori Luettele muutamia prosessoriohjattuja laitteita, joiden käyttö perustuu johonkin ohjelmoituun toimintaan. Mitä ohjelmitavia eli prosessoriohjattuja laitteita sinulla on käytössäsi?	Tehtävä 4. Valmistetaan kaakaota Kirjoita arkikielinen ohje (algoritmi) kaakaon valmistamiseksi. Ohjeen tuloksena tulee saada kaakaojauheesta kupillinen maitokaakaota. (Kaikkia rivejä ei välttämättä tarvita) - ota kattila - lämmitä maito kattilassa lämpimäksi - sekoita kaakaojauhetta lämpimään maitoon - sekoita, kunnes kaakaojauhe on lionnut - kaada kaakao kuppiin - juo
Tehtävä 5. Algoritmit matematiikassa Matematiikan laskulausekkeet ovat myös algoritmeja. Kirjoita matemaattinen algoritmi suorakulmaisen kolmion pinta-alan laskemiseksi. Ala = $a \cdot b / 2$ 	Tehtävä 6. Etunimet aakkosjärjestykseen Kirjoita mahdollisimman tarkka algoritmi, jolla saadaan järjestettyä aakkosjärjestykseen kolme nimeä <i>Assi</i>, <i>Aimo</i>, <i>Aki</i>. (Kaikkia rivejä ei välttämättä tarvita) - Asettele nimet ensimmäisen kirjaimen mukaan järjestykseen - Jos kirjain on sama, tutki toinen kirjain - asettele järjestykseen toisen kirjaimen mukaan - jos lista on järjestyksessä, lopeta Huom! tämä ratkaisu ei ole yleispätevä kaikille nimille.

TEHTÄVIÄ

Tehtävä 7. Ensimmäinen Python-ohjelma

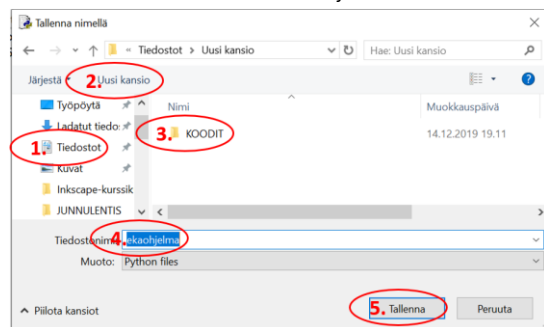
1. Käynnistä IDLE-editori ja valitse Shell-ikkunan ylävalikosta **File – New File**.

2. Kirjoita uuteen tiedostoon oheisen mallin mukaiset koodit.

```
# ensimmäinen ohjelmani, oma nimesi ja päivämäärä
print("Hei maailma")
print("Tämä on ensimmäinen Python-koodini")
```

3. Tallenna tiedosto KODIT- kansioon valitsemalla File – Save as...

Tallennuskansion luonti ja tiedoston nimeämisvaiheet näkyvät oheisesta kuvasta. Tallenna koodi nimellä *ekaohjelma*.



4. Suorita ohjelma valitsemalla Run – Run module.

Tehtävä 8. print-komennolla tulostetaan näytölle

Aloita uusi tiedosto valitsemalla File – New File

Koodaa ohjelma, joka tulostaa koko nimesi yhdelle riville.

Tallenna koodi KODIT-kansioon nimellä *nimentulostus*.

Suorita ohjelma valitsemalla Run – Run Module tai painamalla näppäimistön F5-näppäintä.

Minna Annikki Leino

```
print("Oma Nimi")
```

Tehtävä 9. Virheet tutuksi

Kirjoita tarkoituksella syntaksivirhe ohjelmakoodiin ja tutki, millaisen virheilmoituksen Python-tulkki ilmoittaa. Aloita uusi tiedosto ja kirjoita koodi

```
# syntaksivirheitä
```

```
prin("Väärä komento")
```

Tallenna koodi KODIT kansioon nimellä *virheet*. Suorita ohjelma. Mikä virheilmoitus tulostuu?

```
NameError: name 'prin' is not defined
```

Korjaa print-komento ja poista toinen lainausmerkki

```
# syntaksivirheitä
```

```
print("Väärä komento")
```

```
SyntaxError: EOL while scanning string literal
```

Korjaa virhe ja tallenna koodi uudelleen. Koodi-ikkunoita voi olla avoimina useita. Tarpeettomia ikkunoita on kuitenkin hyvä sulkea.

Tehtävä 10. Tulostuksen rivittäminen

```
print("""Lukujoukkoja:
```

```
Luonnolliset luvut: N= {0, 1, 2, 3, ...}
```

```
Kokonaisluvut: Z= {-3, -2, -1, 0, 1, 2, 3, ...}
```

```
Rationaaliluvut: Q= {Z + murtoluvut}""")
```

 TEHTÄVIÄ

Tehtävä 11. Lukujen ja laskulausekkeiden tulostaminen

Tutki, mitä seuraavat print-lauseet tulostavat. Päätele ensin ja kokeile sitten koodaamalla rivit Pythonilla.

```
print ("Lukujen -7 ja -6 tulo on:", -7*(-6))
```

tulostus: _____

```
print (1*2*3*4+6)
```

tulostus: _____

```
print(1+2+3+4*6)
```

tulostus: _____

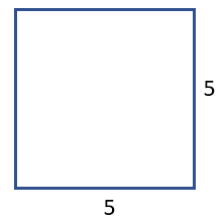
Tehtävä 12. Neliön piiri ja pinta-ala

Koodaa ohjelma, joka laskee oheisen neliön **piirin** ja **pinta-alan**. Ohjelma tulostaa piirin ja pinta-alan. Yhdistä siis tulostuslauseessa merkkijonon loppuun laskulausekkeet; esimerkiksi

```
print ("Neliön piiri on", 4*5)
```

```
print ("Neliön pinta-ala on", 5*5)
```

Tallenna ohjelma KODIT-kansioosi nimellä *neliö* ja testaa ohjelman toiminta.

**Tehtävä 13. Kolmion pinta-ala**

Koodaa ohjelma, joka laskee oheisen kolmion **pinta-alan**. Kuvassa on annettu kolmion kannan pituus ja korkeus. Ohjelma myös tulostaa pinta-alan. Yhdistä siis tulostuslauseessa merkkijonon loppuun laskulauseke.

Tallenna ohjelma KODIT-kansioosi nimellä *kolmio* ja testaa ohjelman toiminta

```
print ("Kolmion pinta-ala on", 820*615/2)
```

**Tehtävä 14. Simon ensimmäinen ohjelma**

Simo on yrittänyt koodata yksinkertaisen ohjelman, mutta koodissa on useita virheitä. Koodissa on sekä syntaksi- että laskennallisia virheitä. Kirjoita koodi syntaktisesti oikein ja korjaa myös matemaattiset virheet. Tallenna korjattu ohjelma KODIT-kansioosi nimellä *omenaostos*.

```
# matemaattinen tehtävä
```

```
print ("Omenat maksavat 2 euroa/kg")
```

```
print ("Matti ostaa omenoita 2,5 kg")
```

```
print ("Omenaostos maksaa silloin:", 2.5*2)
```

 TEHTÄVIÄ

Tehtävä 15. Merkkijono ja laskulauseke tulostuksessa

Täydennä seuraava ohjelma siten, että se tulostaa mainitun matemaattisen lausekkeen arvon. Päättele ensin ja koodaa sitten merkkijonojen perään laskutoimituksen suorittava lauseke. Tallenna valmis ohjelma KOODIT-kansioosi nimellä *aritmetiikkaa*.

print ("Lukujen 25 ja -12 vastaluvut ovat",-25,-(-12))

print ("Lukujen 6 ja 5 summan ja erotuksen tulo on", (6+5)(6-5))*

print ("Lukujen 15 ja -12 tasajako antaa tuloksen", 15/(-12))

print ("Jakolaskun 3 jaettuna 8:lla jakojäännös on", 3%8)

Tehtävä 16. Desimaalipiste liikkuu

Desimaalilukua kerrottaessa ja jaettaessa 10:llä, 100:lla tai vaikkapa 1000:lla, siirtyy desimaalipilkku. Mutta miten se siirtyy? Voit varmistaa tietosi seuraavalla ohjelmalla. Koodaa siis ohjelma ja tallenna KOODIT-kansioon nimellä *kertaluokka*. Suorita ohjelma ja kirjoita 'pilkkusääntö' omin sanoin havaintosi perusteella.

desimaaliluvun kertominen ja jakaminen 10, 100, 1000

*print (3.14 * 10)*

*print (3.14 * 100)*

*print (3.14 * 1000)*

print (314.0 / 10)

print (314.0 / 100)

print (314.0 / 1000)

SÄÄNTÖ: _____

Tehtävä 17. Yksinkertaista Simon lausekkeet

Seuraavassa koodissa Simo on kirjoittanut laskulausekkeet tarpeettoman pitkästi. Tehtäväsi on kirjoittaa lausekkeet lyhyemmin (sievennettynä). Käytä sieventämisessä aritmeettisia operaattoreita. Tallenna yksinkertaistettu ohjelma KOODIT-kansioon nimellä *sievennykset*.

lausekkeiden sievennys

*print (10*2)*

print (7(-5))*

print (6(4+(-10)))*

*print (7**10)*

print (4(-5)+7**5)*

Tehtävä 18. Korjaa Simon tulostuslauseet

Simo on yrittänyt koodata tulostuslauseita, mutta koodissa on useita virheitä. Koodissa on sekä syntaksi- että laskennallisia virheitä. Kirjoita koodi syntaktisesti oikein ja korjaa myös matemaattiset virheet. Tallenna korjattu ohjelma KOODIT-kansioosi nimellä *laskuvirheet*.

print ("Desimaalilukujen summa", 12.45 + 10.55)

*print ("Kolme potenssiin neljä on", 3**4)*

print ("Jakojäännös osamäärästä 12÷5 on", 12%5)

*print ("Lukujen 5,5 ja 12,5 tulo on", 5.5 *12.5)*

Tehtävä 19. Merkkijonomuuttujia

Täydennä alla oleva koodi. Aseta muuttujien arvoiksi omat tietosi. Täydennä myös print-lauseet sellaisiksi, että saat tulostettua muuttujien arvot, kukin omalle rivilleen.

```
etunimi = "Minna"
tokanimi = "Kristiina"
sukunimi = "Virtanen"
```

```
print ("Etunimi:", etunimi)
print ("Toinen nimi:", tokanimi)
print ("Sukunimi:", sukunimi)
```

Tallenna täydennetty ohjelma KODIT-kansioon nimellä *merkkimuuttujat*. Testaa ohjelman toiminta.

Tehtävä 20. Merkkijonojen yhdistäminen eli katenointi

Merkkityyppisiä muuttujia voidaan tulostuslauseessa yhdistää + merkillä. Toisaalta tulostuslauseessa voidaan tulostettavat muuttujat luetella pilkuilla eroteltuna. Kirjoita seuraava koodi ja tutki, miten tulostustavat eroavat toisistaan lopputuloksen kannalta?

```
s = "Suurin"
y = "yhteinen"
t = "tekijä"
print (s+y+t)
print (s,y,t)
```

Tallenna ohjelma KODIT-kansioon nimellä *katenointi*.

TULOSTUKSET EROAVAT (s,y,t) lisää välit muuttujiin väliin

LISÄTEHTÄVÄ: yhdistä tulostuslauseeseen `print (s+y+t)` välilyönnit (" ") muuttujien väliin, jotta myös tämän rivin tulostus saadaan oikein välilyöntien kanssa. `print (s+" "+y+" "+t)`

Tehtävä 21. Merkkien monistaminen

Merkkejä ja merkkijonoja voidaan Pythonissa monistaa * operaattorilla. Esimerkiksi lause `print (5*"OHO")` tulostaa OHOHOHOHOHOHO.

Mieti, miten saat tulostettua oheisen kuvion käyttämällä välilyönti (" ") ja "#"-merkkiä sekä merkkien monistusta. Tallenna ohjelma KODIT-kansioon nimellä *pyramidi*. Testaa tulostus.

```

      #
    ###
  #####
```

```
#pyramidi
print (2*" "+"#")
print (" "+3*"#")
print (5*"#")
```

Tehtävä 22. Simon merkkijonon monistus

Simo on yrittänyt koodata merkkejä toistavan ohjelman, mutta koodissa on paljon syntaksivirheitä ja virheellisiä tunnisteita. Korjaa virheet, tallenna koodi ja suorita ohjelma. Mitä koodi tulostaa?

```
a= "papu"
b= "pata"
toisto_kerrat = 5
print ("Tulostetaan useita kertoja", a+b)
print (toisto_kerrat *( a+b))
```

Oikean tulosteen pitäisi näyttää seuraavalta:

**Tulostetaan useita kertoja papupata
papupatapapupatapapupatapapupatapapupata**

Tehtävä 23. Tietoa näppäimistöltä

Edellisen sivulla on esimerkki, jossa luetaan kaksikokonaislukua a ja b sekä tulostetaan näiden summa. Koodaa ohjelma ja lisää siihen vielä lukujen a ja b aritmeettiset operaatiot *erotus*, *tulo*, *osamäärä*, *tasajako* ja *jakoäännös*.

Operaatioiden tulos voidaan asettaa muuttujaan tai laskulauseke voidaan yhdistää tulostuslauseeseen. Ohjelman pitäisi tulostaa seuraavanlaisesti:

```
# aritmeettiset operaatiot
a = int(input("Anna luku a: "))
b = int(input("Anna luku b: "))

summa = a+b
erotus = a-b
tulo = a*b
osamäärä = a/b
tasajako = a//b
jakoäännös = a%b

print
("Summa:",summa,"Erotus:",erotus,"Tulo:",tulo,"Osamäärä:",osamäärä)

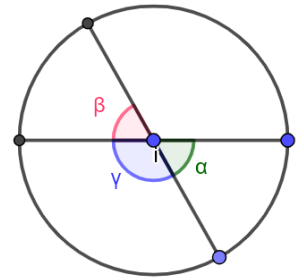
print ("Tasajako:",tasajako,"Jakoäännös:",jakoäännös)
```

Tehtävä 24. Keskuskulma, ristikulma, vieruskulma

Ympyrään on piirretty kaksi halkaisijaa kuvan osoittamalla tavalla. Laadi ohjelma, joka laskee kulmien β (*beta*) ja γ (*gamma*) suuruudet, kun ohjelmaan syötetään kulman α (*alfa*) suuruus. Ohjelma kysyy kulman α ja laskee tämän avulla muut kulmat. Kulmien arvo tulostetaan yhdellä *print*-lauseella.

```
# keskuskulma
alfa = int(input("Anna kulma alfa: "))
beta = alfa

gamma = 180-alfa
```



Tallenna ohjelma KOODIT-kansioon nimellä *keskuskulmat*. Varmista ohjelman toiminta.

Tehtävä 25. Säännöllisen monikulmion kulma

Säännöllisen monikulmion kulman α suuruus saadaan laskettua kaavalla

$$\alpha = \frac{n-2}{n} \cdot 180^\circ, \text{ missä } n \text{ on monikulmion kulmien määrä}$$

Esimerkiksi neliön kulman suuruus on

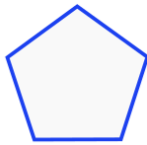
$$\alpha = \frac{4-2}{4} \cdot 180^\circ = 90^\circ$$

säännöllinen monikulmio

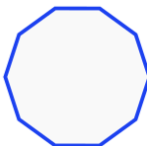
$n = \text{int}(\text{input}(\text{"Anna kulmien määrä: "}))$

$\text{kulma} = (n-2)/n * 180$

$\text{print}(\text{"Monikulmion kulma on "}, \text{kulma}, \text{"astetta ja kulmien summa on "}, n * \text{kulma})$



$n = 5$



$n = 10$

Tehtävä 26. Simon suorakulmio ja kolmio

Simo osaa hyvin geometriaa, mutta ohjelmoinnissa on vielä harjoiteltavaa. Simo on kirjoittanut oheisen koodin, mutta siinä on useita huolimattomuusvirheitä. Korjaa virheet ja tallenna korjattu ohjelma KOODIT-kansioon nimellä *suorakulmion_ja_kolmion_ala*. Leveys ja korkeus annetaan kokonaislukuina.

lasketaan suorakulmion ja

vastaavilla sivujen pituuksilla esitetyn suorakulmaisen kolmion alat

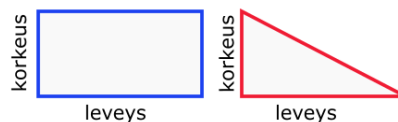
$\text{leveys} = \text{int}(\text{input}(\text{"Anna leveys kokonaislukuna"}))$

$\text{korkeus} = \text{int}(\text{input}(\text{"Anna korkeus kokonaislukuna"}))$

$\text{sk_ala} = \text{leveys} * \text{korkeus}$ # suorakulmion ala

$\text{kolmion_ala} = \text{leveys} * \text{korkeus} / 2$ # vastaavan suorakulmaisen kolmion ala

$\text{print}(\text{"Suorakulmion ala on "}, \text{sk_ala}, \text{"ja kolmion ala on "}, \text{kolmion_ala})$



Tehtävä 27. Float-lukujen pyöristäminen

Desimaaliluvut (float) pyöristetään *round*-komennolla. Lisää seuraaviin tulostuslauseisiin *round*-pyöristykset siten, että luku pyöristetään ja tulostetaan mainittuun tarkkuuteen.

pyöristystä

luku = 10.76547

print ("Tuhannesosan tarkkuudella", round(luku, 3))

print ("Kahden desimaalin tarkkuudella", round(luku, 2))

print ("Kymmenestuhannesosan tarkkuudella", round(luku, 4))

print ("Kokonaislukuna", round(luku))

täydennä puuttuvat pyöristykset

Tallenna täydennetty ohjelma KODIT-kansioosi nimellä *pyöristystä* ja testaa ohjelman toiminta.

Tehtävä 28. Desimaalilukujen kerto- ja jakolaskuja

Koodaa ohjelma, joka kysyy kaksi desimaalilukua ja tulostaa näiden tulon ja osamäärän kahden desimaalin tarkkuudella. Ohjelman toiminta voisi näyttää seuraavalta.

Anna a: 0.0745

Anna b: 1.234

0.0745 * 1.234 = 0.09

0.0745 ÷ 1.234 = 0.06

tulo ja osamäärä

a = float(input("Anna luku a: "))

b = float(input("Anna luku b: "))

print(a, "*", b, "=", a * b)

print(a, "/", b, "=", a / b)

Koodaa tulostuslauseet mallin mukaisesti. Tallenna toimiva ohjelma KODIT-kansioon.

Laske ohjelmaasi käyttämällä seuraavat kerto- ja jakolaskut:

9.999 · 0.9999 = _____

9.999 ÷ 0.9999 = _____

333.33 · 0.33333 = _____

333.33 ÷ 0.33333 = _____

Tehtävä 29. Kertolasku potenssiksi ja summa kertolaskuksi

Simolla on vielä harjoiteltavaa koodin tehokkaassa kirjoittamisessa. Alla on Simon kirjoittama koodi, joka laskee tietyt operaatiot luvuille a ja b. Sievennä laskulausekkeet siten, että molempien muuttujien operaatiot mahtuvat yhdelle riville. Tallenna sievennetty ohjelma KODIT-kansioon nimellä *lyhyet*.

sievennä koodi ja merkitse yhdistetyt operaatiot lyhyemmin

a = float(input("Anna luku a: "))

print(a * 10 ** 5)

b = float(input("Anna luku b: "))

print(b + 10 * 100)

Tehtävä 30. Pankkiirinpyöristys

Osaatko selittää pankkiirin pyöristyssäännön esimerkiksi lukujen 1.455 ja 1.45 osalta, kun molemmat luvut pyöristetään yhden desimaalin tarkkuuteen?



Koodaa ohjelma, joka kysyy kaksi float-lukua. Ohjelman tulee pyöristää ja tulostaa luvut yhden desimaalin tarkkuuteen. Syötä luvut 1.455 ja 1.45.

Miten lukujen pyöristetyt tulokset eroavat?

pankkiirin pyöristys

a = float(input("Anna luku a: "))

b = float(input("Anna luku b: "))

print(round(a, 1), round(b, 1))

Luku 1.455 on Pythonissa pyöristettynä yhden desimaalin tarkkuuteen ____.

Luku 1.45 on Pythonissa pyöristettynä yhden desimaalin tarkkuuteen ____.

Miten selität pyöristysten erilaiset lopputulokset pankkiirin pyöristyssäännön perusteella?

TEHTÄVIÄ

Tehtävä 31. Itseisarvo

Koodaa ohjelma, joka tulostaa käyttäjän antaman luvun itseisarvon:

- kirjoita ohjelman alkuun *from math import **
- ohjelma pyytää käyttäjältä luvun
- ohjelma tulostaa sekä annetun luvun ja sen itseisarvon

Tallenna ohjelma KODIT-kansioon nimellä *itseisarvo* ja testaa ohjelmaa positiivisilla ja negatiivisilla luvuilla. Ohjelman tulosteet voisivat näyttää seuraavalta:

```
# itseisarvo
from math import *
```

```
x = int(input("Anna luku x: "))
print ("Luvun",x,"itseisarvo on ",abs(x))
```

Syötä kokonaisluku: -25
Luvun -25 itseisarvo on 25

Tehtävä 32. Suurin yhteinen tekijä, gcd

Syt on hyödyllinen esimerkiksi murtolukujen supistamisessa. Murtoluvun osoittaja ja nimittäjä saadaan supistettua yksinkertaisimpaan muotoonsa.

Koodaa ohjelma, joka tulostaa kahden kokonaisluvun suurimman yhteisen tekijä (synt). Ohjelma kysyy kaksi kokonaislukua, laskee ja tulostaa lukujen suurimman yhteisen tekijän (gcd). Tallenna ohjelma KODIT-kansioon nimellä synt.

Selvitä ohjelman avulla, mikä on alla olevien murtolukujen osoittajan ja nimittäjän suurin yhteinen tekijä. Supista alla esitetyt murtoluvut yksinkertaisimpaan muotoonsa.

tulostaa kahden kokonaisluvun synt:n

```
from math import *

os = int(input("Anna osoittaja: "))
nim = int(input("Anna nimittäjä: "))
synt = gcd(os,nim)

print ("Lukujen",os,nim,"suurin yhteinen tekijä on",synt)

print ("Murtoluku",os,"/",nim," on supistettuna", int(os/synt),"/",int(nim/synt))

 $\frac{8}{16}$  synt(____), supistettuna ____

 $\frac{72}{90}$  synt(____), supistettuna ____

 $\frac{36}{108}$  synt(____), supistettuna ____
```

Tehtävä 33. Murtoluvun supistaminen

Jatka tehtävän 32 koodia siten, että ohjelma myös supistaa osoittajan (os) ja nimittäjän (nim) suurimmalla yhteisellä tekijällä. Ohjelma tulostaa lopuksi supistetun murtoluvun oheisen mallin mukaisesti.

```
Anna osoittaja: 36
Anna nimittäjä: 108
Lukujen 36 108 suurin yhteinen tekijä on 36
Murtoluku 36 / 108 on supistettuna 1 / 3
```

tulostaa kahden kokonaisluvun synt:n

```
from math import *

os = int(input("Anna osoittaja: "))
nim = int(input("Anna nimittäjä: "))
synt = gcd(os,nim)

print ("Lukujen",os,nim,"suurin yhteinen tekijä on",synt)

print ("Murtoluku",os,"/",nim," on supistettuna", int(os/synt),"/",int(nim/synt))
```

Tehtävä 34. Neliön sivu ja pinta-ala, neliöllinen kasvu

Neliön sivun pituus voidaan laskea $\sqrt{A}$, kun neliön pinta-ala A tunnetaan. Esimerkiksi neliön pinta-ala on 64 cm². Tällöin neliön sivun pituus on $\sqrt{64} = 8$ cm. Koodaa ohjelma, joka kysyy neliön pinta-alan, laskee neliön sivun pituuden (sqrt) ja tulostaa vastauksen. Tallenna ohjelma KODIT-kansioon nimellä *sivunpituus*. Selvitä ohjelman avulla seuraavien neliöiden sivun pituudet.

neliön sivu
from math import *

```
x = int(input("Anna neliön ala "))
print ("Neliön sivun pituus on ",sqrt(x))
```

PERUSMATEMATIIKKA OHJELMOIMALLA

matemaattisen ohjelmoinnin
harjoituskirja 8-luokalle

MALLIRATKAISUT

Tehtävä 1. Merkkijonon merkkien paikka, indeksointi

Oheisessa koodissa on esitelty merkkijono *mjono*. Selvitä, mitkä sanat tulostuvat *mjono* merkkijonosta poimituista merkeistä. Kirjoita kirjaimet viivoille.

poimitaan merkkejä

```
mjono = "PROSENTTIYKSIKKÖ"
```

```
print (mjono[1],mjono[4],mjono[8],mjono[5],mjono[2]) REINO
```

```
print (mjono[9],mjono[3],mjono[10],mjono[8],mjono[8]) YSKII
```

```
print (mjono[-1],mjono[-4],mjono[-5],mjono[-8],mjono[-11]) ÖISIN
```

Tarkista tulostuvat sanat koodaamalla ohjelma. Tallenna valmis ohjelma KOODIT-kansioon nimellä *poimittuja*.

Tehtävä 2. Merkkijonon pituus, tulostus takaperin

Koodaa ohjelma, joka pyytää käyttäjältä kaksi merkkijonoa; *eka* ja *toka*. Ohjelma tulostaa merkkijonojen pituudet. Lopuksi tulostetaan molemmat merkkijonot takaperin. **VIHJE:** merkkijono saadaan tulostettua takaperin ottamalla käyttöön kolmas indeksi, ns. 'harppaus'. Esimerkiksi *eka*="TIMO", niin **print (eka[::-1])** tulostaa merkkijonon *eka* takaperin *OMIT*. Tallenna ohjelmasi KOODIT-kansioon nimellä *takaperin*. Ohjelman toiminta näyttää seuraavalta:

```
Anna merkkijono: iso rikas sika sökösakissa kirosi  
Anna toinen merkkijono: ollapa pallo  
Merkkijonojen pituudet ovat 33 ja 12  
Takaperin:  
isorik assikasökös akis sakir osi  
ollap apallo
```

takaperin

```
eka = input("Anna merkkijono: ")  
toka = input("Anna toinen merkkijono: ")  
print ("Merkkijonojen pituudet ovat",len(eka),"ja",len(toka))  
print ("Takaperin:")  
print (eka[::-1])  
print (toka[::-1])
```

Tehtävä 3. Viipale merkkijonosta

Täydennä alla olevaan ohjelman indeksiviittaukset siten, että ohjelma tulostaa allekkain merkkijonon *lause* merkit seitsemän merkin mittaisissa alimerkkijonoissa eli viipaleissa. **VIHJE:** Kun alimerkkijonosta jättää toisen indeksin pois, tulostuu merkit aloituskohdasta merkkijonon loppuun. Esim. *ali[10:]* tarkoittaa merkkejä indeksistä 10 merkkijonon loppuun. Ole tarkkana indeksien kanssa, jotta kullekin riville tulostuu seitsemän merkkiä. Tallenna valmis ohjelma KOODIT-kansioon nimellä *viipaleet*.

merkkijono viipaleina

```
lause = "MATEMATIIKKAA ON KAIKKIALLA!"
```

```
print (lause[0,7])
```

```
print (lause[7:14])
```

```
print (lause[14:21])
```

```
print (lause[21: ])
```

Tulostuksen pitää näyttää seuraavalta:

**MATEMAT
IIKKAA
ON KAIK
KIALLA!**

Huomaa, että myös välilyönnit ovat merkkejä.

Tehtävä 4. Merkkijonojen pituuksien suhde

Koodaa ohjelma, joka kysyy käyttäjältä kaksi merkkijonoa *mjono1* ja *mjono2*. Ohjelma tulostaa **prosenttilukuna**, kuinka monta prosenttia merkkijonon *mjono1* pituus on merkkijonon *mjono2* pituudesta.

Tallenna toimiva ohjelma KOODIT-kansioon nimellä *pituuksuhde*. Ohjelman toiminta näyttää seuraavalta:

```
Kirjoita jokin merkkijono: keskuskulma  
Kirjoita vielä toinen merkkijono: tangenttikulma  
Ensimmäisen merkkijonon pituus on 78.6 %:a toisesta merkkijonosta
```

merkkijonosuhteet

```
mjono1 = input("Kirjoita jokin merkkijono: ")
```

```
mjono2 = input("Kirjoita vielä toinen merkkijono: ")
```

```
suhde = round(len(mjono1)/len(mjono2)*100,1)
```

```
print ("Ensimmäisen merkkijonon pituus on",suhde,"%:a toisesta  
merkkijonosta")
```

Muistathan, että desimaalilukuja voidaan pyöristää **round(x,n)**-komennolla, jossa *x* on jokin float-tyyppinen luku ja *n* on esitettävien desimaalien määrä. Esimerkiksi **round(2.4567,1)** antaa tulokseksi 2.5.

Pyöristä merkkijonojen suhdeluku yhden desimaalin tarkkuuteen.

TEHTÄVIÄ

Tehtävä 5. Tutkitaan totuusarvoja, True tai False

Tutki alla olevaa koodia ja päättelä, mitä totuusarvoja loogiset lausekkeet tulostavat kullekin riville. Kirjoita päättelysi tulos näkyviin viivoille.

Koodaa ohjelma ja tarkista tulokset. Tallenna ohjelma KOODIT-kansioon nimellä *totuusarvot*.

```
# tutki totuusarvot
x = -7
y = 2
print (y > x)                                totuusarvo True

print (x*y < 0 and abs(x*y)>0) # abs on itseisarvo totuusarvo True

print (not(x==y) or (x != y))                totuusarvo True

print (x%2 == 0 and y%2 != 0)                totuusarvo False
```

Tehtävä 6. Vertailuoperaattorit < ja > sekä ==

Vertailuoperaattoreita voidaan käyttää myös **merkkijonojen vertailussa**. Esimerkiksi "a" < "b" vertailun tulos on *True*, koska kirjain a on aakkosissa ennen b kirjainta. Samoin voidaan vertailla esimerkiksi nimien aakkosjärjestyttä. Vertailu "Timo" < "Tomi" on myös *True*.

Koodaa ohjelma, joka kysyy kaksi etunimeä ja tallentaa nimet muuttujiin *nimi1* ja *nimi2*. Ohjelmassa vertaillaan etunimien järjestys. Nimien aakkosjärjestyksvertailun tulos ilmoitetaan totuusarvoilla *True* tai *False*.

Tallenna ohjelma KOODIT-kansioon nimellä *etunimet*. Ohjelman toiminta näyttää seuraavalta:

```
# etunimien aakkosjärjestys
nimi1 = input("Anna nimi1: ")
nimi2 = input("Anna nimi2: ")
print (nimi1,"on aakkosissa ennen nimeä",nimi2,"tulos:",nimi1<nimi2)
print (nimi2,"on aakkosissa ennen nimeä",nimi1,"tulos:",nimi2<nimi1)
```

Tehtävä 7. Merkkijono lukutyypiksi, jakojäännösoperaattori %

```
# parillisuustestaus
luku = int(input("Syötä kokonaisluku: "))
print ("VÄITE: luku",luku,"on parillinen, tulos: ", luku%2==0)
```

Tehtävä 8. Jaollisuustutkimus

```
# jaollisuus
luku = int(input("Syötä kokonaisluku: "))
print ("Luku",luku,"on jaollinen luvulla 2, tulos:",luku%2==0)
print ("Luku",luku,"on jaollinen luvulla 3, tulos:",luku%3==0)
print ("Luku",luku,"on jaollinen luvulla 5, tulos:",luku%5==0)
```

 TEHTÄVIÄ

Tehtävä 9. Tulostus nimen pituuden ehdolla, if-lause

Koodaa ja täydennä alla esitetty ohjelma. Tehtäväsi on kirjoittaa print-lauseiden korostettuihin kohtiin if-lauseessa suoritettun vertailun tulos: *"nimessä on A- tai a-kirjain", "Nimessä on alle 4 kirjainta", "nimessä ei ole a-kirjainta"* jne.

if-elif-tehtävä

```
etunimi = input("Kirjoita etunimesi: ")
```

```
if len(etunimi) < 4:
```

```
    print("Nimesi on melko lyhyt")
```

```
elif len(etunimi) >= 4 and len(etunimi) <= 6:
```

```
    print("Nimessäsi on 4-6 merkkiä")
```

```
elif len(etunimi) > 6:
```

```
    print("Nimesi on yli 6 merkkiä pitkä")
```

```
if "a" in etunimi or "A" in etunimi:
```

```
    print("Nimessäsi on siis a-kirjain")
```

```
else:
```

```
    print("Nimessäsi ei ole a-kirjainta")
```

Tallenna ohjelma KODIT-kansioon nimellä *nimitutkimus*.

Tehtävä 10. Yhteenlaskutesti

Koodaa ohjelma, joka pyytää käyttäjää syöttämään kaksi desimaalilukua (float) ja tallentaa ne muuttujiin *a* ja *b*. Ohjelma kysyy käyttäjältä lukujen summan ja tarkistaa vastauksen oikeellisuuden if-lauseetilla. Oikeasta vastauksesta saa "Oikein", väärästä vastauksesta viestin "Ei mennyt nappiin".

Tallenna ohjelma KODIT-kansioon nimellä *lukusummat*. Ohjelman toiminta on seuraava:

lukusummat

```
a = float(input("Anna luku a: "))
```

```
b = float(input("Anna luku b: "))
```

```
print("Laske", a, "+", b)
```

```
vastaus = float(input("Vastaus: "))
```

```
if vastaus == a+b:
```

```
    print("Oikein!")
```

```
else:
```

```
    print("Ei mennyt nappiin")
```

Tehtävä 11. Itseisarvo, abs

itseisarvovertailu

```
x = int(input("Anna luku x: "))
```

```
y = int(input("Anna luku y: "))
```

```
if abs(x) > abs(y):
```

```
    print("Luku", x, "on itseisarvoltaan suurempi")
```

```
elif abs(y) > abs(x):
```

```
    print("Luku", y, "on itseisarvoltaan suurempi")
```

```
else:
```

```
    print("Luvut", x, y, "ovat itseisarvoltaan yhtä suuret")
```

Tehtävä 12. Simon lukuvertailuissa korjattavaa

Simo on koodannut huolimattomasti alla olevan ohjelman. Ohjelmassa on syntaksivirheitä ja ajatusvirheitä. **Korjaa virheet** ja koodaa ohjelma toimivaksi. Tallenna se KODIT-kansioon nimellä *Simon_lukuvertailu*.

lukuvertailu ja suuruusero

```
from math import *
```

```
x = int(input("Syötä luku x"))
```

```
y = int(input("Syötä luku y"))
```

```
if y <= x:
```

```
    print(y, 'on pienempi tai yhtä suuri kuin', x)
```

```
else if:
```

```
    print(y, 'on suurempi kuin', x)
```

```
print("Lasketaan lukujen suuruusero itseisarvon avulla")
```

```
etaisyys = abs(x-y)
```

```
print("Lukujen suuruusero on:", etaisyys)
```

TEHTÄVIÄ

Tehtävä 13. Arvaa luku, randint

Koodaa ohjelma, joka arpoo satunnaisluvun (int) väliltä 1-10. Seuraavaksi ohjelma pyytää käyttäjää arvaamaan luvun. Ohjelma vertaa käyttäjän antamaa lukua arvottuun lukuun ja kertoo, oliko arvaus oikein, oliko arvaus liian pieni luku tai oliko arvaus liian suuri luku. Satunnaisluvun tutkimiseen tarvitaan if- elif -else -rakennetta.

Tallenna ohjelma KODIT-kansioon nimellä *lukuarvaus*. Testaa ohjelmaa useita kertoja. Millä todennäköisyydellä arvauksesi osuu oikeaan?

```
# random-kirjaston käyttöesimerkkejä
```

```
from random import *
```

```
satluku= randint(1,10)
```

```
arvaus = int(input("Arvaa luku 1-10: "))
```

```
if arvaus == satluku:
```

```
    print ("Oikein meni!")
```

```
elif arvaus < satluku:
```

```
    print ("Arvasit liian pienen luvun")
```

```
    print ("Oikea luku oli",satluku)
```

```
else:
```

```
    print ("Arvasit liian suuren luvun.")
```

```
    print ("Oikea luku oli",satluku)
```

Tehtävä 14. Yhteenlaskutehtävän tarkistus

Edellisellä teoriasivulla on esitetty **yhteenlaskuohjelman** alkuosa. Koodaa ja täydennä ohjelmaan oikeellisuustarkistus. Oikeellisuustarkistuksessa verrataan lukujen summaa käyttäjän antamaan vaihtoehtoon. Tarkistuksen tuloksen mukaan tulostetaan viesti vastauksen oikeellisuudesta. (Vrt. tehtävä 5.)

Tallenna täydennetty ohjelma KODIT-kansioon nimellä *yhteenlaskut*. Testaa ohjelman toiminta useita kertoja.

```
# yhteenlaskuja päässä laskuna
```

```
from random import *
```

```
print ("Lasketaan kahden satunnaisluvun (kokonaislukuja) summaa")
```

```
alku = int(input("Anna kokonaislukuna lukualueen alaraja: "))
```

```
loppu = int(input("Anna kokonaislukuna lukualueen yläaraja: "))
```

```
a, b = randint(alku,loppu), randint(alku,loppu)
```

```
print ("Laske",a,"+",b)
```

```
vastaus = int(input("Vastaus: "))
```

```
if vastaus == a+b:
```

```
    print ("Oikein meni!")
```

```
else:
```

```
    print ("Ei mennyt nappiin!")
```

```
    print ("Oikea vastaus on",a+b)
```

Tehtävä 15. Kaksi arpanoppaa

```
# arpanopat
```

```
from random import *
```

```
print ("Heitetään kahta noppaa")
```

```
valinta = int(input("Yksi kerrallaan = 1, molemmat yhtä aikaa = 2: "))
```

```
if valinta == 1:
```

```
    print ("1. nopan silmäluku: ",randint(1,6))
```

```
    jatka = input("Paina enter...")
```

```
    print ("2. nopan silmäluku: ",randint(1,6))
```

```
elif valinta == 2:
```

```
    print ("Noppien silmäluvut: ",randint(1,6),"ja",randint(1,6))
```

```
else:
```

```
    print ("Valinta ei mahdollinen")
```

Tehtävä 16. Simon kolikonheitto

Simo yrittää koodata kolikonheittopeliä, jossa on kaksi todennäköistä vaihtoehtoa; 1=*kruuna* tai 2=*klaava*. Simon koodissa kaikki ei ole ihan syntaksin mukaisesti. Tehtäväsi on korjata Simon koodi toimivaksi. Tallenna korjattu toimiva ohjelma KODIT-kansioon nimellä *Simon_kolikonheitto*.

```
# kolikonheitto
```

```
from random import *
```

```
valinta = int(input("Arvaa Kruuna=1 tai Klaava=2: "))
```

```
kolikko = randint(1,2)
```

```
if valinta == kolikko:
```

```
    print ("Hienosti arvattu! Nappiin meni!")
```

```
elif valinta != kolikko:
```

```
    print ("Hupsis, ei osunut oikeaan")
```

Tehtävä 17. Merkkijono while-silmukassa

```
# sanasilmukka  
sana = input("Kirjoita jokin sana: ")  
laskuri = 0 # silmukkalaskurin alustaminen  
while (laskuri < len(sana)):  
    print (sana[laskuri])  
    laskuri += 1 # laskurin kasvatus
```

Tehtävä 18. Parilliset luvut ylärajalla

```
# parillisilmukka  
print ("Tulostetaan parillisia lukuja ylärajaan saakka")  
raja = int(input("Anna yläraja: "))  
laskuri = 0 # silmukkalaskurin alustaminen  
while (laskuri <= raja):  
    if laskuri%2==0:  
        print (laskuri)  
        laskuri += 1 # laskurin kasvatus
```

Tehtävä 19. Lukuarvaus silmukassa

```
# silmukkalukuarvaus  
from random import *  
satluku= randint(1,10)  
while True:  
    arvaus = int(input("Arvaa luku 1-10: "))  
    if arvaus < satluku:  
        print ("Arvasit liian pienen luvun")  
    elif arvaus > satluku:  
        print ("Arvasit liian suuren luvun.")  
    else:  
        print ("Arvasit oikein!")  
        break
```

Tehtävä 20. Kertolaskujen opetusohjelma

```
# kertolaskut, opetusohjelma  
from random import *  
lopetus = "e"  
while True:  
    m = randint(1,10)  
    n = randint(1,10)  
    print ("Laske",m,"*",n,"=")  
    vastaus = int(input("vastaus: "))  
    if vastaus == m*n:  
        print ("Oikein!")  
    else:  
        print ("Oikea vastaus on",m*n)  
    lopetus = input("Haluatko lopettaa k/e? ")  
    if lopetus == "k":  
        break
```

Tehtävä 21. Merkkijono for-silmukassa

Koodaa ohjelma, joka kysyy käyttäjältä merkkijonon ja tulostaa sen merkki kerrallaan *for*-silmukassa. Tallenna ohjelma KODIT-kansioon nimellä *merkkisilmukka*. Ohjelman toiminta näyttää seuraavalta:

```
# for in -silmukka

mjonon = input("Kirjoita merkkijono")

for merkki in mjonon:

    print (merkki)
```

Tehtävä 22. Lasketaan a-kirjaimet

```
# merkkilaskuri

lukumaara = 0

mjonon = input("Kirjoita merkkijono: ")

print ("Lasketaan a-kirjainten esiintymät")

for merkki in mjonon:

    if merkki == "a":

        lukumaara += 1

print ("a-kirjaimia löytyi",lukumaara,"kappaletta")
```

Tehtävä 23. Säännöllisiä lukujonoja

Koodaa alla oleva ohjelma, joka tulostaa allekkain parilliset luvut 0:sta 40:een saakka laskusäännön $2*i$ mukaan. Tallenna ohjelma KODIT-kansioon nimellä *silmukkalukujonot*.

```
# silmukkalukujonot

for i in range(21):

    print (2*i+1) # a)

for i in range(21):

    print (i**2) # b)

for i in range(21):

    print (i**2-1) # c)

for i in range(21):

    print (i**2-i) # d)
```

Tehtävä 24. Merkkikuvioita

```
.
.
#####
####
###
##
#

o#####
oo#####
ooo#####
oooo#####
ooooo#####

# for-kuviot

for i in range(1,6): # a)

    print (i*"#")

for i in range(1,6): # b)

    print ((6-i)*"#")

for i in range(1,6): # c)

    print (i*"o" + (6-i)*"#")
```

Tehtävä 25. Murtolukujen yhteen- ja vähennyslaskuja Pythonilla

Oheessa on ohjelmakoodi, joka laskee murtolukujen a ja b yhteen- ja vähennyslaskut ja tulostaa vastaukset. Koodaa ohjelma ja tallenna se KOODIT-kansioon nimellä *murtolukulaskut*.

```
# Murtolukulaskuja
```

```
from fractions import Fraction
```

```
a = Fraction(3,4)
```

```
b = Fraction(6,12)
```

```
print ("Lasketaan murtolukujen",a,"ja",b,"laskutoimituksia")
```

```
print ("Summa", a+b)
```

```
print ("Erotus", a-b)
```

```
print ("Tulo",a*b)
```

```
print ("Osamäärä",a/b)
```

Tehtävä 26. Murtolukujen kerto- ja jakolaskuja

Lisää tehtävää 25 koodattuun murtolukulaskuohjelmaan myös lukujen a ja b **kertolaskulauseke** sekä **jakolaskulauseke**. Lisää koodiin myös näille **tulostuslauseet**. Laske alla esitetyt murtolukujen laskutoimitukset ja tarkista tulokset ohjelmaa käyttäen.

```
# Murtolukulaskuja
```

```
from fractions import Fraction
```

```
a = Fraction(3,4)
```

```
b = Fraction(6,12)
```

```
print ("Lasketaan murtolukujen",a,"ja",b,"laskutoimituksia")
```

```
print ("Summa", a+b)
```

```
print ("Erotus", a-b)
```

```
print ("Tulo",a*b)
```

```
print ("Osamäärä",a/b)
```

Tehtävä 27. Kahden murtoluvun välissä oleva luku

```
# murtoluvun osoittaja ja nimittäjä
```

```
from fractions import Fraction
```

```
a = Fraction(3,5)
```

```
b = Fraction(3,4)
```

```
os1 = a.numerator # osoittaja
```

```
nim1 = a.denominator # nimittäjä
```

```
os2 = b.numerator # osoittaja
```

```
nim2 = b.denominator # nimittäjä
```

```
c = Fraction(os1+os2,nim1+nim2)
```

```
print ("Luku",c,"on lukujen",a,"ja",b,"välissä")
```

Tehtävä 28. Juomatehtailijan pullolaskuri

```
# pullolaskuri
```

```
from fractions import *
```

```
print ("Lasketaan täysien 1/3 l pullojen määrä")
```

```
litrat = int(input("Syötä pullolettava litramäärä: "))
```

```
pulloja = litrat/Fraction(1,3)
```

```
print ("Tarvittavien tyhjien pullojen määrä: ",pulloja)
```

Tehtävä 29. Yhtälönratkaisua Pythonilla

Koodaa esimerkin mukainen yhtälön ratkaisuohjelman. Tallenna ohjelma KODIT-kansioon nimellä *yhtälö*.

Ratkaise ohjelman avulla myös seuraavat yhtälöt. Muuta siis if-lauseen ehtoa yhtälöä vastaavaksi:

- a. $-2x = 8$, $x =$
- b. $8x = x + 63$, $x =$
- c. $4(x-3) = 2(x-4)$, $x =$

Tehtävä 30. Ongelmanratkaisu koodaamalla

Muodosta seuraavissa sanallisissa tehtävissä yhtälö. Koodaa yhtälö ohjelmaan ja etsi ratkaisu suorittamalla ohjelma.

- a. mikä luku kolmella kerrottuna on 129?
- V:
- b. Vilmalla on rahaa 18 euroa enemmän kuin Minjalla. Kuinka paljon on kummallakin, kun heillä on yhteensä 98 euroa?
- V:

Tehtävä 31. Sanallisia tehtäviä koodaamalla

Muodosta seuraavissa sanallisissa tehtävissä yhtälö. Koodaa yhtälö ohjelmaan ja etsi ratkaisu suorittamalla ohjelma.

- a. Karilla on rahaa 16 euroa enemmän kuin Merjalla. Kuinka paljon on kummallakin, kun heillä on yhteensä 48 euroa?
- V:
- b. Jimi on kolme vuotta vanhempi kuin Anne, joka on viisi vuotta vanhempi kuin Sofia. Kuinka vanhoja lapset ovat, kun heidän yhteenlaskettu ikänsä on 28 vuotta?
- V:

Tehtävä 32. Verrantoyhtälö koodaamalla

```
# verranto
for x in range(0,25):
    if x/24 == 5/6:
        print ("Ratkaisu on: ",x)
        break
```

PERUSMATEMATIIKKA OHJELMOIMALLA

matemaattisen ohjelmoinnin
harjoituskirja 9-luokalle

MALLIRATKAISUT

Tehtävä 1. Salasana <pre># salasana passwd = "kaksoiskartio" while True: salasana = input("Anna salasana: ") if salasana == passwd: print("Salasana oikein, tervetuloa!") break else: print("Väärä salasana!")</pre>	Tehtävä 2. Luvun neliö ja kuutio <pre># neliökuutio from random import * lopetus='e' while True: luku = randint(1,15) print("Luvun neliö on",luku**2,"ja kuutio",luku**3) vastaus = int(input("Mikä luku on kyseessä? ")) if vastaus == luku: print("Oikein!") else: print("Ei mennyt oikein. Luku on",luku) lopetus = input("Haluatko lopettaa k/e? ") if lopetus == 'k': break</pre>
Tehtävä 3. Ympyrän pinta-ala ja kehän pituus <pre># ympyrätehtävä from math import * lopetus='e' print("Lasketaan ympyrän ala ja kehän pituus") while True: r = float(input("Anna ympyrän säde: ")) print("Kehän pituus on",round(2*pi*r,2)) print(r,"-säteisen ympyrän ala on",round(pi*r**2,2)) lopetus = input("Haluatko lopettaa k/e? ") if lopetus == 'k': break</pre>	Tehtävä 4. Tarkista syöte neliöjuuressa <pre># neliöjuuri from math import * lopetus='e' while True: print("Lasketaan positiivisen luvun neliöjuuri") luku = float(input("Anna juurettava: ")) if luku < 0: print("Luku ei saa olla negatiivinen") else: print("Luvun neliöjuuri on",sqrt(luku)) lopetus = input("Haluatko lopettaa k/e? ") if lopetus == 'k': break</pre>

Tehtävä 5. Potenssilaskualgoritmi

```
# potenssilasku

lopetus='e'

print ("Lasketaan kokonaislukujen potensseja")

while True:

    k = int(input("Anna kantaluku: "))

    x = int(input("Anna eksponentti: "))

    tulos = 1

    for i in range(x):

        tulos = k*tulos

    print (k,"potenssiin",x,"on",tulos)

    # sama while-silmukalla

    laskuri = 1

    tulos = 1

    while laskuri<=x:

        tulos = k*tulos

        laskuri += 1

    print (k,"potenssiin",x,"on",tulos)

    lopetus = input("Haluatko lopettaa k/e? ")

    if lopetus == 'k':

        break
```

Tehtävä 6. Kymmenlukuja ja binäärilukuja

```
# binääriluku kymmenluvuksi

print ("Muunnetaan binääriluku 10-luvuksi")

while True:

    b = input("Syötä bin.luku, esim 1010. e lopettaa: ")

    if b == "e":

        break

    else:

        print (int(b,2))

print ("Muunnetaan 10-luku binääriluvuksi")

while True:

    n = input("Syötä 10-luku,e lopettaa: ")

    if n == "e":

        break

    else:

        print (bin(int(n)))
```

Tehtävä 7. Kateetin ratkaisu

```
# kateetin ratkaisu

from math import *

print ("Ratkaistaan suorak. kolmion toinen kateetti")

while True:

    b = float(input("Anna toisen kateetin pituus: "))

    c = float(input("Anna hypotenuusan pituus: "))

    a = sqrt(pow(c,2)-pow(b,2))

    print ("Hypotenuusan pituus on",c)

    print ("Kateettien pituudet ovat",round(a,2),"ja",b)

    lopetus = input("Lopetetaanko k/e?: ")

    if lopetus == "k":

        break
```

Tehtävä 8. Simo koodaa Pythagoraan kolmikkoja

```
from math import *

n = int(input("Mikä on kolmikoiden raja c:lle? "))

for a in range(1,n+1):

    for b in range(a+1,n+1):

        for c in range(b+1,n+1):

            if pow(a,2)+pow(b,2) == pow(c,2):

                print (a,b,c)
```

Tehtävä 9. Prosenttiluku

prosenttiluku

print ("Lasketaan prosenttiluku")

while True:

kokonainen = float(input("Anna luku, josta osa otetaan: "))

osa = float(input("Mikä osa kokonaisesta? "))

prosluku = osa/kokonainen*100

print ("Luku",osa,"Luvusta",kokonainen,"on",round(prosluku,1),"%")

lopetus = input("Lopetetaanko k/e?: ")

if lopetus == "k":

break

Tehtävä 10. Prosenttiarvo

prosenttiarvo

print ("Lasketaan prosenttiarvoja rahamäärästä")

while True:

kokonainen = float(input("Anna euromäärä, josta arvo lasketaan: "))

prosenttiluku = float(input("Anna prosenttiluku: "))

prosarvo = prosenttiluku/100*kokonainen

 print (prosenttiluku,"%:a
rahamäärästä",kokonainen,"on",round(prosarvo,1),"€")

lopetus = input("Lopetetaanko k/e?: ")

if lopetus == "k":

break

Tehtävä 11. Muutosprosentti

muutosprosentti

print ("Lasketaan muutosprosentteja")

while True:

alkuperainen = float(input("Anna alkuperäinen arvo: "))

muuttunut = float(input("Anna muuttunut arvo: "))

muutospros = (muuttunut - alkuperainen)/alkuperainen*100

 print ("Muutos
arvosta",alkuperainen,"arvoon",muuttunut,"on",round(muutospros,1),"%:
a")

lopetus = input("Lopetetaanko k/e?: ")

if lopetus == "k":

break

Tehtävä 12. Simon alennus/korotusprosentti

Simon prosenttilaskuri

print ("Ohjelma laskee korotetun/alennetun hinnan")

print ("Syötä korotusprosentti positiivisena ja alennusprosentti negatiivisena
lukuna")

while True:

hinta = float(input("Anna hinta: "))

muutospros = float(input("Anna muutosprosentti: "))

kerroin = muutospros/100 # muutetaan %-luku desimaalikertoimeksi

muutoskerroin = 1.0 + kerroin

uusihinta = muutoskerroin * hinta

print ("Hinta muutoksen jälkeen:",uusihinta)

lopetus = input("Lopetetaanko k/e?: ")

if lopetus == "k":

break

Tehtävä 13. Listaharjoitus	Tehtävä 14. Virheellinen paikkaindeksi
Tehtävä 15. Indeksointi takaperin	Tehtävä 16. Otetaan listasta viipaleita <pre># listat arvosana = [8,6,9,10,7] nimet = ["Kimmo","Virpi","Taina","Jussi","Heli"] lista = [] # tyhjä lista print (arvosana) print (nimet) print ("Listan nimet pituus:",len(nimet)) nimet.append("Timo") arvosana.append(8) print (arvosana) print (nimet) arvosana[1] = 10 arvosana[4] = 10 nimet.append("Sanni") nimet[0] = "Janne" print (nimet) print (arvosana) print (nimet[6]) print (nimet[-1]) print (nimet[-2]) print (nimet[-3]) print (nimet[: -1]) print (nimet[0:4]) print (nimet[2:5]) print (nimet[: : 2])</pre>

Tehtävä 17. Alkioiden syöttäminen listaan

```
# lukulista, tehtävä 17
luvut = [] # tyhjä lista
lopetus = "e"
while True:
    luku = float(input("Syötä luku: "))
    luvut.append(luku)
    lopetus = input("Haluatko lopettaa k/e? ")
    if lopetus == "k":
        break
print ("Luvut:",luvut)
```

Tehtävä 18. Alkioiden summa for-silmukassa

```
# jatkoa edelliseen
# tehtävä 18
summa = 0
for i in luvut:
    summa = summa + i
print ("Listan lukujen summa on:",summa)
```

Tehtävä 19. Alkioiden keskiarvo

```
# tehtävä 19
print ("Listan alkioiden keskiarvo on: ", summa/len(luvut))
```

Tehtävä 20. Alkioiden vaihteluvälin pituus

```
# tehtävä 20
print ("Listan vaihteluväli on: ",max(luvut)- min(luvut))
```

Tehtävä 21. Kolmen luvun summa ja erotus aliohjelmina

Tehtävä 21, proseduurit

def summa(a,b,c):

print("Lukujen summa", a+b+c)

lukujen erotus

def erotus(a,b,c):

print("Lukujen erotus",a-b-c)

pääohjelma

n = int(input("Anna kokonaisluku: "))

s = int(input("Anna toinen kokonaisluku: "))

t = int(input("Anna kolmas kokonaisluku: "))

summa(n,s,t)

erotus(n,s,t)

keskiarvo(n,s,t)

print("Aliohjelmat suoritettu")

Tehtävä 22. Kolmen luvun keskiarvo aliohjelma

Tehtävä 22,lukujen keskiarvo

def keskiarvo(a,b,c):

print ("Lukujen keskiarvo",(a+b+c)/3)

Tehtävä 23. Jaollisuustutkimus

jaollisuustutka

def jaolliset(luku):

#while-silmukalla

j=1

while j<=100:

if j%luku==0:

print (j)

j+=1

for-silmukalla

for i in range(1,101):

if i%luku==0:

print (i)

pääohjelma

while True:

lopetus = "e"

luku = int(input("Anna luku: "))

print ("Tulostetaan kaikki",luku,"jaolliset 100:n asti")

jaolliset(luku)

lopetus = input("Lopetetaanko? k/e ")

if lopetus == "k":

break

Tehtävä 24. Simon K13 laskuri

Simon K13 laskuri

def tervehdys(nimi):

print ("Hei",nimi)

def ikaraja(svuosi, nyt):

ika = nyt - svuosi

if ika < 13:

print ("Et voi katsoa K13 elokuvia")

else:

print ("Voit katsoa K13 elokuvia")

pääohjelma

while True:

lopetus = "e"

nimi = input("Kerro nimesi? ")

nyt = int(input("Mikä vuosi on nyt? "))

svuosi = int(input("Minä vuonna olet syntynyt? "))

tervehdys(nimi)

print ("Selvitetään ikäraja...")

ikaraja(svuosi,nyt)

lopetus = input("Lopetetaanko? k/e ")

if lopetus == "k":

break

Tehtävä 25. Suorakulmainen särmiö

```
# suorakulmainen särmiö
def tilavuus(a,b,c):
    return(a*b*c)

def kokpa(a,b,c):
    return (2*(a*b+b*c+a*c))

# pääohjelma
print (""""Lasketaan suorakulmaisen särmiön
tilavuus ja kokonaispinta-ala. Anna mitat""")
l = float(input("leveys: "))
p = float(input("pituus: "))
k = float(input("korkeus: "))
# funktiokutsut
print("Särmiön tilavuus on",tilavuus(l,p,k))
print(" ja kok. pinta-ala", kokpa(l,p,k))
```

Tehtävä 26. Suorakulmainen kolmio ja sen pinta-ala

```
# kolmiolaskut
def onkosuorakulmainen(a,b,c):
    if (a**2+b**2 == c**2):
        return True
    else:
        return False

def piiri(a,b,c):
    return round(a+b+c,2)

def pinta_ala(a,b):
    return round(a*b/2,2)

# pääohjelma
print (""""Lasketaan kolmion piiri ja, jos se on
suorakulmainen, niin myös pinta-ala. Syötä sivujen pituudet""")
a = float(input("a: "))
b = float(input("b: "))
c = float(input("c: "))
# funktiokutsut
print ("Kolmion piiri",piiri(a,b,c))
if onkosuorakulmainen(a,b,c):
    print ("Pinta-ala", pinta_ala(a,b))
else:
    print("Pinta-alaa ei voida laskea näillä tiedoilla")
```

Tehtävä 27. Lukulistan minimi, maksimi ja keskiarvo funktioina

```
# listafunktiot
def pienin(lista):
    return min(lista)

def suurin(lista):
    return max(lista)

def keskiarvo(lista):
    summa = 0
    for i in lista:
        summa = summa + i
    return round(summa/len(lista),2)
```

Tehtävä 28. Kertoma n!

```
# kertoma
n = int(input("Minkä luvun kertoma lasketaan: "))
laskuri = 1
kertoma = 1
while laskuri <= n:
    if n == 0:
        print ("0! = 1")
        break
    else:
        kertoma = laskuri * kertoma
        laskuri += 1
print ("Luvun",n,"kertoma on",kertoma)
```

```
# pääohjelma
luvut = [] # tyhjä lista
lopetus = "e"
while True:
    luku = float(input("Syötä luku: "))
    luvut.append(luku)
    lopetus = input("Haluatko lopettaa k/e? ")
    if lopetus == "k":
        break
print ("Listan pienin luku on", pienin(luvut))
print ("Listan suurin luku on", suurin(luvut))
print ("Lukujen keskiarvo on", keskiarvo(luvut))
```

Tehtävä 29. Tehtävän pääohjelma

```
# pääohjelma
while True:
    print ("Lasketaan ympyräkappaleen tilavuus ja pinta-ala")
    kappale = input("Valitse kappale: ")
    l = ymp.lieriö, k = ymp.kartio, p = pallo
    if kappale == "l":
        r = float(input("Anna lieriön pohjan säde: "))
        h = float(input("Anna lieriön korkeus: "))
        print ("Lieriön tilavuus on", round(lieriön_tilavuus(r,h),2))
        print ("Lieriön vaipan pinta-ala on", round(lieriön_vaipan_ala(r,h),2))
        print ("Kokonaispinta-ala on", round(lieriön_vaipan_ala(r,h) + 2*ympyrän_ala(r),2))

    if kappale == "k":
        r = float(input("Anna kartion pohjan säde: "))
        h = float(input("Anna kartion korkeus: "))
        print ("Kartion tilavuus on", round(kartion_tilavuus(r,h),2))
        print ("Kartion vaipan pinta-ala on", round(kartion_vaipan_ala(r,h),2))
        print ("Kokonaispinta-ala on", round(kartion_vaipan_ala(r,h) + ympyrän_ala(r),2))

    if kappale == "p":
        r = float(input("Anna pallon säde: "))
        print ("Pallon tilavuus on", round(pallon_tilavuus(r),2))
        print ("Pallon pinta-ala on", round(pallon_ala(r),2))

    lopetus=input("Haluatko lopettaa k/e? ")
    if lopetus == "k":
        break
```

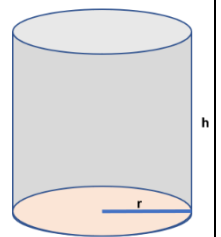
Tehtävä 30. Ympyrälieriö

```
# ympyräkappaleet
from math import * # tarvitaan pii

def ympyrän_ala(r):
    return pi*pow(r,2)

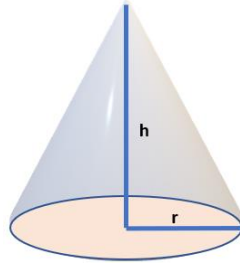
def lieriön_tilavuus(r,h):
    return ympyrän_ala(r)*h

def lieriön_vaipan_ala(r,h):
    return 2*pi*r*h
```

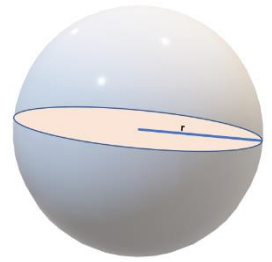


Tehtävä 31. Ympyräkartio

```
def kartion_tilavuus(r,h):  
    return lieriön_tilavuus(r,h)/3  
  
def kartion_vaipan_ala(r,h):  
    return pi*r*sqrt(pow(r,2)+pow(h,2))
```

**Tehtävä 32. Pallo**

```
def pallon_tilavuus(r):  
    return 4*pi*pow(r,3)/3  
  
def pallon_ala(r):  
    return 4*pi*pow(r,2)
```



Tehtävä 33. Koulutodistus

```
# oppiaineiden arvosanat
from statistics import *
aineet = ["MA", "AI", "FY", "KE", "BI", "GE", "HY", "EA", "RU"]
arvosanat = [9, 8, 9, 7, 10, 8, 8, 7, 8] # arvosanat
# tulostetaan aine ja arvosana
for i in range(0, len(aineet)):
    print(aineet[i], arvosanat[i])
print("Arvosanojen keskiarvo", round(mean(arvosanat), 2))
```

Tehtävä 34. Jatkoa koulutodistukseen

```
print("Arvosanojen mediaani on", median(arvosanat))
print("Tyyppiarvo eli moodi on", mode(arvosanat))
```

Tehtävä 35. Tikkaohjelma

```
# tikkakisa
from statistics import *
tikat = [0, 0, 0, 0, 0]
for i in range(5):
    print(i+1, ".tikka")
    tikat[i] = int(input("osuma: "))
print("Osumat:", tikat)
print("Keskiarvo:", mean(tikat))
print("Pienin:", min(tikat), "Suurin:", max(tikat))
print("Vaihteluvälin pituus", max(tikat) - min(tikat))
```

Tehtävä 36. Simon matematiikan kokeet

```
# Simon matematiikan kokeet
from statistics import *
koe = ["Polynomit", "Trigonometria", "Geometria", "Tilastot", "Kertaus"]
arvosana = [9.25, 8.5, 10, 7.75, 8.75]
i = 0
while i < len(koe):
    print(koe[i], arvosana[i])
    i += 1
print("-----")
print("Keskiarvo", mean(arvosana))
```

TEHTÄVIÄ

Tehtävä 37. Kivi, paperi, sakset

```
# kolikonheitto

from random import *

valinta = int(input("Arvaa Kruuna=1 tai Klaava=2: "))

kolikko = randint(1,2)

if valinta == kolikko:

    print ("Hienosti arvattu! Nappiin meni!")

elif valinta != kolikko:

    print ("Hupsis, ei osunut oikeaan")
```

Tehtävä 38. Arpanopan silmäluvun todennäköisyys

```
# arpanoppatutkimus

from random import *

heittoja = int(input("Kuinka monta heittoa? "))

suotuisia = 0 # alustus

valittu = int(input("Valitse silmäluku 1 - 6: "))

for i in range(heittoja):

    silmaluku = randint(1,6)

    if silmaluku == valittu:

        suotuisia += 1

print ("Silmälukua",valittu,"""
tuli""",suotuisia,"kappaletta")

print ("Suotuisien suhde kaikkiin heittoihin:",suotuisia/heittoja)
```

Tehtävä 39. Silmälukujen todennäköisyysjakauma

```
# silmälukujen tod.näköisyysjakauma

from random import *

lukumäärät = [0,0,0,0,0,0]

for i in range(1000000):

    silmaluku = randint(1,6)

    lukumäärät[silmaluku-1] += 1

print ("Silmälukujen todennäköisyydet")

a = 0

while a < len(lukumäärät):

    print ("Silmäluku",a+1,"tod.näköisyys on",lukumäärät[a]/1000000)

    a += 1
```

Tehtävä 40. Kaksi arpanoppaa

```
# kaksi noppaa

from random import *

lukumäärät = [0,0,0,0,0,0,0,0,0,0]

for i in range(10000):

    noppa1 = randint(1,6)

    noppa2 = randint(1,6)

    summa = noppa1 + noppa2

    lukumäärät[summa-2] += 1

print (lukumäärät)
```

