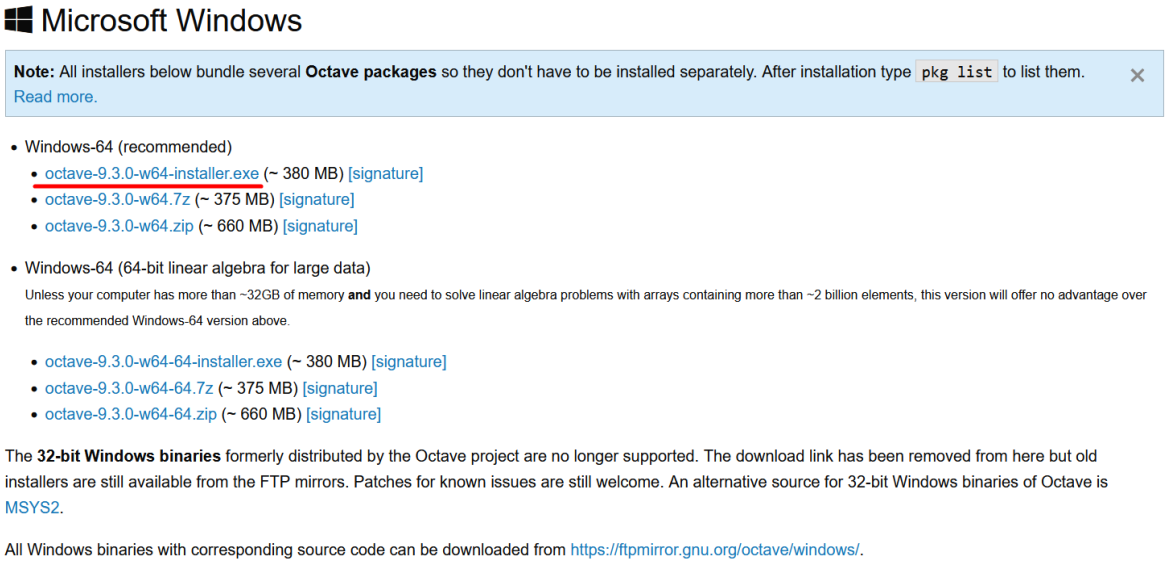
Octave instaliavimas, funkcijos, generatoriaus radimas

Turinys

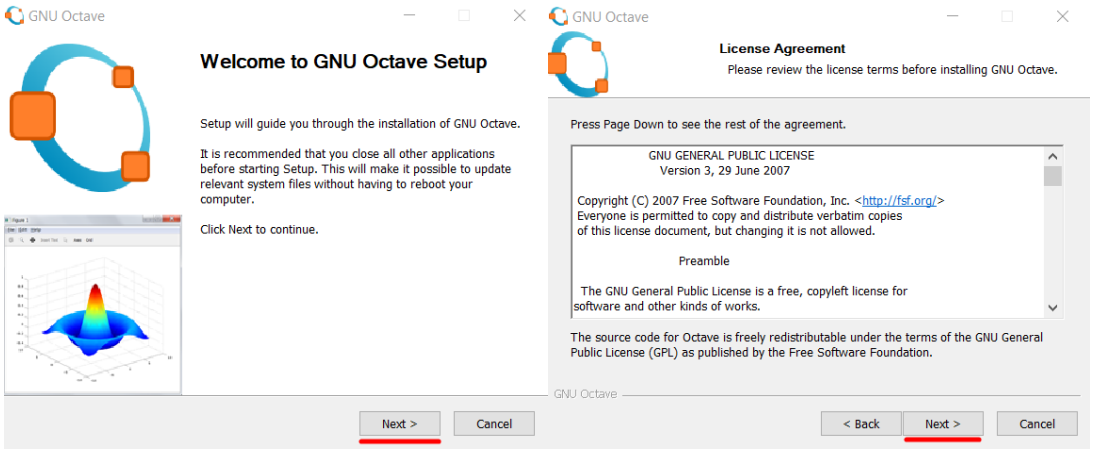
1. Octave įrankio instaliavimas

Atsisiųskite naujausią Octave įrankio instaliavimo failo versiją (gali būti ir naujesnė) iš GNU Octave oficialios svetainės (žr. 1 pav).

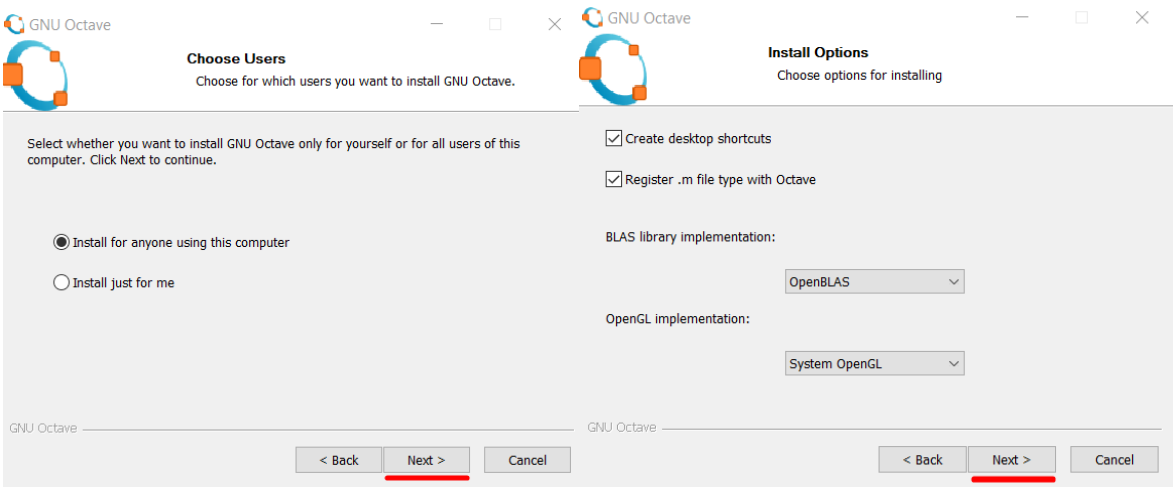


1 pav. Octave įrankio instaliavimo failo atsisiuntimas

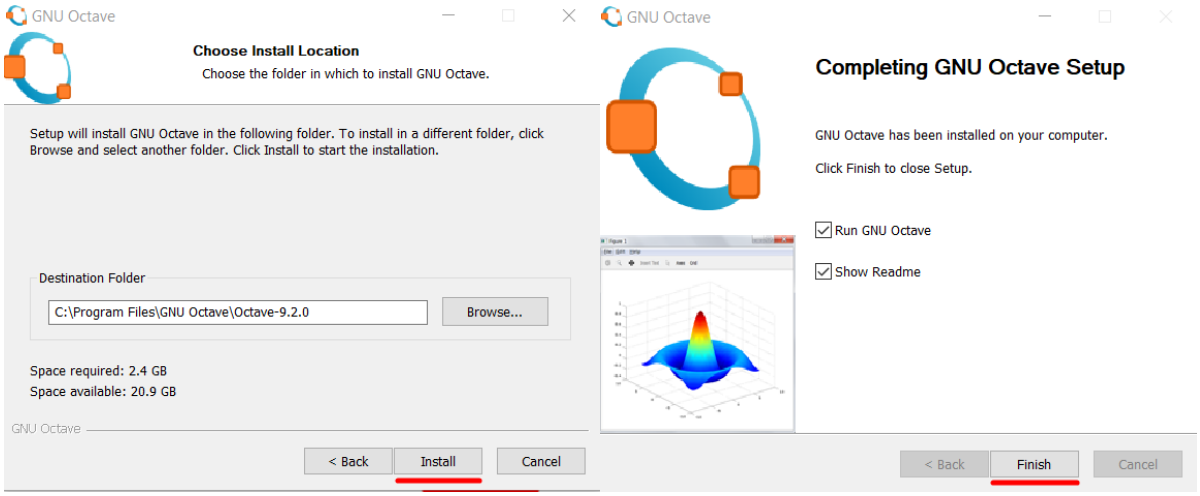
Instaliuokite įrankį naudodami atsisiųstą instaliavimo failą ir sekite toliau esančius instaliavimo žingsnius pateikiamus 2-4 pav. (instaliavimo pasirinkimai pažymėti raudona spalva).



2 pav. Octave įrankio instaliavimas, sutikimai

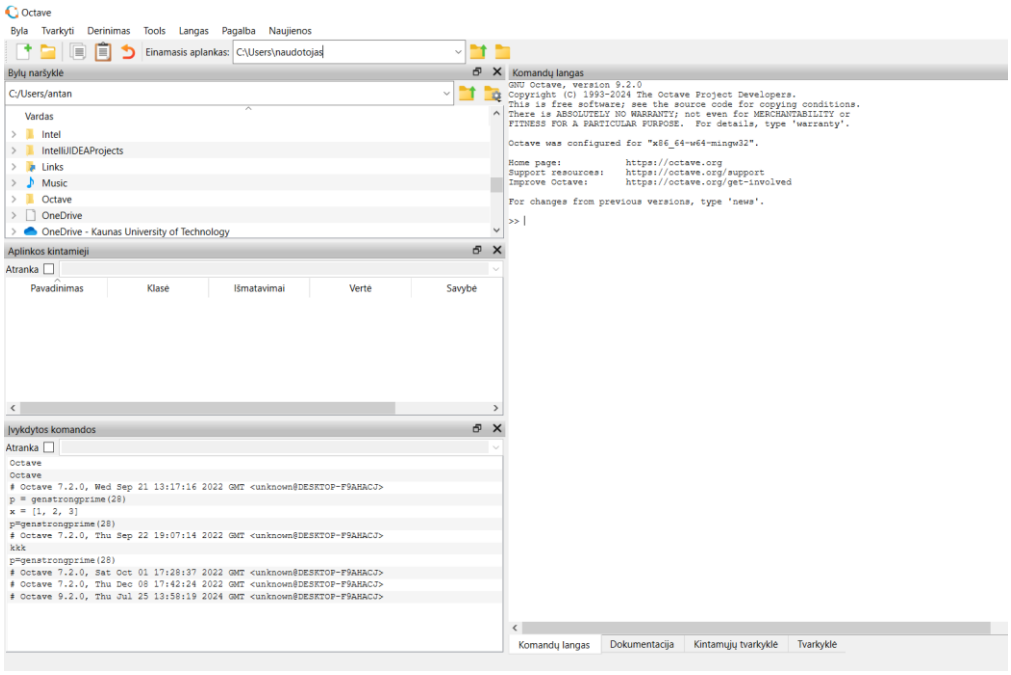


3 pav. Octave įrankio instaliavimas, papildomos parinktys



4 pav. Octave įrankio instaliavimas ir instaliavimo užbaigimas

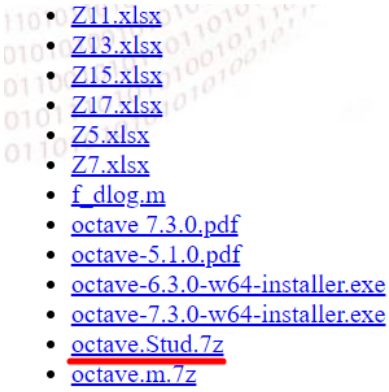
Galiausiai po instaliavimo automatiškai arba atvėrus įrankį matomas Octave pagrindinis langas pateikiamas 5 pav.



5 pav. Pagrindinis Octave įrankio langas

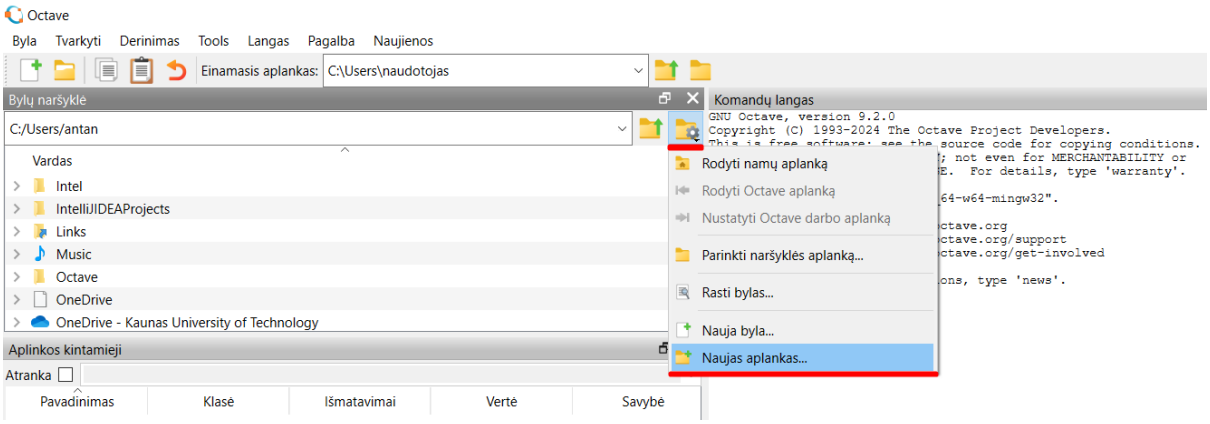
2. Octave aplinkos parengimas darbui

Iš svetainės *KriptoGama* atsisiųskite (žr. 5 pav.) darbui reikalingus Octave funkcijų failus patalpintus zip archyve *octave.Stud.7z*.

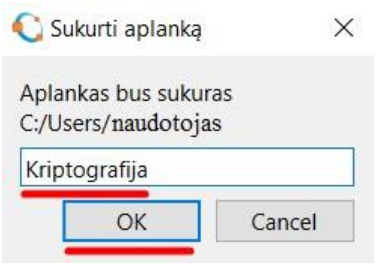


5 pav. Octave įrankio funkcijų failų atsisiuntimas

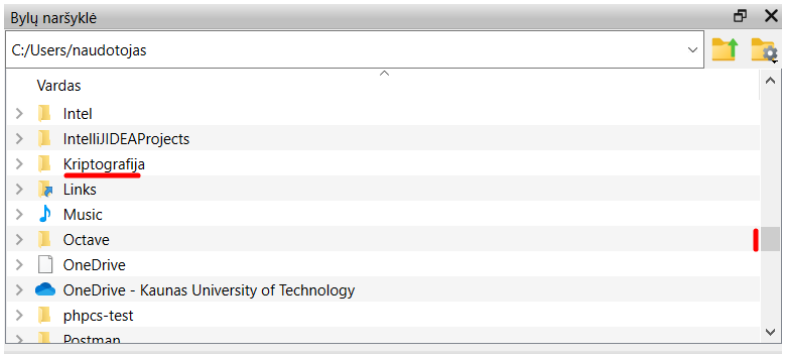
Octave įrankyje skiltyje *Bylų naršyklė* iškleiskite nustatymų parinktį ir pasirinkite parinktį *Naujas aplankas* (žr. 6 pav.). Atsivėrusiame modaliniame lange įrašykite aplankui norimą pavadinimą (pvz. pavadinimas *Kriptografija*) ir paspauskite sukūrimą patvirtinantį mygtuką *Ok* (žr. 7 pav.). Sukurtą aplanką galima rasti (žr. 8 pav.) vykdant paiešką *Bylų naršyklė*.



6 pav. Naujo aplanko kūrimo parinkties pasirinkimas

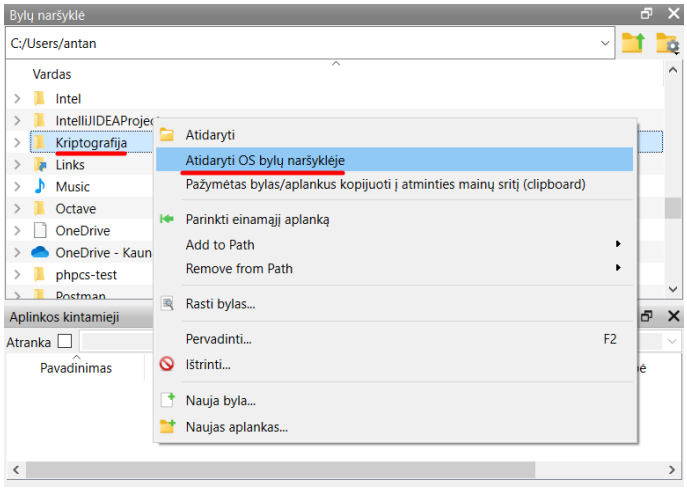


7 pav. Pavadinimo naujam aplankui suteikimas ir sukūrimo patvirtinimas



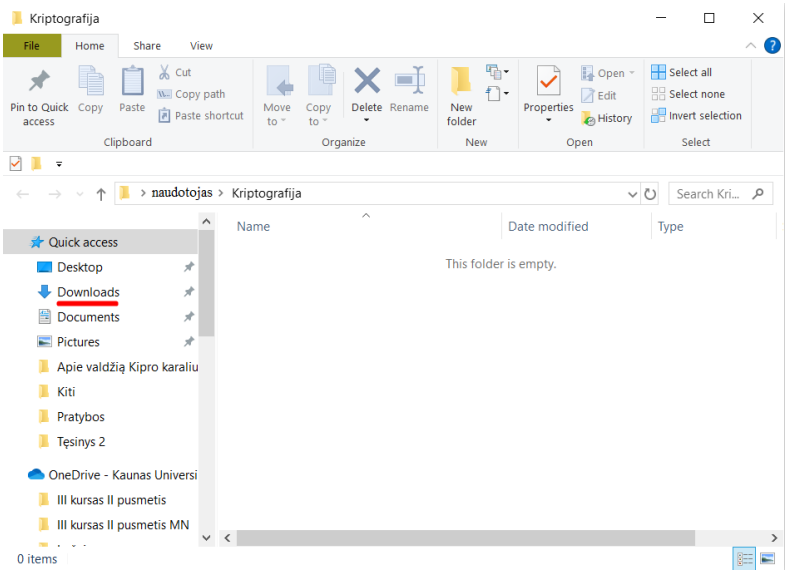
8 pav. Sukurto aplanko paieška *Bylų naršyklė*

Atidarykite sukurtą aplanką operacinės sistemos naršyklėje (žr. 9 pav.), paspausdami dešiniu pelės klavišu ant sukurto aplanko ir pasirinkdami parinktį *Atidaryti* OS bylų naršyklėje.

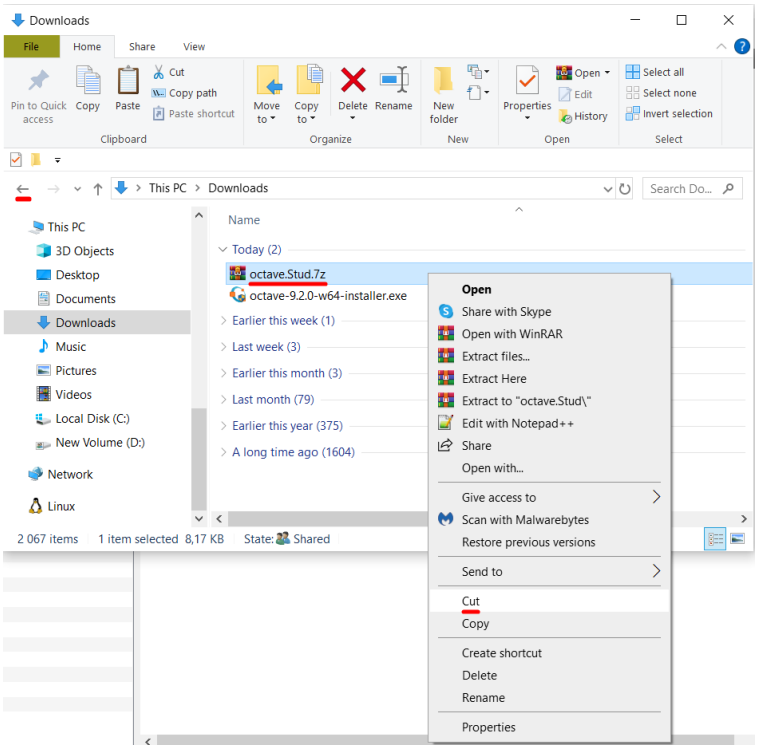


9 pav. Sukurto aplanko atvėrimas operacinės sistemos naršyklėje

Paspauskite mygtuką *Atsisiuntimai/Downloads* ir atidarytame aplanke suraskite Octave funkcijų archyvą *octave.Stud.7z* ir paspaudę dešiniu mygtuku ant jo pasirinkite parinktį *Cut* ir kairėje pusėje esančios rodyklės pagalba sugrįžkite į prieš tai buvusį aplanką *Kriptografija* (žr. 10-11 pav.).

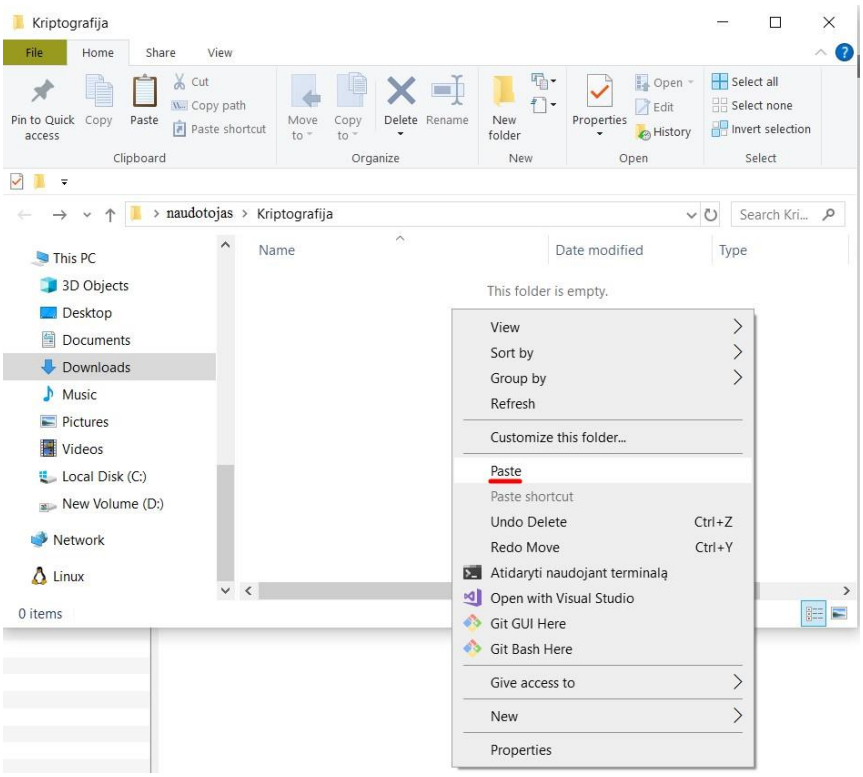


10 pav. Atsisiuntimų aplanko atvėrimas

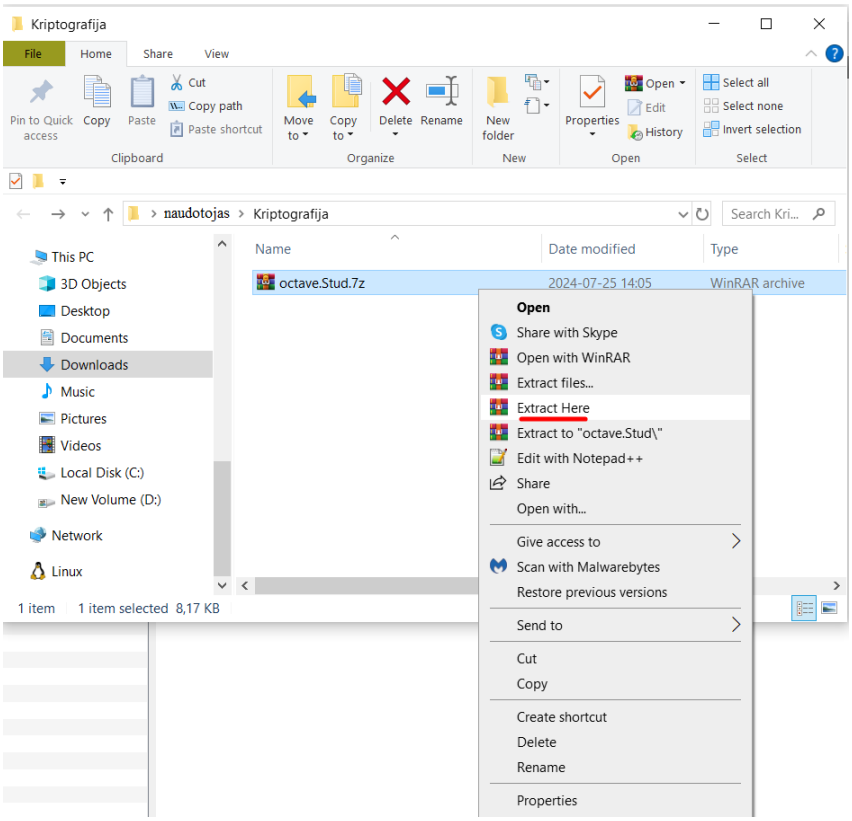


11 pav. Octave funkcijų archyvo iškirpimas ir grįžimas į prieš tai buvusį aplanką

Aplanke *Kriptografija* paspauskite dešinį mygtuką ant tuščios erdvės ir pasirinkite parinktį *Paste* (žr. 12 pav.) ir įklijavus archyvą ant jo paspauskite dešiniu klavišu ir pasirinkite parinktį *Extract Here* pateikiamą 13 pav.

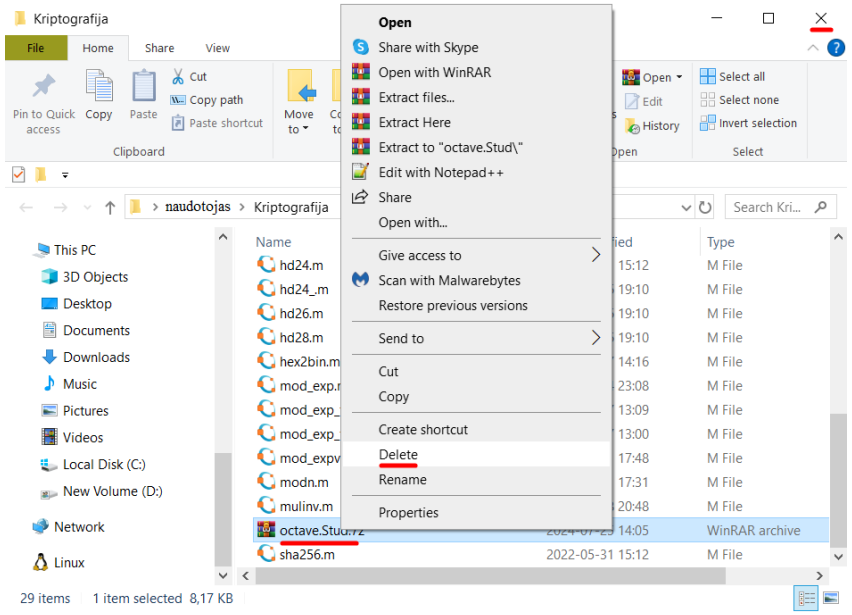


12 pav. Octave funkcijų archyvo įklijavimas



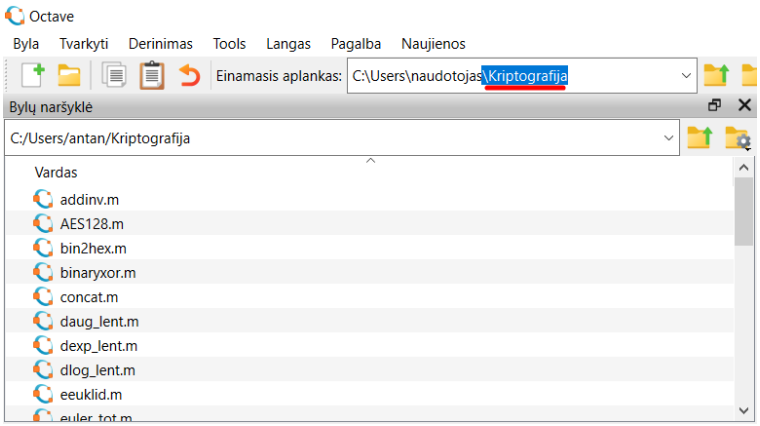
13 pav. Octave funkcijų archyvo iškleidimas

Išarchyvavus funkcijų failus, galima (žr. 14 pav.) ištrinti funkcijų archyvą *octave.Stud.7z* paspaudžiant ant jo dešiniu klavišu ir pasirenkant parinktį *Delete*. Taip pat, užverkite OS naršyklės langą paspaudus dešiniajame kampe esantį užvėrimo mygtuką.



14 pav. Octave funkcijų archyvo pašalinimas ir OS naršyklės lango užvėrimas

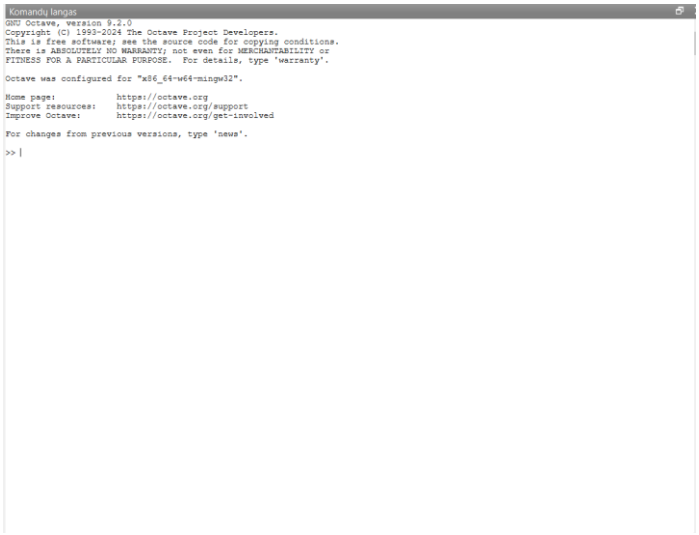
Siekiant užkrauti funkcijas į Octave įrankį, kiekvieno paleidimo metu, būtina papildomai nurodyti į jūsų sukurtą aplanką (pvz. *\Kriptografija*) ir paspausti klaviatūroje *Enter* mygtuką. Jeigu visi žingsniai buvo tinkamai atlikti laikantis aukščiau pateiktų nurodymų, funkcijų failai yra pateikiami *Bylų naršyklė* skiltyje ir darbui su Octave yra pasiruošta.



15 pav. Funkcijų failų įkėlimas į Octave įrankį

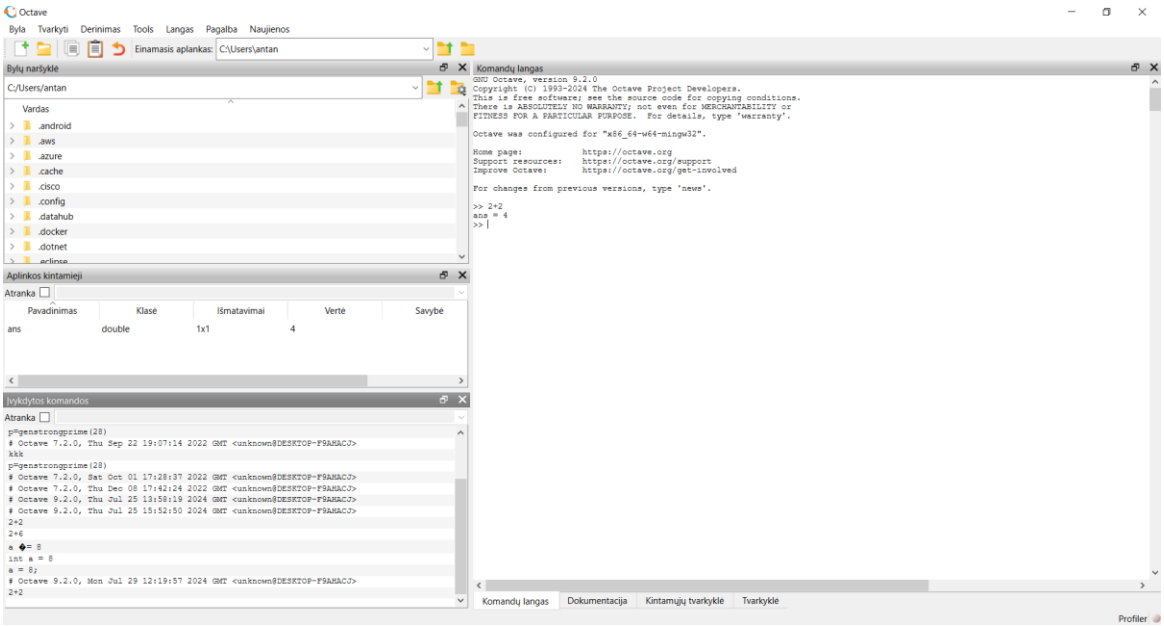
3. Kurse naudojamos Octave funkcijos

Kurse toliau naudosime Octave langą *Komandų langas* (žr. 16 pav.) komandoms įvesti (funkcijos, kintamieji ir kt.) ir rezultatams peržiūrėti.



16 pav. Octave komandų įvedimo langas

Taip pat, kintamuosius galima peržiūrėti *Aplinkos kintamieji* lange, o įvykdytas komandas *Įvykdytos komandos* lange. Komandos 2+2 vykdymo ir prieš tai aptarti langai bendrai pateikiami 17 pav.



17 pav. Octave aplinkos kintamųjų peržiūros langas

3.1 Octave funkcijų paaiškinimai

Kurse naudojami Octave kintamieji ir funkcijos pateikiamos 1 ir 2 lentelėse, o 3 lentelė konvertavimas į dvejetainę, dešimtainę, šešioliktainę skaičių sistemas.

1 lentelė. Kurse naudojami Octave kintamieji

Nr.	Kintamasis	Pavyzdys
1	Sveikasis skaičius.	10
2	Eilutės tipo kintamasis.	"abcd"

2 lentelė. Kurse naudojamos Octave funkcijos

Nr.	Funkcija	Paaiškinimas	**Pavyzdys** (vedant į *Komandų langas)*
1	**int64**	Konvertuoti x vertę į 64 bitų sveikųjų skaičių tipą, plačiau.	>> x=int64(1451668545) x = 1451668545
2	**randi**	Atsitiktinių skaičių generavimas, plačiau.	>> x=randi(2^28-1) x = 2.3294e+08
3	**genprime**	Pirminio skaičiaus generavimas nurodant bitų skaičių.	>> p=genprime(28) p = 194865859
4	**genstrongprime**	Stipraus pirminio skaičiaus generavimas nurodant bitų skaičių.	p=genstrongprime(28) p = 201318479
5	**isprime**	Patikrinama ar tam tikras skaičius yra pirminis.	>> isprime(p) ans = 1
6	**gcd**	Didžiausio bendro daliklio radimas, plačiau.	>>gcd(15, 9) >> gcd(7,9) ans=3 ans = 1
7	**mod**	Modulio skaičiavimas g **mod** p, plačiau.	>> mod(23,19) ans = 4
8	**daug_lent**	Daugybos lentelės g **mod** p generavimas kvadratinės matricos pavidalu.	>> daug_lent(11) ans = 1 2 3 4 5 6 7 8 9 10 2 4 6 8 10 1 3 5 7 9 3 6 9 1 4 7 10 2 5 8 …
9	**mulinv**	Modulinės inversijos skaičiavimas g^{-1} **mod** p.	>> ml=mulinv(5, 6) ml = 5
10	**mod_exp**	Modulinės eksponentės skaičiavimas g^x **mod** p.	>> mod_exp(5,13, 211504967) ans = 163178290 >> mod_exp(2,2,11) ans = 4
11	**dexp_lent**	Daugybos lentelės g^x **mod** p generavimas kvadratinės matricos pavidalu.	>> dexp_lent(11) ans = 1 2 3 4 5 6 7 8 9 10 2 4 8 5 10 9 7 3 6 1 3 9 5 4 1 3 9 5 4 1 …
12	**factor**	Skaičiaus faktorizavimas pirminiais, plačiau.	>> f=factor(15) f = 3 5
13	**dec2bin**	Dešimtainio skaičiaus konvertavimas į dvejetainį, atitinkantį neneigiamąjį sveikąjį skaičių, kaip vienetų ir nulių eilutę, plačiau.	>> xb=dec2bin(14) xb = 1110
14	**dec2hex**	Dešimtainio skaičiaus konvertavimas į šešioliktainį, atitinkantį neneigiamąjį sveikąjį skaičių, plačiau.	>> xh=dec2hex(14) xh = E >> šešioliktainioilgis=16; >> kh=dec2hex(14, šešioliktainioilgis)

			kh =000000000000000E
15	**bin2dec**	Dvejetainio skaičiaus konvertavimas į dešimtainį, plačiau.	>> x=bin2dec("1110") x = 14
16	**bin2hex**	Dvejetainio skaičiaus konvertavimas į šešioliktainį.	>> xh=bin2hex("1110") xh = E
17	**binaryxor**	Xor radimas dviems dvejetainiams skaičiams.	>> bxr=binaryxor("1110", "1100") bxr = 10
18	**h24,** **h26** **h28,** **sha256** **(originali)**	Eilutės reikšmei santraukos sha256 apskaičiavimas grąžinant 6, 7, 7, 64 paskutinius šešioliktainius skaitmenis.	>> h24('Labas Broniau!') h = 3D5FC1 >> h26('Labas Broniau!') h = 03D5FC1 >> h28('Labas Broniau!') h = 03D5FC1 >> sha256("Labas Broniau!") h = 49E7BE56B8996FDEA455493A86B44DB7F4BB7965C6B1E192C8B9C941503D5FC1
19	**hd24,** **hd26,** **hd28**	Kaip ir nr. 18 esančioms funkcijoms atliekami analogiški skaičiavimai, tik grąžinant dešimtainę vertę.	>> h=int64(hd24("Labas Broniau!")) h = 4022209 >> h=hd26("Labas Broniau!") h = 4022209 >> h=hd28("Labas Broniau!") h = 4022209
20	**concat**	Kelių elementų sujungimas į vieną eilutę.	>> c=concat(1234, "abcd") c = 1234abcd
21	**AES128**	Šifruojamas 1 atvirojo teksto 128 bitų ilgio blokas, atitinkantis 16 ASCII simbolių. Parametrai: m(in) – užšifruojama eilutės tipo žinutė; Kh – slaptas simetrinis raktas šešioliktainiu pavidalu; NR – AES128 naudojamų raundų skaičius; fun –"e" eilutės tipo kintamasis, kuriuo iškviečiama šifravimo funkcija, o įvedus "d" iškviečiama dešifravimo funkcija	>> m="Labas Broniau!"; >> Kh=dec2hex(14,32); >> NR=1; >> fun="e" **Šifravimas:** >> C=AES128(m,Kh,NR,fun) C = 018c0376c823368c4e8c82d49b0024f2 **Iššifravimas:** >> fun="d" >> ms=AES128(C,Kh,NR,fun) ms= Labas Broniau!

3 lentelė. Konvertavimas į dvejetainę, dešimtainę, šešioliktainę skaičių sistemas

Dvejetainis	Dešimtainis	Šešioliktainis
0000	0	0
0001	1	1
0010	2	2
0011	3	3
0100	4	4
0101	5	5
0110	6	6
0111	7	7
1000	8	8
1001	9	9
1010	10	A
1011	11	B
1100	12	C
1101	13	D
1110	14	E
1111	15	F

Pavyzdžiui.
Dvejetainio skaičiaus perskaičiavimas į dešimtainį:
$1000 = 1 \cdot 2^3 + 0 \cdot 2^2 + 0 \cdot 2^1 + 0 \cdot 2^0 = 8.$
Dvejetainio skaičiaus konvertavimas į dešimtainį:
6 4 6 5 10 11 12
0110 0100 0110 0101 1010 1011 1100 = 6465101112.
Dvejetainio skaičiaus konvertavimas į šešioliktainį:
F E F 7 F A D
1111 1110 1111 0111 1111 1010 1101 = FEF7FAD.

3.2 Keleto Octave funkcijų platesni taikymo pavyzdžiai

Kurse naudojami viešieji parametrai ***p*** (pagrinde 28 bitų ilgio) ir generatorius ***g***.
Ciklinė grupė: $\mathbf{Z}_p^*$ = {1, 2, 3, …, ***p***-1}; **•mod *p***, **:mod *p***.

Daugybos modulio, modulinės inversijos, diskretinės eksponentės funkcijos rezultatai **mod 11** pateikiami 4-6 lentelėse.

4 lentelė. Daugybos lentelė *** mod 11**

Daugybos Z_{11}^* lentelė $g \bullet v$ mod 11										
Z_{11}^* (g/v)	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
1	1	2	3	4	5	6	7	8	9	10
2	2	4	6	8	10	1	3	5	7	9
3	3	6	9	1	4	7	10	2	5	8
4	4	8	1	5	9	2	6	10	3	7
5	5	10	4	9	3	8	2	7	1	6
6	6	1	7	2	8	3	9	4	10	5
7	7	3	10	6	2	9	5	1	8	4
8	8	5	2	10	7	4	1	9	6	3
9	9	7	5	3	1	10	8	6	4	2
10	10	9	8	7	6	5	4	3	2	1

Octave ***g*** **mod 11** pavyzdžiai:

```
>> mod(11,11)
ans = 0
>> mod(-9,11)
ans = 2
>> mod(5+9,8)
ans = 6
>> mod(5+2,11)
ans = 7
>> mod(5-9,7)
ans = 3
>> mod(9^2,11)
ans = 4
```

5 lentelė. Modulinė inversija **mod 11**

Modulinė inversija Z_{11}^*
1^{-1}= 1 mod 11
2^{-1}= 6 mod 11
3^{-1}= 4 mod 11
4^{-1}= 3 mod 11
5^{-1}= 9 mod 11
6^{-1}= 2 mod 11
7^{-1}= 8 mod 11
8^{-1}= 7 mod 11
9^{-1}= 5 mod 11
10^{-1}= 10 mod 11

Sugeneruotam atsitiktiniam skaičiui modulinės inversijos radimas ir pasitikrinimas:

```
>> t=int64(randi(2^28-1))
t = 58435490
>> t_m1=mulinv(t,11)
t_m1 = 194971802
>> mod(t*t_m1,11)
ans = 1 ← jeigu 1 inversija rasta teisingai
```

Pagrindinė funkcija naudojama kriptografijoje yra Diskretinės Eksponentės funkcija – DEF: $DEF_g(x) = g^x \bmod p = a$.

6 lentelė. Diskretinės eksponentės daugybos lentelė **mod 11**

	Diskretinės eksponentės Z_{11}^ lentelė g^x mod 11										
^(g/x)	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
1	1	1	2	3	4	5	6	7	8	9	1
2	1	2	4	8	5	10	9	7	3	6	1
3	1	3	9	5	4	1	3	9	5	4	1
4	1	4	5	9	3	1	4	5	9	3	1
5	1	5	3	4	9	1	5	3	4	9	1
6	1	6	3	7	9	10	5	8	4	2	1
7	1	7	5	2	3	10	4	6	9	8	1
8	1	8	9	6	4	10	3	2	5	7	1
9	1	9	4	3	5	1	9	4	3	5	1
10	1	10	1	10	1	10	1	10	1	10	1

Octave g^x **mod 11** pavyzdžiai:

```
>> mod_exp(11,11,11)
ans = 0
>> mod_exp(-9,5,11)
ans = 10
>> mod_exp(5+9,6, 11)
ans = 3
>> mod_exp(5+2,7,8)
ans = 7
>> mod_exp(5-9,3,11)
ans = 2
>> mod_exp(9^2,8,6)
ans = 3
```

Generatoriaus radimas

Kriptografijoje generatorius g yra matematinės grupės elementas, kuris, taikant grupės operaciją (pvz., kėlimą laipsniu), gali sugeneruoti kiekvieną kitą grupės elementą. Šis elementas yra esminis kriptografiniuose protokoluose, tokiuose kaip Difio Helmanas (angl. *Diffie Helman*), kur padeda saugiai sukurti bendrus slaptus raktus. Šis principas taip pat naudojamas kituose algoritmuose, pavyzdžiui, ElGamalio ir DSA (angl. *Digital Signature Algorithm*), kuriuose svarbu saugiai perduoti informaciją ar autentifikuoti pranešimus.

Apskritai sudėtinga užduotis rasti generatorius aibėje $\mathbf{Z}_p^* = \{1, 2, 3, \ldots, p\text{-}1\}$, tačiau naudojant stiprų pirminį p ir *Lagranžo teoremą grupės teorijoje*, generatorių $\mathbf{Z}_p^*$ galima rasti atsitiktine tvarka. Paieška laikoma užbaiga jei tenkinamos dvi sąlygos:

1. jeigu p ir q yra **stiprūs pirminiai** $p = 2 \cdot q + 1 \rightarrow q = (p\text{-}1)/2$;
2. jeigu visi $g \in \Gamma$, $g^q \neq 1$ **mod** p ir $g^2 \neq 1$ **mod** p. **Tik 40% skaičių yra generatoriai.**

Pavyzdinis generatoriaus radimas (***g*** didinamas po vienetą, kol *ans* $g^q \neq 1$ **mod** ***p*** ir $g^2 \neq 1$ **mod** ***p***):

```
>> p=genstrongprime(28)
p = 187086587
>> isprime(p)
ans = 1
>> q=(p-1)/2
q = 93543293
>> isprime(q)
ans = 1
>> g=2
>> mod_exp(g,q,p)
ans = 187086586
>> g=3;
>> mod_exp(g,q,p)
ans = 1
>> g=4;
>> mod_exp(g,q,p)
ans = 1
>> mod_exp(g,2,p)
ans = 4
```

```
>> p=genstrongprime(28)
p = 144668519
>> q=(p-1)/2
q = 72334259
>> g=2;
>> mod_exp(g,q,p)
ans = 1
>> g=7;
>> mod_exp(g,q,p)
ans = 144668518
>> mod_exp(g,2, p)
ans = 49
```

```
>> p=genstrongprime(28)
p = 211504967
>> q=(p-1)/2
q = 105752483
>> g=5
g = 5
>> mod_exp(g,q,p)
ans = 211504966
>> mod_exp(g,2, p)
ans = 25
```

Užduotys generatoriaus radimui.

1. Turėdami pirminį skaičių ***p***, pradėkite nuo ***g***=2 ir didinkite ***g*** po vienetą, kol rasite pirmąjį generatorių kartu nustatydami, kuris iš toliau pateiktų generatorių ***g*** buvo surastas duotam pirminiam skaičiui:
   1. ***p***=int64(264221733);
   2. ***p***=int64(197566703);
   3. ***p***=int64(144668519);
   4. ***p***=int64(224013599).

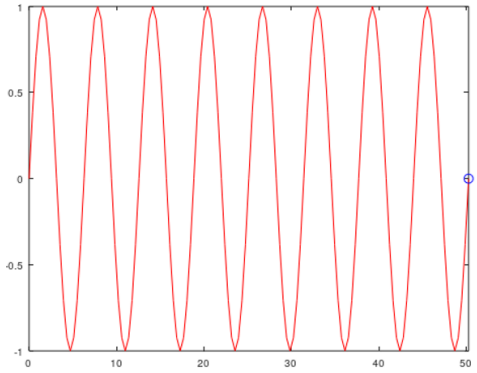
   Generatoriai ***g*** pranešimui ***m***:
   1. ***g*** = 2;
   2. ***g*** = 5;
   3. ***g*** = 111;
   4. ***g*** = 7.
2. Turėdami pirminį skaičių ***p*** ir numanomą generatorių ***g***, nustatykite, ar pateiktas ***g*** iš tikrųjų yra generatorius aibėje $\mathbf{Z}_p^*$ (turi būti tenkinamos generatoriaus paieškos radimo sąlygos):
   1. ***p***=int64(201718619), ***g***=1;
   2. ***p***=int64(214682879), ***g***=214682878;
   3. ***p***=int64(151248827), ***g***=2;
   4. ***p***=int64(265736063), ***g***=5.

Peržiūros komandos

Toliau pateikiamos komandos skirtos tik peržiūrai (plačiau gilintis nebūtina).

Sinuso (žr. 18 pav.) grafiko atvaizdavimas:

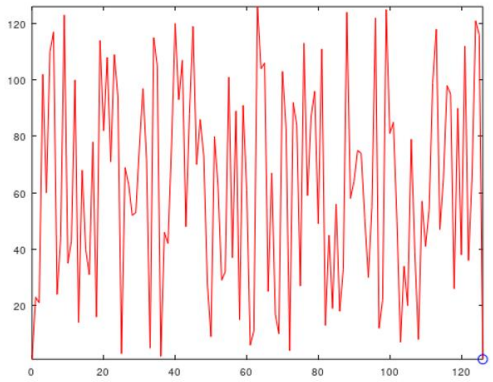
```
>> p128sin
xrange = 16 * pi;
step = xrange/128;
x = 0:step:xrange;
y = sin(x);
comet(x, y)
```



18 pav. Sinuso grafikas

Diskretinės eksponentės (žr. 19 pav.) grafiko atvaizdavimas:

```
>> p128def
p = 127;
g = 23;
x = 0:p-1;
a = mod_expv(g, x, p);
comet(x, a)
```



19 pav. Diskretinės eksponentės grafikas