

# 1. Matlabi ja Octave põhioperatsioonid, funktsioonid ja graafika

GNU Octave on matemaatikapaketi Matlab vabavaraline kloon. Selle saab alla laadida aadressilt

<http://octave.sourceforge.net>

Kui kasutate operatsioonisüsteemi Windows, tuleb sellel lk valida link Windows installer (Octave and Octave-Forge). Installatsioonifaili suurus on ca 70mb.

Octave või Matlabi käivitamiseks tuleb lihtsalt klõpsata vastavale ikoonile. Octave käivitamisel avaneb aken, mis sisaldab nn käsurida. Viimase lõpus on märk `>` ja selle järel kursor. Matlabi käivitamisel avaneb (olenevalt konfiguratsioonist) mitu akent, kusjuures üks neist on tööaken, mis sisaldab eelpoolkirjelatud käsurida.

Käsureale kursori kohale on võimalik sisestada mitmesuguseid käsked. Vaatleme alguses mõningaid elementaarseid operatsioone.

*Arvu esitus.* Käsitleme esialgu reaalarve (hiljem vaatleme ka kompleksarve). Komakoha tähistamiseks kasutatakse punkti. Näiteks sisestame käsureale arvu 1.5 ja vajutame enterile. Matlab-Octave kuvab seejärel

```
ans=1.5000
```

See tähendab, et Matlab-Octave sisestas arvu 1.5 ja salvestas selle muutuja `ans` all. Muutuja `ans` on Matlab-Octaves reserveeritud viimase operatsiooni tulemuse salvestamiseks. Sisestame nüüd `-0.36` ja vajutame enterile. Kuvatakse

```
ans=-0.36000
```

See tähendab, et `ans` sai uue väärtuse: `-0.36`. Punkti ette ei ole vaja nulli kirjutada, nt viimase arvu võib sisestada ka kujul `-.36`

Arvu võib esitada ka eksponenti kasutades. Sellel on järgmine kuju:

*arv e kümne astendaja*

Näiteks sisestades `2.4e3` ja vajutades enter, saab muutuja `ans` väärtuse  $2.4 \times 10^3 = 2400$ , st kuvatakse

```
ans=2400
```

Sisestades `-45e-3` saab `ans` väärtuse `-0.045` jne.

*Muutujad ja neile väärtuste omistamine.* Muutujast `ans` oli juba juttu. Üldiselt võib Matlab-Octaves muutujaks olla suvalise pikkusega sümbolite jada, mis koosneb numbritest, tähtedest ja märkidest `_`. Seejuures esimene sümbol ei tohi olla number. Näiteks võivad muutujateks olla `x`, `z.1`, `R12`, `alpha` jne.

Muutujale mingi väärtuse omistamine toimub võrdusmärgi kasutades. Selle operatsiooni kuju on järgmine:

*Muutuja=väärtus*

Näiteks sisestades käsureale `t=5` ja vajutades enter, omistab Matlab-Octave muutujale `t` väärtuse 5 ja kuvab järgmisele reale

```
t=5
```

Sisestades `c1=8e-3` ja vajutades enter, omistatakse muutujale `c1` väärtus 0.008 ja kuvatakse

```
c1=0.008
```

Omistamisoperatsioon ei ole sümmeetriline, nt  $t=5$  asemel ei saa sisestada  $5=t$ . Viimasest käsust ei saaks Matlab-Octave aru.

Muutujale pi on algselt omistatud arvu  $\pi$  ligikaudne väärtus. Kui sisestada pi ja vajutada enter, kuvatakse

```
pi=3.1416
```

Algselt väärtusi omavaid muutujaid on teisigi (nt i on imaginaarühik). Samas saab taolisi muutujaid ümber salvestada. Näiteks sisestades pi=10 omandab muutuja pi uue väärtuse 10, sisestades i=1 saab muutuja i uue väärtuse 1 jne.

*Avaldised.* Aritmeetiliste tehete jaoks kasutatakse järgmisi märke:

```
liitmine +  
lahutamine -  
korrutamine *  
jagamine /  
astendamise ^
```

Astendamise märgi saab eesti klaviatuurilt kas AltGr Ä või AltGr Shift Ä vajutades. Tehete grupeerimiseks saab kasutada ümarsulge ( ).

Tehete järjekord on järgmine: kõigepealt astendamine, siis korrutamine-jagamine ja lõpuks liitmine-lahutamine. Grupisiseseid astendamisi, korrutamisi-jagamisi ja liitmisi-lahutamisi sooritatakse vasakult paremale. Näiteks tehte  $2 \times \frac{3}{4} \times 5$  saab sisestada käsureale järgmiselt:  $2*3/4*5$ . Peale enterile vajutamist kuvatakse vastus

```
ans=7.5
```

Tehte  $\frac{2 \times 3}{4 \times 5}$  saab sisestada järgmiselt:  $2*3/(4*5)$ . Vastus on

```
ans=0.3
```

Tehte  $2^{3^2}$  saab sisestada järgmiselt:  $2^(3^2)$ . Vastus on

```
ans=512
```

**NÄITEÜLESANNE 1.** Sisestada muutujate väärtused  $x = 3$ ,  $y = -1$  ja arvutada  $z = xy - 3(x + y)^2$ .

Lahendus. Sisestame  $x=3$  vajutame enter, sisestame  $y=-1$  ja vajutame enter. Lõpuks sisestame  $z=x*y-3*(x+y)^2$  ja vajutame enter. Kuvatakse vastus

```
z=-15
```

**HARJUTUSÜLESANNE 1.** Kasutades muutujate  $x$  ja  $y$  väärtusi näiteülesandest 1 arvutada  $w_1 = 4(x - y)^3 - \frac{6}{(x+y)^2}$ ,  $u_1 = \frac{x^3 - y + \frac{y}{x-8}}{y + \frac{1}{x} - 2x^4 y^5}$  ja  $v = (x w_1 - \frac{x}{y})^{\frac{1}{u_1 - y}}$ .

**Lahendus**

*Octavest väljumiseks* tuleb käsurealt sisestada `exit` või `quit` Matlabist väljumiseks tuleb ülevalt menüüst valida vastav käsk.

*Skriptid.* Kui arvutused sisaldavad palju käske, on mõistlik need salvestada eraldi faili ja sealt käivitada. Taolisi faile nimetatakse skriptideks (ingl. script), käibekeeles ka programmideks.

Kui on plaanis skripte kasutada, tuleb Matlabis ja Octaves fikseerida kataloog, milles need asuvad (seda nim töökataloogiks). Matlabis saab töökataloogi valida käsuakna kohal olevast menüüst. Kui Octave on vaikeparameetrite järgi installeeritud, on tema töökataloog käivitamisel automaatselt järgmine:

```
c:\Octave\versiooni nr_gcc 4.4.0\bin
```

Viimane sisaldab lisaks mitmeid installeeritud Octave programme. Seetõttu on mõistlik luua eraldi töökataloog skriptide jaoks. Töökataloogi aadressi saab Octaves muuta käsuga `cd` järgmiselt:

```
cd kataloogi aadress
```

Näiteks kui luua `c` kettale skriptide jaoks kataloog nimega `scripts`, siis tuleb

Octaves käsurealt sisestada `cd c:\scripts` ja vajutada enter.

Matlab-Octave skriptide laiendiks on `.m` Skripti käivitamiseks tuleb käsurealt sisestada skripti nimi ilma laiendita. Näiteks kui tahetakse käivitada skript nimega `skr.m` tuleb sisestada `skr` ja vajutada enter.

Skripte saab luua ja toimetada suvalise tekstiredaktoriga. Matlabi on tekstiredaktor sisse ehitatud. Octave installatsiooniga on kaasas Notepad++, mida võib edukalt skriptide jaoks kasutada. Seega tuleb skriptidega töötamisel kasutada kahte akent: ühes on tekstiredaktor, milles asub avatud skript ja teine on Matlabi käsuaken või Octave. Iga kord, kui skripti muudetakse, tuleb see enne käivitamist salvestada.

**NÄITEÜLESANNE 2.** Luua ja käivitada skript `pr1.m`, mis sisestab  $x_1 = 5$ ,  $x_2 = -6$  ja arvutab  $z = x_1^2 - x_2^2 + \frac{x_1}{x_1 - x_2}$ .

Lahendus. Avame tekstiredaktoris uue faili, sisestame sinna järgmised read:

```
x1=5
x2=-6
z=x1^2-x2^2+x1/(x1-x2)
```

ja salvestame selle faili nimega `pr1.m` Matlab-Octave töökataloogi. Matlab-Octave käsureale kirjutame `pr1` ja vajutame enter. Kuvatakse järgmised read:

```
x1=5
x2=-6
z=-16
```

Kui me ei soovi, et Matlab-Octave kuvaks mingite tehete tulemusi, siis tuleb skriptis vastava rea lõppu panna semikoolon; Näiteks muudame eelmist skripti järgmiselt:

```
x1=5;
x2=-6;
z=x1^2-x2^2+x1/(x1-x2)
```

ja salvestame uuesti. Peale käivitamist kuvatakse ainult viimase operatsiooni tulemus:

```
z=-16
```

Skripti saab lisada ka kommentaare. Selleks tuleb kommentaarirea ette kirjutada `%`. Näiteks võib eeltoodud skripti täiendada järgmiselt:

```
%Sisestame x1 ja x2
x1=5;
x2=-6;
%Arvutame z
z=x1^2-x2^2+x1/(x1-x2)
```

**HARJUTUSÜLESANNE 2.** [Skeemil 1](#) toodud vooluahela parameetrid on järgmised:  $R_1 = 10\Omega$ ,  $R_2 = 3\Omega$ ,  $R_3 = 15\Omega$ ,  $R_4 = 10\Omega$ ,  $R_5 = 6\Omega$ ,  $E = 50V$ . Arvutada ahela kogutakistus ja vool. Skript salvestada `s1.m` nime all.

[Lahendus](#)

**HARJUTUSÜLESANNE 3.** [Skeemil 2](#) toodud vooluahela parameetrid on järgmised:  $R_1 = 12\Omega$ ,  $R_2 = 10\Omega$ ,  $R_3 = 15\Omega$ ,  $R_4 = 6\Omega$ ,  $E = 12V$ . Arvutada ahela kogutakistus ja vool. Skript salvestada `s2.m` nime all.

[Lahendus](#)

*Funktsioonid.* Matlab-Octaves on suur hulk sisseehitatud funktsioone. Neid võib leida Matlabi Helpist või Octavega kaasasolevast dokumentatsioonist. Mõned näited:

<i>eksponentfunktsioon</i> ( $e^x$ )	<code>exp</code>
<i>siinus</i>	<code>sin</code>
<i>kosiinus</i>	<code>cos</code>

<i>tangens</i>	tan
<i>kotangens</i>	cot
<i>naturaallogaritm</i>	log
<i>kümnendlogaritm</i>	log10
<i>arkussinus</i>	asin
<i>arkuskosinus</i>	acos
<i>arkustangens</i>	atan
<i>arkuskotangens</i>	acot
<i>ruutjuur</i>	sqrt

Funktsiooni kasutamiseks tuleb sisestada tema nimi ja selle järel argument ümarsulgudes. Näiteks selleks, et arvutada  $u = \arctan(z)$ , kus  $z = 50$ , tuleb sisestada käsurealt või kirjutada skripti järgmised read:

```
z=50
u=atan(z)
```

NÄITEÜLESANNE 3. Arvutada avaldise  $z = 2 \cos(5 + 3t) + 3 \tan(1 - 2t)(1 - t^5)$  väärtus kohal  $t = 3$ .

Lahendus. Kirjutame skripti järgmised read:

```
t=3;
z=2*cos(5+3*t)+3*tan(1-2*t)*(1-t^5)
```

salvestame skripti ja käivitame Matlabist või Octavest. Vastus on  $z=-2454$

HARJUTUSÜLESANNE 4. Arvutada  $v = \tan(3t + u) - \frac{\arcsin(\frac{t}{50})}{\sqrt{1+u^2}}$  kui  $t = 43$  ja  $u = -1.7$ . Vastav skript salvestada s3.m nime all.

**Skript** Vastus:  $v=-15.771$

Matlab-Octaves saab ka ise funktsioone defineerida. Selleks on mitmeid võimalusi. Vaatleme siinkohal lihtsaimat võimalust, mille korral funktsioon defineeritakse samas skriptis, kus teostatakse ülejäänud arvutusi (üldiselt saab funktsioonide jaoks luua ka eraldi skripte). Taolise funktsiooni defineerimise käsu üldkuju on järgmine:

*funkts. sümbol=@(argumendi sümbol,argumendi sümbol,...)funkts. valem*

NÄITEÜLESANNE 4. Defineerida funktsioon  $f(x) = x^3 - x^2 + 1$  ja arvutada  $f(5)$ .

Lahendus. Kirjutame skripti järgmised read:

```
f=@(x)x^3-x^2+1;
f(5)
```

salvestame skripti ja käivitame Matlabist või Octavest. Antakse vastus  $\text{ans}=101$ .

Saab defineerida ka mitme muutuva funktsioone. Siis, vastavalt eeltoodud definitsiooni üldkujule, tuleb @ järel sulgudes argumendid eraldada komadega.

NÄITEÜLESANNE 5. Defineerida funktsioon  $\Psi(u, v) = \sqrt{u} + \cos(uv) + 5$  ja arvutada  $d = \Psi(3, 4)$ .

Lahendus. Kirjutame skripti järgmised read:

```
Psi=@(u,v)sqrt(u)+cos(u*v)+5;
d=Psi(3,4)
```

salvestame ja käivitame. Vastus:  $d=7.5759$

HARJUTUSÜLESANNE 5. Defineerida funktsioon  $z(x_1, x_2, y) = ye^{x_1^2 - x_2^2}$  ja arvutada  $v_1 = z(0, 5, 6)$  ning  $v_2 = z(1, 1, 1)$ . Vastav skript salvestada s4.m nime all.

**Skript** Vastused:  $v_1=8.3328e-11$   $v_2=1$

HARJUTUSÜLESANNE 6. Mittelineaarse takisti volt-amperkarakteristik on  $u(i) = 6i + 4 \arctan(\frac{i}{3})$ . Defineerida funktsioonid  $u(i)$  ja  $r(i)$  ning arvutada  $r_1 = r(1)$ ,  $r_2 = r(2)$ ,  $u_1 = u(1)$ ,  $u_2 = u(2)$ . Vastav skript salvestada s5.m nime all.

**Skript** Vastused:  $r_1=7.287$   $r_2=7.176$   $u_1=7.287$   $u_2=14.352$

*Graafikud.* Matlabis ja Octaves on rohkelt võimalusi mitmesuguste graafikute ja diagrammide joonestamiseks. Vaatleme siinkohal kahemõõtmeliste graafikute joonestamist käskudega `fplot` ja `plot`.

Kõige lihtsam moodus ühe muutuja funktsiooni graafiku saamiseks on kasutada käsku `fplot`, mille kuju on järgmine:

```
fplot('funkts. valem',[argumendi väikseim väärtus,argumendi suurim väärtus])
```

Näiteks funktsiooni  $\sin t$  graafiku saamiseks lõigul  $0 \leq t \leq 10$  tuleks sisestada käsk

```
fplot('sin(t)',[0,10])
```

Graafiku paremasse ülemisse nurka tekib automaatselt nn legend. Legendi teksti saab muuta lisades käsu

```
legend('legendi tekst')
```

Graafikule saab lisada võrgu, so vertikaalsed ja horisontaalsed punktiirjooned, mis jooksevad läbi telgede sõlmpunktide. Selleks on käsk

```
grid('on')
```

Nii Matlab kui ka Octave avavad graafiku eraldi aknas. Matlabis on võimalik graafikut eksportida samast aknast. Selleks tuleb menüüst valida File, Save as ning sobiv faili formaat, milles soovitakse eksportida. Octaves saab graafikut eksportida käsuga `print`, mille kuju on järgmine:

```
print -dformaad faili nimi koos laiendiga
```

Näiteks joonise `figure1` eksportimisel `jpg`, `jpeg`, `ps` või `pdf` formaadis tuleb lisada vastavalt üks järgmistest käskudest:

```
print -djpg figure1.jpg
```

```
print -djpeg figure1.jpeg
```

```
print -dps figure1.ps
```

või

```
print -dpdf figure1.pdf
```

**NÄITEÜLESANNE 6.** Joonestada funktsiooni  $f(x) = x^3 - x$  graafik lõigul  $\frac{1}{2} \leq x \leq \frac{3}{2}$  koos võrguga. Lisada sobiv legend ja salvestada joonis nime `joonis1.pdf` all.

Lahendus. Kirjutame skripti järgmised read:

```
fplot('x^3-sqrt(x)',[.5,1.5]);
```

```
legend('x^3-x');
```

```
grid('on');
```

Octave kasutamisel lisame ka käsu

```
print -dpdf joonis1.pdf
```

Seejärel salvestame ja käivitame. Graafik on [siin](#)

Ühel joonisel on võimalik kujutada ka mitu graafikut. Selleks tuleb käsus `fplot` funktsioonide valemid eraldada komadega ja asetada nurksulgudesse. Käsus legend tuleb samuti erinevate graafikute kohta käivad tekstid komadega eraldada.

**NÄITEÜLESANNE 7.** Joonestada funktsioonide  $\sin t$ ,  $\cos t$  ja  $\sin 4t$  graafikud lõigul  $[-2, 7]$ . Lisada sobiv legend ja salvestada joonis nime `joonis2.jpg` all.

Lahendus. Kirjutame skripti järgmised read:

```
fplot('[sin(t),cos(t),sin(4*t)]',[-2,7]);
```

```
legend('sin t','cos t','sin 4t');
```

Octave kasutamisel lisame ka käsu

```
print -djpg joonis2.jpg
```

Seejärel salvestame ja käivitame. Graafik on [siin](#)

**HARJUTUSÜLESANNE 7.** Joonestada funktsioonide  $y_1 = 60e^{-\frac{1}{x}}$  ja  $y_2 = 7x^2 + 6x$

graafikud lõigul [1, 3] võrguga. Lisada sobiv legend ja salvestada joonis nime joonis3.gif all. Vastav skript salvestada s6.m nime all.

### Skript Joonis

Graafikutele on võimalik lisada telgede märgendeid käskudega

```
xlabel('märgendi tekst')
ylabel('märgendi tekst')
```

Lisaks on võimalik muuta graafikul oleva teksti suurust, stiili ja asukohta, joonte stiili, suurust ja värvi jm. Kõige selle kohta saab täpsemat infot Matlabi Helpist või Octave dokumentatsioonist.

Käsku fplot on suhteliselt lihtne kasutada, kuid selle võimalused on üsna piiratud. Üks oluline kitsendus on see, et käsus esinevad funktsioonid tohivad sisaldada vaid ühte muutujat. Näiteks käsk

```
fplot('sin(314*t)',[0,0.1])
```

on korrektne, kuid

```
fplot('sin(w*t)',[0,0.1])
```

enam mitte (isegi juhul, kui muutujale w on eelnevalt väärtus antud).

Rohkemate võimalustega on käsk plot, mille kuju on järgmine:

```
plot(xvektor,yvektor)
```

Suurused *xvektor* ja *yvektor* on võrdse pikkusega reaalarvudest koosnevad vektorid:

$$xvektor = (x_1, x_2, \dots, x_n)$$
$$yvektor = (y_1, y_2, \dots, y_n).$$

Käsk plot toimib järgmiselt: tasandile joonestatakse punktid  $P(x_i, y_i)$ ,  $i = 1, \dots, n$  ja need ühendatakse järjest sirglõikudega. Seega tekib sirgelõikudest koosnev pidev joon. Selleks, et graafik oleks sile (ilma nähtavate murdekohadeta), peavad punktid  $P(x_i, y_i)$  paiknema piisavalt tihedalt. See tähendab, et vektorite *xvektor* ja *yvektor* komponendid peavad olema valitud piisavalt lühikese sammuga.

Ühtlase sammuga vektori loomiseks saab kasutada järgmist käsku:

```
vektori tähis = esimene element : samm : viimane element
```

Näiteks vektori  $x = 1, 3, 5, 7, 9, 11, 13$  saab luua järgmise käsuga:

```
x=1:2:13
```

Vektori  $t_1 = 0, 0.01, 0.02, \dots, 0.5$  saab luua käsuga

```
t1=0:0.01:0.5
```

või

```
t1=0:1e-2:0.5
```

Funktsiooni väärtuste vektori loomiseks piisab funktsiooni rakendamisest argumenti vektorile.

NÄITEÜLESANNE 8. Joonestada funktsiooni  $y = 5\sqrt{x} + 17$  graafik lõigul [0, 3], lisada telgede märgendid ja salvestada nime joonis4.pdf all.

Lahendus. Koostame järgmise skripti

```
%Arvutame x vektori piisavalt lühikese sammuga (olgu selleks 0.001)
```

```
x=0:1e-3:3;
```

```
%Arvutame y vektori
```

```
y=5*sqrt(x)+17;
```

```
%Joonestame graafiku
```

```
plot(x,y)
```

```
xlabel('x')
```

```
ylabel('y')
```

Octave korral lisame ka käsu

```
print -dpdf joonis4.pdf
```

Salvestame skripti ja käivitame. Graafik on [siin](#)

Käsuga `plot` automaatselt legendi ei teki. Legendi saab soovi korral lisada käsuga `legend` nii nagu `fplot` korral. Soovitav on skriptis argumenti ja funktsiooni vektorite loomise käskude järgi panna ; Vastasel juhul kuvab Matlab-Octave need pikad vektorid ekraanile.

HARJUTUSÜLESANNE 8. Sisestada  $\omega = 100$  ja joonestada funktsiooni  $z = \sin(\omega t)$  graafik lõigul  $[1, 1.1]$ . Lisada võrk, telgede märgendid ja sobiv legend. Salvestada joonis nime `joonis5.pdf` all. Vastav skript salvestada nime `s7.m` all.

[Skript Joonis](#)

Käsuga `plot` saab joonestada ka mitu graafikut ühte teljestikku. Selleks tuleb käsku kirjutada mitu vektorite *xvektor* ja *yvektor* paari komadega eraldatult.

NÄITEÜLESANNE 9. Joonestada funktsioonide  $z = \tan x$  ja  $z = e^{2-x}$  graafikud lõigul  $[2, 3]$ , lisada telgede märgendid, võrk, legend ja salvestada nime `joonis6.gif` all.

Lahendus. Koostame järgmise skripti

```
%Arvutame x vektori
x=2:1e-4:3;
%Arvutame z vektorid
z1=tan(x);
z2=exp(2-x);
%Joonestame graafikud
plot(x,z1,x,z2)
xlabel('x')
ylabel('x')
grid('on')
legend('tan x','exp(2-x)')
```

Octave korral lisame ka käsu

```
print -dgif joonis6.gif
```

Salvestame skripti ja käivitame. Graafik on [siin](#)

HARJUTUSÜLESANNE 9. Kolmefaasilise EMJ allika sagedus on 50Hz ja amplituud 325V. Joonestada kõigi kolme faasi graafikud ühes teljestikus intervallil  $t \in [0, 0.05s]$ . Lisada sobiv legend ja telgede märgendid. Salvestada joonis nime `joonis7.pdf` all. Vastav skript salvestada nime `s8.m` all.

[Lahendus](#)