

13.1

Kirjoitetaan ohjelma.

```
#Ohjelma kysyy käyttäjältä positiivisen kokonaisluvun n,
#laskee tulon luvun n kertoman ja tulostaa sen näytölle.
#Ohjelma tarkastaa, onko käyttäjän antama kokonaisluku
#positiivinen.

print("Ohjelma laskee luvun n kertoman.")
#Kysytään luku n
n = int(input("Anna positiivinen kokonaisluku n: "))
print()

#Tutkitaan, onko käyttäjän syöte virheellinen.
#Toistetaan niin kauan kuin käyttäjän antama luku  $n \leq 0$ .
while n <= 0:
    print("Et antanut positiivista kokonaislukua.")
    n = int(input("Anna positiivinen kokonaisluku n: "))
    print()

#Asetetaan muuttujan kertoma alkuarvoksi 1.
kertoma = 1

#Silmukka käy läpi luvut n, ..., 1.
for i in range(n, 0, -1):
    kertoma *= i

#Tulostetaan kertoma
print("Luvun kertoma:", kertoma)
```

13.2

Kirjoitetaan ohjelma.

```
#Ohjelma kysyy käyttäjältä positiivisen kokonaisluvun n,
#laskee n:n ensimmäisen parillisen positiivisen
#kokonaisluvun summan ja tulostaa sen näytölle.
#Ohjelma tarkastaa, onko käyttäjän antama kokonaisluku
#positiivinen.

print("Ohjelma laskee n:n ensimmäisen parillisen")
print("positiivisen kokonaisluvun summan.")
print() #Tulostaa tyhjän rivin.

#Kysytään luku n
n = int(input("Anna positiivinen kokonaisluku n: "))

#Tutkitaan, onko käyttäjän syöte virheellinen.
#Toistetaan niin kauan kuin käyttäjän antama luku  $n \leq 0$ .
while n <= 0:
    print("Et antanut positiivista kokonaislukua.")
    n = int(input("Anna positiivinen kokonaisluku n: "))

#Asetetaan muuttujan summa alkuarvoksi 0.
summa = 0

#Silmukka käy läpi luvut 1, ..., n.
for i in range(1, n + 1):
    summa += 2 * i

#Tulostetaan summa.
print("Lukujen summa:", summa)
```

13.3

Kirjoitetaan ohjelma.

```
#Ohjelma ilmoittaa, onko käyttäjän syöttämä luku alkuluku.

import math #Otetaan käyttöön moduuli math.

print("Ohjelma ilmoittaa, onko syöttämäsi luku alkuluku.")
print("Ohjelma tutkii lukua 1 suurempia kokonaislukuja.")

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

#Tutkitaan, onko käyttäjän syöte virheellinen.
#Toistetaan, kunnes käyttäjä antaa lukua 1 suuremman luvun.

while luku <= 1:
    print("Annoit virheellisen syötteen.")
    luku = int(input("Anna tutkittava luku: "))

#Pyöristetään neliöjuuri alaspäin lähimpään kokonaislukuun.
loppu = math.isqrt(luku)

#Asetetaan muuttujan arvoksi totuusarvo True.
alkuluku = True

#Silmukka käy läpi luvut 2, ..., loppu.
for i in range(2, loppu + 1):
    #Tutkitaan, onko luku jaollinen luvulla i.
    if luku % i == 0:
        #jos jaollinen, asetetaan totuusarvoksi False
        alkuluku = False
        break #Lopettaa silmukan suorituksen.

#Tutkitaan, onko muuttujan alkuluku arvo True.
if alkuluku == True:
    print(f"Luku {luku} on alkuluku.")
else:
    print(f"Luku {luku} on jaollinen ainakin luvulla {i}.")
    print(f"Luku {luku} ei ole alkuluku.")
```

Huomaa, että ohjelman lopussa tulostukseen on käytetty f-strings-tulostusta, jolla tulostus saadaan muotoiltua kauniimmin. Ennen lainausmerkkejä tulostettavan tekstin alkuun kirjoitetaan kirjain f ja tulostukseen halutut muuttujien arvot kirjoitetaan aaltosulkeiden sisään.

13.4

Tutkitaan ohjelmia kirjoittamalla ja suorittamalla ne.

Ohjelma 1:

```
for i in range(1, 11):  
    if i % 3 == 0:  
        break  
    print(i)  
print("Suoritus päättyi.")
```

Kun ohjelma suoritetaan, se tulostaa:

```
1  
2  
Suoritus päättyi.
```

Silmukka jättää tulostamatta luvut 3–11.

Varattu sana `break` lopettaa silmukan suorituksen, ja ohjelman suoritus jatkuu silmukan jälkeen tulevasta lauseesta.

Ohjelma 2:

```
for i in range(1, 11):  
    if i % 3 == 0:  
        continue  
    print(i)  
print("Suoritus päättyi.")
```

Kun ohjelma suoritetaan, se tulostaa:

```
1  
2  
4  
5  
7  
8  
10  
Suoritus päättyi.
```

Silmukka jättää tulostamatta luvut 3, 6 ja 9.

Varattu sana `continue` keskeyttää silmukan meneillään olevan kierroksen suorituksen, ja ohjelman suoritus jatkuu silmukan seuraavan kierroksen alusta.

13.5

Kirjoitetaan ohjelma.

```
#Ohjelma lajittelee listan luvut suurimmasta pienimpään.
#Lajittelu toteutetaan sisäkkäisillä silmukoilla.

lista = [1, 4, 82, 18, 3, 1]

print("Lajiteltava lista:", lista)

pituus = len(lista) #Selvitetään listan pituus.

#Uloppe silmukka käy läpi luvut 0, ..., pituus - 2.
for i in range(pituus - 1):
    #Sisempi silmukka käy läpi luvut i + 1, ..., pituus - 1.
    for j in range(i+1, pituus):
        #Onko luku paikalla i suurempi kuin luku paikalla j?
        if lista[i] < lista[j]:
            #Vaihdetaan lukujen järjestystä.
            apu = lista[i]
            lista[i] = lista[j]
            lista[j] = apu

print("Lajiteltu lista:", lista)
```

13.6

Kirjoitetaan ohjelma.

```
#Ohjelma kysyy käyttäjältä 5 kokonaislukua, lisää ne
#listaan ja tulostaa listan. Ohjelma lajittelee listan
#luvut suurimmasta pienimpään. Lajittelu toteutetaan
#sisäkkäisillä silmukoilla.

lista = []

print("Ohjelma lajittelee listan alkiot suurimmasta pienimpään.")
print("Anna 5 lukua, jotka haluat lajitella.")

for i in range(5):
    luku = int(input("Anna luku: "))
    lista.append(luku)

print("Lajiteltava lista:", lista)

pituus = len(lista) #Selvitetään listan pituus.

#Ulompi silmukka käy läpi luvut 0, ..., pituus - 2.
for i in range(pituus - 1):
    #Sisempi silmukka käy läpi luvut i + 1, ..., pituus - 1.
    for j in range(i+1, pituus):
        #Onko luku paikalla i suurempi kuin luku paikalla j?
        if lista[i] < lista[j]:
            #Vaihdetaan lukujen järjestystä.
            apu = lista[i]
            lista[i] = lista[j]
            lista[j] = apu

print("Lajiteltu lista:", lista)
```

13.7

Kirjoitetaan ohjelma.

```
#Ohjelma kysyy käyttäjältä positiivisen kokonaisluvun n,
#laskee n:n ensimmäisen parillisen positiivisen
#kokonaisluvun summan ja tulostaa sen näytölle.
#Ohjelma tarkastaa, onko käyttäjän antama kokonaisluku
#positiivinen.

print("Ohjelma laskee n:n ensimmäisen parillisen")
print("positiivisen kokonaisluvun summan.")
print() #Tulostaa tyhjän rivin.

#Kysytään luku n
n = int(input("Anna positiivinen kokonaisluku n: "))

#Tutkitaan, onko käyttäjän syöte virheellinen.
#Toistetaan niin kauan kuin käyttäjän antama luku  $n \leq 0$ .
while n <= 0:
    print("Et antanut positiivista kokonaislukua.")
    n = int(input("Anna positiivinen kokonaisluku n: "))

#Asetetaan muuttujan summa alkuarvoksi 0.
summa = 0

#Silmukka käy läpi luvut 1, ..., n.
for i in range(1, n + 1):
    summa += 2 * i

#Tulostetaan summa.
print("Lukujen summa:", summa)
```

13.8

Kirjoitetaan ohjelma.

```
#Ohjelma kysyy käyttäjältä kaksi vuosilukua ja tulostaa kaikki
#karkausvuodet, jotka ovat näiden vuosien välissä. Ohjelma
#tulostaa lopuksi karkausvuosien lukumäärän.

print("Ohjelma tulostaa annettujen vuosien välissä olevat karkausvuodet.")
print("Lopuksi ohjelma tulostaa karkausvuosien lukumäärän.")

vuosi_1 = int(input("Anna ensimmäinen vuosiluku: "))
vuosi_2 = int(input("Anna toinen vuosiluku: "))

summa = 0

#Tutkitaan, onko käyttäjän syöttämät vuosiluvut oikeassa järjestyksessä.
if vuosi_1 > vuosi_2:
    apu = vuosi_1
    vuosi_1 = vuosi_2
    vuosi_2 = apu

#Silmukka käy läpi luvut vuosi_1, ..., vuosi_2.
for i in range(vuosi_1, vuosi_2 + 1):
    #Tutkitaan, onko vuosi karkausvuosi.
    if (i % 100 == 0 and i % 400 == 0) or (i % 4 == 0 and i % 100 != 0):
        print(i)
        summa += 1

print("Karkausvuosien lukumäärä:", summa)
```

13.9

Kirjoitetaan ohjelma.

```
#Ohjelma kysyy käyttäjältä eksponentin k ja ilmoittaa, onko
#luku 2 ** k - 1 Mersennen alkuluku.

import math #Otetaan käyttöön moduuli math.

#Samalle riville tulostettavan tekstin voi jakaa koodissa
#useammalle riville +-operaattorin avulla.
print("Ohjelma ilmoittaa, onko Mersennen luku 2**k - 1 " +
      "Mersennen alkuluku.")
print("Tutkitaan tapauksia, joissa eksponentti k kuuluu " +
      "joukkoon {2, 3,..., 50}.")

luku_k = int(input("Anna eksponentti k: "))

#Tutkitaan, onko käyttäjän syöte virheellinen.
#Toistetaan niin kauan kuin käyttäjä antaa virheellisen arvon.
while luku_k < 2 or luku_k > 50:
    print("Annoit virheellisen syötteen.")
    luku_k = int(input("Anna luku k: "))

luku = 2**luku_k - 1

#Pyöristetään neliöjuuri alaspäin lähimpään kokonaislukuun.
loppu = math.isqrt(luku)

alkuluku = True #Asetetaan muuttujan arvoksi totuusarvo True.

#Silmukka käy läpi luvut 2, ..., loppu.
for i in range(2, loppu + 1):
    #Tutkitaan, onko luku jaollinen luvulla i.
    if luku % i == 0:
        #Jos jaollinen, asetetaan totuusarvoksi False.
        alkuluku = False
        break #Keskeyttää silmukan suorituksen.

#Tutkitaan, onko muuttujan alkuluku arvo True.
if alkuluku == True:
    print(f"Luku 2 **{luku_k} - 1 on Mersennen alkuluku.")
else:
    print(f"Luku 2**{luku_k} - 1 ei ole Mersennen alkuluku.")
```

Huomaa, että ohjelman lopussa tulostukseen on käytetty f-strings-tulostusta, jolla tulostus saadaan muotoiltua kauniimmin. Ennen lainausmerkkejä tulostettavan tekstin alkuun kirjoitetaan kirjain f ja tulostukseen halutut muuttujien arvot kirjoitetaan aaltosulkeiden sisään.

13.10

Kirjoitetaan ohjelma.

```
#Ohjelma kysyy käyttäjältä lajiteltavien kokonaislukujen
#määrän, kysyy luvut, lisää ne listaan ja tulostaa listan.
#Ohjelma lajittelee listan luvut pienimmästä suurimpaan ja
#tulostaa listan. Ohjelma kääntää lajitellun listan suurimmasta
#pienimpään ja tulostaa listan.
```

```
lista = []
```

```
lkm = int(input("Kuinka monta lukua haluat lajitella? "))
```

```
#Kysytään luvut. Silmukka käy läpi luvut 0, ..., lkm - 1
#eli silmukka suoritetaan yhteensä lkm kertaa.
```

```
for i in range(lkm):
    luku = int(input("Anna luku: "))
    lista.append(luku)
```

```
print("Lajiteltava lista:", lista)
```

```
pituus = len(lista) #Selvitetään listan pituus.
```

```
#Ulompi silmukka käy läpi luvut 0, ..., pituus - 2.
```

```
for i in range(pituus - 1):
    #Sisempi silmukka käy läpi luvut i + 1, ..., pituus - 1.
    for j in range(i+1, pituus):
        #Onko luku paikalla i suurempi kuin luku paikalla j?
        if lista[i] > lista[j]:
            #Vaihdetaan lukujen järjestystä.
            apu = lista[i]
            lista[i] = lista[j]
            lista[j] = apu
```

```
print("Lajiteltu lista:", lista)
```

```
#Käännetään lista.
```

```
lista.reverse()
print("Käännetty lista:", lista)
```

13.11

- a) Tutkitaan ohjelmia kirjoittamalla ja suorittamalla ne. Huomaa, että kirjan 1. painoksessa ohjelmien 1 ja 2 sisennykset ovat virheelliset.

```
1) for i in range(5):  
    for j in range (i + 1):  
        print("*")
```

Kun ohjelma suoritetaan, se tulostaa:

```
*  
*  
*  
*  
*  
*  
*  
*  
*  
*  
*  
*  
*  
*  
*
```

```
2) for i in range(5):  
    for j in range(i + 1):  
        print("*", end = "")
```

Kun ohjelma suoritetaan, se tulostaa:

```
*****
```

```
3) for i in range(5):  
    for j in range(i + 1):  
        print("*", end = "")  
    print()
```

Kun ohjelma suoritetaan, se tulostaa:

```
*  
**  
***  
****  
*****
```

b) Kirjoitetaan ohjelma.

```
#Ohjelma tulostaa:
#*****
#*****
#*****
#***
#**
#*

#Silmukka käy läpi luvut 6, ..., 1.
for i in range(6,0,-1):
    #Silmukka tulostaa i tähteä samalle riville.
    for j in range(i):
        print("*", end = "")
    #rivinvaihto
    print()
```

13.12

Kirjoitetaan ohjelma.

```
#Ohjelma määrittää piin likiarvon neliön ja neljäsosaympyrän
#avulla. Neliön pisteet määräytyvät epäyhtälöistä  $0 \leq x \leq 1$  ja
# $0 \leq y \leq 1$ , neljäsosaympyrän epäyhtälöistä  $0 \leq x \leq 1$ ,
# $0 \leq y \leq 1$  ja  $x^2 + y^2 \leq 1$ .
#Ohjelma kysyy käyttäjältä arvottavien pisteiden lukumäärän,
#arpoo neliön pisteiden x- ja y-koordinaatit, laskee kuinka
#moni pisteistä on myös neljäsosaympyrän piste ja tulostaa
#piin likiarvon 2 desimaalin tarkkuudella.

import random

print("Ohjelma määrittää piin likiarvon.")

lkm = int(input("Kuinka monta pistettä arvotaan? "))
osumat = 0

#Silmukka suoritetaan lkm kertaa.
for i in range(lkm):
    #Arvotaan pisteen x- ja y-koordinaatit.
    x = random.uniform(0, 1)
    y = random.uniform(0, 1)
    #Tutkitaan, onko piste neljännesympyrän piste.
    if x ** 2 + y ** 2 <= 1:
        osumat += 1

likiarvo = osumat / lkm * 4

print("Piin likiarvo 2 desimaalin tarkkuudella:", round(likiarvo, 2))
```

13.13

Kirjoitetaan ohjelma.

```
#Ohjelma selvittää, kuinka monella tavalla positiivinen
#kokonaisluku n voidaan ilmoittaa kolmen eri positiivisen
#kokonaisluvun summana. Ohjelma kysyy käyttäjältä luvun n,
#tulostaa kaikki mahdolliset summat sekä summien lukumäärän.

print("Ohjelma selvittää, kuinka monella tavalla positiivinen "+"
      "kokonaisluku n voidaan")
print("ilmoittaa kolmen eri positiivisen kokonaisluvun summana.")
luku = int(input("Anna kokonaisluku: "))
summa = 0
laskuri = 0

#Silmukka käy läpi luvut 1, ..., luku.
for i in range(1, luku + 1):
    #Silmukka käy läpi luvut i + 1, ..., luku.
    for j in range(i + 1, luku + 1):
        #Silmukka käy läpi luvut j + 1, ..., luku.
        for k in range(j + 1, luku + 1):
            #Lasketaan lukujen summa.
            summa = i + j + k
            #Tutkitaan, onko summa sama kuin luku.
            if summa == luku:
                print(i, "+", j, "+", k, "=", luku)
                laskuri += 1
print("Summia löytyi", laskuri, "kappaletta.")
```

13.14

Kirjoitetaan ohjelma.

```
#Ohjelma tulostaa täydelliset luvut, jotka ovat pienempiä
#kuin 10 000.

print("Täydelliset luvut, joka ovat pienempiä kuin 10 000:")

#Silmukka käy läpi luvut 1, ..., 9999.
for i in range(1, 10000):
    summa = 0
    #Silmukka käy läpi luvut 1, ..., i.
    for j in range(1, i):
        #Tutkitaan, onko luku j luvun i tekijä.
        if i % j == 0:
            summa += j
    #Tutkitaan, onko tekijöiden summa yhtä suuri kuin luku i.
    if summa == i:
        print(i)
```

13.15

Kirjoitetaan ohjelma.

```
#Ohjelma kysyy käyttäjältä merkkijonon ja tulostaa sen käännettynä
#poistaen samalla kaikki välilyönnit. Ohjelma kysyy, haluaako
#käyttäjää jatkaa.
```

```
#Asetetaan muuttujan jatketaanko alkuarvoksi k.
jatketaanko = "k"
```

```
#Toistetaan niin kauan kuin käyttäjä haluaa jatkaa.
while jatketaanko == "k":
    print("Palindromiohjelma")
    merkkijono = input("anna merkkijono: ")
    print(merkkijono)
    lista = []
    #Luetaan silmukassa merkkijonoa merkki kerrallaan.
    for i in merkkijono:
        #Tutkitaan, onko merkki jokin muu kuin välilyönti.
        if i != " ":
            #Lisätään merkki listaan.
            lista.append(i)
    #Käännetään lista.
    lista.reverse()
    merkkijono2 = ""
    #Luetaan silmukassa listaa merkki kerrallaan.
    for j in lista:
        #Lisätään merkki uuteen merkkijonoon.
        merkkijono2 += j

    print("on käännettynä")
    print(merkkijono2)
    print()
    jatketaanko = input("Jatketaanko k/e? ")
print()
print("Ohjelman suoritus päättyi")
```