

# Programmeerimine

9. loeng

# Täna loengus

Kordamine

# Kordamine

- Muutujate skoop
- Kahekordsed tsüklid
- Listid
- Funktsioonid

# Vaheeksami näidisülesandeid

- Arvude arv

## Vaheeksami näidisülesandeid

- Arvude arv

# Vaheeksami näidisülesandeid

- Arvude arv

## 1. Arvude arv (5p)

Kirjuta funktsioon `arvude_arv`, mis võtab argumentiks sõne, mis koosneb semikoolonitega eraldatud komponentidest ja tagastab täisarvu, mis ütleb, mitu antud komponenti kujutavad mingit märgita täisarvu.

Näited:

```
>>> arvude_arv("aartaa;bbbb;234;45;ccwec")
2
>>> arvude_arv("aaaa;bbbb;ee;rrr;ccc")
0
>>> arvude_arv("b1aa3;b123aa")
0
>>> arvude_arv("3")
1
>>> arvude_arv("")
0
```

# Vaheeksami näidisülesandeid

- Arvude arv

## 1. Arvude arv (5p)

Kirjuta funktsioon `arvude_arv`, mis võtab argumentiks sõne, mis koosneb semikoolonitega eraldatud komponentidest ja tagastab täisarvu, mis ütleb, mitu antud komponenti kujutavad mingit märgita täisarvu.

Näited:

```
>>> arvude_arv("aartaa;bbbb;234;45;ccwec")
2
>>> arvude_arv("aaaa;bbbb;ee;rrr;ccc")
0
>>> arvude_arv("b1aa3;b123aa")
0
>>> arvude_arv("3")
1
>>> arvude_arv("")
0
```

# Vaheeksami näidisülesandeid

- Arvude arv

## 1. Arvude arv (5p)

Kirjuta funktsioon `arvude_arv`, mis võtab argumendiks sõne, mis koosneb semikoolonitega eraldatud komponentidest ja tagastab täisarvu, mis ütleb, mitu antud komponenti kujutavad mingit märgita täisarvu.

Näited:

```
>>> arvude_arv("aartaa;bbbb;234;45;ccwec")
2
>>> arvude_arv("aaaa;bbbb;ee;rrr;ccc")
0
>>> arvude_arv("blaa3;b123aa")
0
>>> arvude_arv("3")
1
>>> arvude_arv("")
0
```

```
def arvude_arv(s):
    li = s.split(';')
    print(li)
```

```
arvude_arv("aartaa;bbbb;234;45;ccwec")
```

# Vaheeksami näidisülesandeid

- Arvude arv

## 1. Arvude arv (5p)

Kirjuta funktsioon `arvude_arv`, mis võtab argumendiks sõne, mis koosneb semikoolonitega eraldatud komponentidest ja tagastab täisarvu, mis ütleb, mitu antud komponenti kujutavad mingit märgita täisarvu.

Näited:

```
>>> arvude_arv("aartaa;bbbb;234;45;ccwec")
2
>>> arvude_arv("aaaa;bbbb;ee;rrr;ccc")
0
>>> arvude_arv("blaa3;b123aa")
0
>>> arvude_arv("3")
1
>>> arvude_arv("")
0
```

```
def arvude_arv(s):
    li = s.split(';')
    print(li)
```

```
arvude_arv("aartaa;bbbb;234;45;ccwec")
```

# Vaheeksami näidisülesandeid

- Arvude arv

## 1. Arvude arv (5p)

Kirjuta funktsioon `arvude_arv`, mis võtab argumendiks sõne, mis koosneb semikoolonitega eraldatud komponentidest ja tagastab täisarvu, mis ütleb, mitu antud komponenti kujutavad mingit märgita täisarvu.

Näited:

```
>>> arvude_arv("aartaa;bbbb;234;45;ccwec")
2
>>> arvude_arv("aaaa;bbbb;ee;rrr;ccc")
0
>>> arvude_arv("blaa3;b123aa")
0
>>> arvude_arv("3")
1
>>> arvude_arv("")
0
```

```
def arvude_arv(s):
    li = s.split(';')
    print(li)
```

```
arvude_arv("aartaa;bbbb;234;45;ccwec")
```

```
['aartaa', 'bbbb', '234', '45', 'ccwec']
```

## Vaheeksami näidisülesandeid

- Arvude arv

## Vaheeksami näidisülesandeid

- Arvude arv

# Vaheeksami näidisülesandeid

- Arvude arv

```
def arvude_arv(s):  
    li = s.split(';')  
    mitu = 0  
    for el in li:  
        if el.isnumeric():  
            mitu += 1  
            print(mitu)  
  
print(arvude_arv("aartaa;bbbb;234;45;ccwec"))
```

# Vaheeksami näidisülesandeid

- Arvude arv

```
def arvude_arv(s):  
    li = s.split(';')  
    mitu = 0  
    for el in li:  
        if el.isnumeric():  
            mitu += 1  
            print(mitu)  
  
print(arvude_arv("aartaa;bbbb;234;45;ccwec"))
```

# Vaheeksami näidisülesandeid

- Arvude arv

```
def arvude_arv(s):  
    li = s.split(';')  
    mitu = 0  
    for el in li:  
        if el.isnumeric():  
            mitu += 1  
            print(mitu)  
  
print(arvude_arv("aartaa;bbbb;234;45;ccwec"))
```

1

2

None

# Vaheeksami näidisülesandeid

- Arvude arv

```
def arvude_arv(s):  
    li = s.split(';')  
    mitu = 0  
    for el in li:  
        if el.isnumeric():  
            mitu += 1  
            print(mitu)  
  
print(arvude_arv("aartaa;bbbb;234;45;ccwec"))
```

1

2

None

# Vaheeksami näidisülesandeid

- Arvude arv

```
def arvude_arv(s):  
    li = s.split(';')  
    mitu = 0  
    for el in li:  
        if el.isnumeric():  
            mitu += 1  
            print(mitu)  
  
print(arvude_arv("aartaa;bbbb;234;45;ccwec"))
```

1

2

None

# Vaheeksami näidisülesandeid

- Arvude arv

```
def arvude_arv(s):  
    li = s.split(';')  
    mitu = 0  
    for el in li:  
        if el.isnumeric():  
            mitu += 1  
            print(mitu)  
  
print(arvude_arv("aartaa;bbbb;234;45;ccwec"))
```

1

2

None

Ei ole õige!

## Vaheeksami näidisülesandeid

- Arvude arv

## Vaheeksami näidisülesandeid

- Arvude arv

# Vaheeksami näidisülesandeid

- Arvude arv

```
def arvude_arv(s):  
    li = s.split(';')  
    mitu = 0  
    for el in li:  
        if el.isnumeric():  
            mitu += 1  
    print(mitu)  
  
print(arvude_arv("aartaa;bbbb;234;45;ccwec"))
```

# Vaheeksami näidisülesandeid

- Arvude arv

```
def arvude_arv(s):  
    li = s.split(';')  
    mitu = 0  
    for el in li:  
        if el.isnumeric():  
            mitu += 1  
    print(mitu)  
  
print(arvude_arv("aartaa;bbbb;234;45;ccwec"))
```

# Vaheeksami näidisülesandeid

- Arvude arv

```
def arvude_arv(s):  
    li = s.split(';')  
    mitu = 0  
    for el in li:  
        if el.isnumeric():  
            mitu += 1  
    print(mitu)  
  
print(arvude_arv("aartaa;bbbb;234;45;ccwec"))
```

0

0

1

2

2

None

# Vaheeksami näidisülesandeid

- Arvude arv

```
def arvude_arv(s):  
    li = s.split(';')  
    mitu = 0  
    for el in li:  
        if el.isnumeric():  
            mitu += 1  
    print(mitu)  
  
print(arvude_arv("aartaa;bbbb;234;45;ccwec"))
```

0

0

1

2

2

None

Ei ole õige!

## Vaheeksami näidisülesandeid

- Arvude arv

## Vaheeksami näidisülesandeid

- Arvude arv

# Vaheeksami näidisülesandeid

- Arvude arv

```
def arvude_arv(s):  
    li = s.split(';')  
    mitu = 0  
    for el in li:  
        if el.isnumeric():  
            mitu += 1  
    print(mitu)  
  
print(arvude_arv("aartaa;bbbb;234;45;ccwec"))
```

# Vaheeksami näidisülesandeid

- Arvude arv

```
def arvude_arv(s):  
    li = s.split(';')  
    mitu = 0  
    for el in li:  
        if el.isnumeric():  
            mitu += 1  
    print(mitu)  
  
print(arvude_arv("aartaa;bbbb;234;45;ccwec"))
```

# Vaheeksami näidisülesandeid

- Arvude arv

```
def arvude_arv(s):  
    li = s.split(';')  
    mitu = 0  
    for el in li:  
        if el.isnumeric():  
            mitu += 1  
    print(mitu)  
  
print(arvude_arv("aartaa;bbbb;234;45;ccwec"))
```

2

None

# Vaheeksami näidisülesandeid

- Arvude arv

```
def arvude_arv(s):  
    li = s.split(';')  
    mitu = 0  
    for el in li:  
        if el.isnumeric():  
            mitu += 1  
    print(mitu)  
  
print(arvude_arv("aartaa;bbbb;234;45;ccwec"))
```

2

None

# Vaheeksami näidisülesandeid

- Arvude arv

```
def arvude_arv(s):  
    li = s.split(';')  
    mitu = 0  
    for el in li:  
        if el.isnumeric():  
            mitu += 1  
    print(mitu)  
  
print(arvude_arv("aartaa;bbbb;234;45;ccwec"))
```

2

None

Ei ole õige!

## Vaheeksami näidisülesandeid

- Arvude arv

## Vaheeksami näidisülesandeid

- Arvude arv

# Vaheeksami näidisülesandeid

- Arvude arv

```
def arvude_arv(s):  
    li = s.split(';')  
    mitu = 0  
    for el in li:  
        if el.isnumeric():  
            mitu += 1  
    return mitu  
  
print(arvude_arv("aartaa;bbbb;234;45;ccwec"))
```

# Vaheeksami näidisülesandeid

- Arvude arv

```
def arvude_arv(s):  
    li = s.split(';')  
    mitu = 0  
    for el in li:  
        if el.isnumeric():  
            mitu += 1  
    return mitu  
  
print(arvude_arv("aartaa;bbbb;234;45;ccwec"))
```

# Vaheeksami näidisülesandeid

- Arvude arv

```
def arvude_arv(s):  
    li = s.split(';')  
    mitu = 0  
    for el in li:  
        if el.isnumeric():  
            mitu += 1  
    return mitu  
  
print(arvude_arv("aartaa;bbbb;234;45;ccwec"))
```

2

## Vaheeksami näidisülesandeid

- Arvude arv

## Vaheeksami näidisülesandeid

- Arvude arv

# Vaheeksami näidisülesandeid

- Arvude arv

```
def arvude_arv(s):  
    li = s.split(';')  
    mitu = 0  
    for el in li:  
        if el.isnumeric():  
            mitu += 1  
    return mitu  
  
print(arvude_arv("aartaa;bbbb;234;45;ccwec"))  
print(arvude_arv("blaa3;bl23aa"))  
print(arvude_arv("3"))  
print(arvude_arv(""))
```

# Vaheeksami näidisülesandeid

- Arvude arv

```
def arvude_arv(s):  
    li = s.split(';')  
    mitu = 0  
    for el in li:  
        if el.isnumeric():  
            mitu += 1  
    return mitu  
  
print(arvude_arv("aartaa;bbbb;234;45;ccwec"))  
print(arvude_arv("blaa3;bl23aa"))  
print(arvude_arv("3"))  
print(arvude_arv(""))
```

# Vaheeksami näidisülesandeid

- Arvude arv

```
def arvude_arv(s):  
    li = s.split(';')  
    mitu = 0  
    for el in li:  
        if el.isnumeric():  
            mitu += 1  
    return mitu  
  
print(arvude_arv("aartaa;bbbb;234;45;ccwec"))  
print(arvude_arv("blaa3;bl23aa"))  
print(arvude_arv("3"))  
print(arvude_arv(""))
```

2

0

1

0

# Vaheeksami näidisülesandeid

- Minuteid keskööst

## Vaheeksami näidisülesandeid

- Minuteid keskööst

# Vaheeksami näidisülesandeid

- Minuteid keskkööst

Programmis peab olema funktsioon `minuteid_keskkööst`, mis võtab argumendiks eespool mainitud formaadis kellaaja, ning tagastab täisarvu, mis näitab, mitu minutit on selleks kellaajaks keskkööst möödunud.

Näited:

```
>>> minuteid_keskkööst("08:45")
525
>>> minuteid_keskkööst("00:00")
0
```

# Vaheeksami näidisülesandeid

- Minuteid keskkööst

Programmis peab olema funktsioon `minuteid_keskkööst`, mis võtab argumendiks eespool mainitud formaadis kellaaja, ning tagastab täisarvu, mis näitab, mitu minutit on selleks kellaajaks keskkööst möödunud.

Näited:

```
>>> minuteid_keskkööst("08:45")
525
>>> minuteid_keskkööst("00:00")
0
```

# Vaheeksami näidisülesandeid

- Minuteid keskkööst

Programmis peab olema funktsioon `minuteid_keskkööst`, mis võtab argumendiks eespool mainitud formaadis kellaaja, ning tagastab täisarvu, mis näitab, mitu minutit on selleks kellaajaks keskkööst möödunud.

Näited:

```
>>> minuteid_keskkööst("08:45")
525
>>> minuteid_keskkööst("00:00")
0
```

```
def minuteid_keskkööst(s):
    l = s.split(':')
    minuteid = int(l[0])*60 + int(l[1])
    return minuteid

print(minuteid_keskkööst("08:45"))
print(minuteid_keskkööst("00:00"))
```

# Vaheeksami näidisülesandeid

- Minuteid keskkööst

Programmis peab olema funktsioon `minuteid_keskkööst`, mis võtab argumendiks eespool mainitud formaadis kellaaja, ning tagastab täisarvu, mis näitab, mitu minutit on selleks kellaajaks keskkööst möödunud.

Näited:

```
>>> minuteid_keskkööst("08:45")
525
>>> minuteid_keskkööst("00:00")
0
```

```
def minuteid_keskkööst(s):
    l = s.split(':')
    minuteid = int(l[0])*60 + int(l[1])
    return minuteid

print(minuteid_keskkööst("08:45"))
print(minuteid_keskkööst("00:00"))
```

# Vaheeksami näidisülesandeid

- Minuteid keskkööst

Programmis peab olema funktsioon `minuteid_keskkööst`, mis võtab argumendiks eespool mainitud formaadis kellaaja, ning tagastab täisarvu, mis näitab, mitu minutit on selleks kellaajaks keskkööst möödunud.

Näited:

```
>>> minuteid_keskkööst("08:45")
525
>>> minuteid_keskkööst("00:00")
0
```

```
def minuteid_keskkööst(s):
    l = s.split(':')
    minuteid = int(l[0])*60 + int(l[1])
    return minuteid
```

```
print(minuteid_keskkööst("08:45"))
print(minuteid_keskkööst("00:00"))
```

525

0

# Vaheeksami näidisülesandeid

- Kinokava

# Vaheeksami näidisülesandeid

- Kinokava

# Vaheeksami näidisülesandeid

- Kinokava

Kui faili *filmid.txt* sisu on näiteks

```
16:15;Raev;135
16:15;Valemaandumine;99
17:00;Maagia kuuvalgel;100
17:30;Kirsitubakas;93
17:50;Kohtunik;142
21:00;Annabelle;94
21:25;Raev;135
```

siis programmi kasutamine võiks näha välja näiteks järgmine

```
Sisesta vaba aja alguskellaaeg: 17:00
Sisesta vaba aja lõpukellaaeg: 20:00
17:00 Maagia kuuvalgel (100 minutit)
17:30 Kirsitubakas (93 minutit)
```

# Vaheeksami näidisülesandeid

- Kinokava

Kui faili *filmid.txt* sisu on näiteks

```
16:15;Raev;135
16:15;Valemaandumine;99
17:00;Maagia kuuvalgel;100
17:30;Kirsitubakas;93
17:50;Kohtunik;142
21:00;Annabelle;94
21:25;Raev;135
```

siis programmi kasutamine võiks näha välja näiteks järgmine

```
Sisesta vaba aja alguskellaaeg: 17:00
Sisesta vaba aja lõpukellaaeg: 20:00
17:00 Maagia kuuvalgel (100 minutit)
17:30 Kirsitubakas (93 minutit)
```

# Vaheeksami näidisülesandeid

- Kinokava

Kui faili *filmid.txt* sisu on näiteks

```
16:15;Raev;135
16:15;Valemaandumine;99
17:00;Maagia kuuvalgel;100
17:30;Kirsitubakas;93
17:50;Kohtunik;142
21:00;Annabelle;94
21:25;Raev;135
```

siis programmi kasutamine võiks näha välja näiteks järgmine

```
Sisesta vaba aja alguskellaaeg: 17:00
Sisesta vaba aja lõpukellaaeg: 20:00
17:00 Maagia kuuvalgel (100 minutit)
17:30 Kirsitubakas (93 minutit)
```

```
algused = []
pealkirjad = []
kestused = []
```

# Vaheeksami näidisülesandeid

- Kinokava

Kui faili *filmid.txt* sisu on näiteks

```
16:15;Raev;135
16:15;Valemaandumine;99
17:00;Maagia kuuvalgel;100
17:30;Kirsitubakas;93
17:50;Kohtunik;142
21:00;Annabelle;94
21:25;Raev;135
```

siis programmi kasutamine võiks näha välja näiteks järgmine

```
Sisesta vaba aja alguskellaaeg: 17:00
Sisesta vaba aja lõpukellaaeg: 20:00
17:00 Maagia kuuvalgel (100 minutit)
17:30 Kirsitubakas (93 minutit)
```

```
algused = []
pealkirjad = []
kestused = []
```

# Vaheeksami näidisülesandeid

- Kinokava

Kui faili *filmid.txt* sisu on näiteks

```
16:15;Raev;135
16:15;Valemaandumine;99
17:00;Maagia kuuvalgel;100
17:30;Kirsitubakas;93
17:50;Kohtunik;142
21:00;Annabelle;94
21:25;Raev;135
```

siis programmi kasutamine võiks näha välja näiteks järgmine

```
Sisesta vaba aja alguskellaaeg: 17:00
Sisesta vaba aja lõpukellaaeg: 20:00
17:00 Maagia kuuvalgel (100 minutit)
17:30 Kirsitubakas (93 minutit)
```

```
algused = []
pealkirjad = []
kestused = []
f = open('filmid.txt')
for rida in f:
    l = rida.split(';')
    algused += [l[0]]
    pealkirjad += [l[1]]
    kestused += [int(l[2])]
f.close()
```

# Vaheeksami näidisülesandeid

- Kinokava

# Vaheeksami näidisülesandeid

- Kinokava

# Vaheeksami näidisülesandeid

- Kinokava

```
print(algused)
print(pealkirjad)
print(kestused)
```

# Vaheeksami näidisülesandeid

- Kinokava

```
print(algused)
print(pealkirjad)
print(kestused)
```

# Vaheeksami näidisülesandeid

- Kinokava

```
print(algused)
print(pealkirjad)
print(kestused)
```

```
['16:15', '16:15', '17:00', '17:30', '17:50', '21:00', '21:25']
['Raev', 'Valemaandumine', 'Maagia kuuvalgel', 'Kirsitubakas', 'Kohtunik', 'Annabelle', 'Raev']
[135, 99, 100, 93, 142, 94, 135]
```

# Vaheeksami näidisülesandeid

- Kinokava

```
print(algused)
print(pealkirjad)
print(kestused)
```

```
['16:15', '16:15', '17:00', '17:30', '17:50', '21:00', '21:25']
['Raev', 'Valemaandumine', 'Maagia kuuvalgel', 'Kirsitubakas', 'Kohtunik', 'Annabelle', 'Raev']
[135, 99, 100, 93, 142, 94, 135]
```

# Vaheeksami näidisülesandeid

- Kinokava

```
print(algused)
print(pealkirjad)
print(kestused)
```

```
['16:15', '16:15', '17:00', '17:30', '17:50', '21:00', '21:25']
['Raev', 'Valemaandumine', 'Maagia kuuvalgel', 'Kirsitubakas', 'Kohtunik', 'Annabelle', 'Raev']
[135, 99, 100, 93, 142, 94, 135]
```

```
algus = input('Sisesta vaba aja alguskellaeg: ')
lopp = input('Sisesta vaba aja lõpukellaeg: ')
algus = minuteid_keskkööst(algus)
lopp = minuteid_keskkööst(lopp)
for i in range(len(algused)):
    filmi_algus = minuteid_keskkööst(algused[i])
    filmi_lopp = filmi_algus + kestused[i]
    if filmi_algus >= algus and filmi_lopp <= lopp:
        print(algused[i] + ' ' + pealkirjad[i] + ' (' + str(kestused[i]) + ' minutit)')
```

# Vaheeksami näidisülesandeid

- Kinokava

```
print(algused)
print(pealkirjad)
print(kestused)
```

```
['16:15', '16:15', '17:00', '17:30', '17:50', '21:00', '21:25']
['Raev', 'Valemaandumine', 'Maagia kuuvalgel', 'Kirsitubakas', 'Kohtunik', 'Annabelle', 'Raev']
[135, 99, 100, 93, 142, 94, 135]
```

```
algus = input('Sisesta vaba aja alguskellaeg: ')
lopp = input('Sisesta vaba aja lõpukellaeg: ')
algus = minuteid_keskkööst(algus)
lopp = minuteid_keskkööst(lopp)
for i in range(len(algused)):
    filmi_algus = minuteid_keskkööst(algused[i])
    filmi_lopp = filmi_algus + kestused[i]
    if filmi_algus >= algus and filmi_lopp <= lopp:
        print(algused[i] + ' ' + pealkirjad[i] + ' (' + str(kestused[i]) + ' minutit)')
```

# Vaheeksami näidisülesandeid

- Kinokava

```
print(algused)
print(pealkirjad)
print(kestused)
```

```
['16:15', '16:15', '17:00', '17:30', '17:50', '21:00', '21:25']
['Raev', 'Valemaandumine', 'Maagia kuuvalgel', 'Kirsitubakas', 'Kohtunik', 'Annabelle', 'Raev']
[135, 99, 100, 93, 142, 94, 135]
```

```
algus = input('Sisesta vaba aja alguskellaeg: ')
lopp = input('Sisesta vaba aja lõpukellaeg: ')
algus = minuteid_keskkööst(algus)
lopp = minuteid_keskkööst(lopp)
for i in range(len(algused)):
    filmi_algus = minuteid_keskkööst(algused[i])
    filmi_lopp = filmi_algus + kestused[i]
    if filmi_algus >= algus and filmi_lopp <= lopp:
        print(algused[i] + ' ' + pealkirjad[i] + ' (' + str(kestused[i]) + ' minutit)')
```

```
Sisesta vaba aja alguskellaeg: 17:00
Sisesta vaba aja lõpukellaeg: 20:00
17:00 Maagia kuuvalgel (100 minutit)
17:30 Kirsitubakas (93 minutit)
```

# Vaheeksami näidisülesandeid

- Kinokava - II

# Vaheeksami näidisülesandeid

- Kinokava - II

# Vaheeksami näidisülesandeid

- Kinokava - II

```
algus = input('Sisesta vaba aja alguskellaaeg: ')
lõpp = input('Sisesta vaba aja lõpukellaaeg: ')
algus = minuteid_keskkööst(algus)
lõpp = minuteid_keskkööst(lõpp)

f = open('filmid.txt')
for rida in f:
    l = rida.split(';')
    f_algus = minuteid_keskkööst(l[0])
    f_lõpp = f_algus + int(l[2])
    if f_algus >= algus and f_lõpp <= lõpp:
        print(l[0] + ' ' + l[1] + ' (' + l[2].strip() + ' minutit)')
f.close()
```

# Vaheeksami näidisülesandeid

- Kinokava - II

```
algus = input('Sisesta vaba aja alguskellaaeg: ')
lõpp = input('Sisesta vaba aja lõpukellaaeg: ')
algus = minuteid_keskkööst(algus)
lõpp = minuteid_keskkööst(lõpp)

f = open('filmid.txt')
for rida in f:
    l = rida.split(';')
    f_algus = minuteid_keskkööst(l[0])
    f_lõpp = f_algus + int(l[2])
    if f_algus >= algus and f_lõpp <= lõpp:
        print(l[0] + ' ' + l[1] + ' (' + l[2].strip() + ' minutit)')
f.close()
```

# Vaheeksami näidisülesandeid

- Kinokava - II

```
algus = input('Sisesta vaba aja alguskellaaeg: ')
lõpp = input('Sisesta vaba aja lõpukellaaeg: ')
algus = minuteid_keskkööst(algus)
lõpp = minuteid_keskkööst(lõpp)
```

```
f = open('filmid.txt')
for rida in f:
    l = rida.split(';')
    f_algus = minuteid_keskkööst(l[0])
    f_lõpp = f_algus + int(l[2])
    if f_algus >= algus and f_lõpp <= lõpp:
        print(l[0] + ' ' + l[1] + ' (' + l[2].strip() + ' minutit)')
f.close()
```

```
Sisesta vaba aja alguskellaaeg: 17:00
Sisesta vaba aja lõpukellaaeg: 20:00
17:00 Maagia kuuvalgel (100 minutit)
17:30 Kirsitubakas (93 minutit)
```

# Vaheeksami näidisülesandeid

- Seotud read

## Vaheeksami näidisülesandeid

- Seotud read

# Vaheeksami näidisülesandeid

- Seotud read

Näide. Kui faili *paarid.txt* sisu on

```
23 12
45 55
987 1
1 34
56 232
232 44
44 87
87 45
12 12
```

siis peaks programm väljastama

```
987 1
1 34
```

# Vaheeksami näidisülesandeid

- Seotud read

Näide. Kui faili *paarid.txt* sisu on

```
23 12
45 55
987 1
1 34
56 232
232 44
44 87
87 45
12 12
```

siis peaks programm väljastama

```
987 1
1 34
```

# Vaheeksami näidisülesandeid

- Seotud read

Näide. Kui faili *paarid.txt* sisu on

```
23 12
45 55
987 1
1 34
56 232
232 44
44 87
87 45
12 12
```

siis peaks programm väljastama

```
987 1
1 34
```

```
f = open('paarid.txt')
read = f.readlines()
f.close()
print(read)
```

# Vaheeksami näidisülesandeid

- Seotud read

Näide. Kui faili *paarid.txt* sisu on

```
23 12
45 55
987 1
1 34
56 232
232 44
44 87
87 45
12 12
```

siis peaks programm väljastama

```
987 1
1 34
```

```
f = open('paarid.txt')
read = f.readlines()
f.close()
print(read)
```

# Vaheeksami näidisülesandeid

- Seotud read

Näide. Kui faili *paarid.txt* sisu on

```
23 12
45 55
987 1
1 34
56 232
232 44
44 87
87 45
12 12
```

siis peaks programm väljastama

```
987 1
1 34
```

```
f = open('paarid.txt')
read = f.readlines()
f.close()
print(read)
```

```
['23 12\n', '45 55\n', '987 1\n', '1 34\n', '56 232\n', '232 44\n', '44 87\n', '87 45\n', '12 12']
```

# Vaheeksami näidisülesandeid

- Seotud read

## Vaheeksami näidisülesandeid

- Seotud read

## Vaheeksami näidisülesandeid

- Seotud read

```
f = open('paarid.txt')
```

```
read = f.readlines()
```

```
f.close()
```

```
for i in range(len(read) - 1):
```

```
    esimene = read[i].split()
```

```
    teine = read[i + 1].split()
```

```
    if esimene[1].strip() == teine[0]:
```

```
        print(esimene[0], esimene[1])
```

```
        print(teine[0], teine[1].strip())
```

```
        break
```

## Vaheeksami näidisülesandeid

- Seotud read

```
f = open('paarid.txt')
```

```
read = f.readlines()
```

```
f.close()
```

```
for i in range(len(read) - 1):
```

```
    esimene = read[i].split()
```

```
    teine = read[i + 1].split()
```

```
    if esimene[1].strip() == teine[0]:
```

```
        print(esimene[0], esimene[1])
```

```
        print(teine[0], teine[1].strip())
```

```
        break
```

## Vaheeksami näidisülesandeid

- Seotud read

```
f = open('paarid.txt')
read = f.readlines()
f.close()
```

```
for i in range(len(read) - 1):
    esimene = read[i].split()
    teine = read[i + 1].split()
    if esimene[1].strip() == teine[0]:
        print(esimene[0], esimene[1])
        print(teine[0], teine[1].strip())
        break
```

```
987 1
1 34
```

Suur tänu osalemast

ja

kohtumiseni!