

# შესავალი GNU/Linux სისტემებში

## სავარჯიშოები

არჩილ ელიზბარაშვილი  
2023

1. ვთქვათ შელში იმყოფებით `/home/user/Documents` დირექტორიაში. გადადით `/usr/share/common-licenses` დირექტორიაში. გამოიყენეთ როგორც აბსოლუტური გზა, ასევე ფარდობითი.
2. ვთქვათ იმყოფებით `/usr/local/bin` დირექტორიაში. გადადით `/usr/share/common-licenses` დირექტორიაში. გამოიყენეთ როგორც აბსოლუტური გზა, ასევე ფარდობითი.
3. ვთქვათ იმყოფებით რომელიმე დირექტორიაში. გადადით თქვენს პირად დირექტორიაში როგორც აბსოლუტური, ასევე ფარდობითი გზით.
4. შექმენით ფაილები შემდეგი დასახელებით: `„name“`, `„name surname“`, `„name\surname“`, `„'name 'surname'“`, `„'\".\"",` `„'/\"\"`. ოპერაცია შეასრულეთ შელში ერთი ბრძანებით!
5. HTML დასახელების დირექტორიაში შექმენით 1000 ცალი html გაფართოების ფაილი (გაფართოება უნდა იყოს დიდი, პატარა და შერეული ასოებით მოცემული). ოპერაცია შეასრულეთ შელში ერთი ბრძანებით!
6. შექმენით ფაილები `„name1“`, `„name2“` და `„name3“` დასახელებით როგორც `„txt“` გაფართოებით, ასევე მის გარეშე. ოპერაცია შეასრულეთ შელში ერთი ბრძანებით!
7. შექმენით ფაილები `„name1“`, `„name2“` და `„name3“` დასახელებით როგორც `„txt“` გაფართოებით (გაფართოება მიუთითეთ დიდი და პატარა ასო-ნიშნების ყველა შესაძლო კომბინაციით: `txt`, `tXt`, `tXT...`) ასევე მის გარეშე. ოპერაცია შეასრულეთ შელში ერთი ბრძანებით!
8. შექმენით ფაილები `„name100“`, `„name101“` ... `„name2023“` დასახელებით როგორც `„txt“` გაფართოებით (გაფართოება მიუთითეთ დიდი და პატარა ასო-ნიშნების ყველა შესაძლო კომბინაციით: `txt`, `tXt`, `tXT...`) ასევე მის გარეშე. ოპერაცია შეასრულეთ შელში ერთი ბრძანებით!
9. შექმენით სამ ასო-ნიშნიანი დასახელების დირექტორიები, რომლის პირველი ასო-ნიშანი იქნება A-დან Z-ის ჩათვლით, მეორე a-დან z-ის ჩათვლით, ხოლო მესამე ციფრი. ოპერაცია შეასრულეთ შელში ერთი ბრძანებით!
10. `test` დირექტორიაში შექმენით სამ ასო-ნიშნიანი ყველა შესაძლო დასახელების მქონე ფაილი `“`, `“`, `“`, `“` და `“` ასო-ნიშნებით.
11. შექმენით ყველა შესაძლო ორ ასო-ნიშნიანი დასახელების დირექტორიები, რომელთა დასახელებებიც თვლის თექვსმეტობითი სისტემის სიმბოლოების

ყველა კომბინაციას წარმოადგენს. ოპერაცია შეასრულეთ შედეგში ერთი ბრძანებით!

12. შექმენით სამ ასო-ნიშნის ყველა შესაძლო დასახელების დირექტორია ციფრებით. მათში სათითაოდ შექმენით ყველა შესაძლო ერთი, ორი, სამი და ოთხ სიმბოლიანი დასახელების ფაილები A, B და C ასო-ნიშნებით. ოპერაცია შეასრულეთ შედეგში ერთი ბრძანებით!

შემდეგ კი ეკრანზე გამოიტანეთ შემდეგი ფაილები:

ა)  $\overline{01234\vee56789}/\vee\overline{CB}^1$

ბ) ყველა 2 ასო-ნიშნის დასახელების ფაილი, რომელიც მდებარეობს იმ დირექტორიებში, რომლის დასახელებაშიც არ შედის სიმბოლო „5“

გ) ყველა 1 და 3 ასო-ნიშნის დასახელების ფაილი, სადაც არ შევა სიმბოლო „B“ და რომლებიც მდებარეობს დირექტორიებში, რომლის დასახელებაშიც შედის მხოლოდ სიმბოლოები „1“, „3“ და „9“

13. შექმენით სამ ასო-ნიშნის ყველა შესაძლო დასახელების დირექტორია " და ' ასო-ნიშნებით.
14. დაუშვათ გაქვთ ბევრი jpg, bmp და png გაფართოების ფაილი "Photos/" დირექტორიაში. ამოარჩიეთ მათგან ის jpg და png ფაილები, რომელთა დასახელებაშიც ფიგურირებს ერთი ციფრი მაინც და რომლის პირველი ასო-ნიშნის D-დან P-მდეა. ოპერაცია შეასრულეთ შედეგში ერთი ბრძანებით!
15. დაუშვათ გაქვთ ბევრი jpg, bmp და png გაფართოების ფაილი "Photos/" დირექტორიაში. ამოარჩიეთ მათგან ის jpg და png ფაილები (გაფართოება შესაძლებელია დიდი, პატარა ან შერეული ასოებით იყოს მოცემული!), რომელთა დასახელებაშიც ფიგურირებს ერთი ციფრი მაინც და რომლის პირველი ასო-ნიშნის D-დან P-მდეა. ოპერაცია შეასრულეთ შედეგში ერთი ბრძანებით!
16. დაუშვათ გაქვთ ბევრი flv, vob და ogg გაფართოების ვიდეო ფაილი "Videos/" დირექტორიაში. ამოარჩიეთ მათგან ის flv და ogg ფაილები (გაფართოება შესაძლებელია დიდი, პატარა ან შერეული ასოებით იყოს მოცემული!), რომელთა დასახელებაშიც ფიგურირებს ერთი ციფრი მაინც და რომლის ბოლო ასო-ნიშნის K-დან Q-მდეა. ოპერაცია შეასრულეთ შედეგში ერთი ბრძანებით!
17. დაუშვათ გაქვთ ბევრი pdf და txt გაფართოების ფაილი "Documents/" დირექტორიაში. ამოარჩიეთ მათგან ის pdf ფაილები, რომლის დასახელებაც 7 ასო-ნიშნისაა (გაფართოების გარეშე) და ის txt ფაილები, რომელთა დასახელებაშიც ფიგურირებს 2 ციფრი მაინც და რომლის პირველი ასო-ნიშნის C-დან R-მდეა. ოპერაცია შეასრულეთ შედეგში ერთი ბრძანებით!
18. მე-15, მე-16 და მე-17 დავალებების პირობის გათვალისწინებით გადააკოპირეთ TEST დირექტორიაში ხუთასონიშნის ყველა სურათის, ვიდეოს და დოკუმენტის ფაილი, რომელიც ციფრზე მთავრდება. ოპერაცია შეასრულეთ შედეგში ერთი ბრძანებით!

---

<sup>1</sup> X ნიშნავს X-ის უარყოფას, ხოლო V ნებისმიერი სიმბოლოა

19. მე-15 და მე-16 დავალებების პირობის გათვალისწინებით გადააკოპირეთ TEST დირექტორიაში ხუთასონიშნიანი სურათის ფაილებისა და შვიდასონიშნიანი ვიდეოს ფაილების გარდა სხვა სურათებისა და ვიდეოფაილები. ოპერაცია შეასრულეთ შელში ერთი ბრძანებით!
20. გამოიტანეთ ეკრანზე ორსიმბოლოიანი ფაილების (იგულისხმება ყველა ტიპის ფაილი) გარდა ყველა დანარჩენი. ოპერაცია შეასრულეთ შელში ერთი ბრძანებით!
21. მე-15, მე-16 და მე-17 დავალებების პირობის გათვალისწინებით გადაიტანეთ TEST დირექტორიაში სურათის ყველა ის ფაილები, რომელთა დასახელებაშიც არ შედის „a“ ასო-ნიშანი, ვიდეოს ყველა ის ფაილები, რომელთა დასახელებაშიც არ შედის „b“ ასო-ნიშანი და დოკუმენტის ყველა ის ფაილები, რომელთა დასახელებაშიც არ შედის „c“ ასო-ნიშანი. ოპერაცია შეასრულეთ შელში ერთი ბრძანებით!
22. მე-15, მე-16 და მე-17 დავალებების პირობის გათვალისწინებით გადაიტანეთ TEST დირექტორიაში სურათის ყველა ის ფაილები, რომელთა დასახელებაშიც არ შედის ლათინური პატარა ასო-ნიშნები (გაფართოების გარდა), ვიდეოს ყველა ის ფაილები, რომელთა დასახელებაშიც არ შედის ციფრები და დოკუმენტის ყველა ის ფაილები, რომელთა დასახელებაშიც არ შედის ლათინური დიდი ასო-ნიშნები (გაფართოების გარდა). ოპერაცია შეასრულეთ შელში ერთი ბრძანებით!
23. აღწერეთ ხესებრი სტრუქტურით შელის შემდეგი ბრძანება:
- ```
ls /usr/{local/{bin,lib},lib/{gimp/?.?*,emacs/[0-9]*.[1-9]}}
```
24. აღწერეთ ხესებრი სტრუქტურით შელის შემდეგი ბრძანება: `ls /usr/{share/{common-licenses/{,L}GPL-?,doc/HOWTO},{local/{bin,lib},lib/{gimp/?.?*,emacs/[0-9]*.[1-9]}}`
25. ვთქვათ გაქვთ ფაილი `toto`, რომლის შიგთავსშიც ჩაწერილია სხვა ფაილის სახელი. გამოიტანეთ ამ უკანასკნელის შიგთავსი ეკრანზე. ოპერაცია შეასრულეთ შელში ერთი ბრძანებით<sup>2</sup> მილით ან/და მილის გარეშე.
26. ვთქვათ გაქვთ ფაილი `toto`, რომლის შიგთავსშიც ჩაწერილია სხვა ფაილის სახელი. ამავდროულად მასში კიდევ სხვა ფაილის სახელია ჩაწერილი. გამოიტანეთ ამ უკანასკნელის ზომა ეკრანზე. ოპერაცია შეასრულეთ შელში ერთი ბრძანებით მილით ან/და მილის გარეშე.
27. ვთქვათ გაქვთ `foo` დასახელების დირექტორიაში `bar` ფაილი, რომელშიც შიგთავსად ჩაწერილია სხვა ფაილის სრული გზა. გამოიტანეთ ამ უკანასკნელის ზომა ეკრანზე. ოპერაცია შეასრულეთ შელში ერთი ბრძანებით მილით ან/და მილის გარეშე.
28. შექმენით ფაილი დასახელებით `toto` თუ ის არ არსებობს.
29. შექმენით ფაილი დასახელებით `toto`, თუ `titi` დასახელების ფაილი არ არსებობს. ეკრანზე შეცდომის შეტყობინება არ გამოიტანოთ.

---

<sup>2</sup> ოპერაციის ერთი ბრძანებით შესრულებაში იგულისხმება შედგენილი ბრძანებაც, რომელიც გადაბმულია სხვადასხვა ოპერატორებით.

30. შექმენით ფაილი დასახელებით toto თუ file დასახელების ფაილი არსებობს, თუ არადა შექმენით ფაილი titi. ეკრანზე შეტყობინება არ გამოიტანოთ.
31. გაუშვით „ls -l“ ბრძანება ერთ ტერმინალზე, ხოლო მისი შედეგი გამოიტანეთ მეორე ტერმინალზე.
32. გაუშვით toto ფაილის შიგთავსის ნახვის ბრძანება ერთ ტერმინალზე, ხოლო მისი შედეგი გამოიტანეთ მეორე ტერმინალზე.
33. აკოპირეთ toto ფაილის შიგთავსის titi-ში. თუ toto ფაილი არ არსებობს არ გამოიტანოთ ეკრანზე შეცდომის შეტყობინება.
34. დაამატეთ /etc დირექტორიაში არსებული ყველა conf გაფართოების ფაილების სია თქვენი პირადი დირექტორიის list\_conf.txt ფაილში. ოპერაცია შეასრულეთ შელში ერთი ბრძანებით!
35. დაამატეთ /etc დირექტორიაში არსებული ყველა conf გაფართოების ფაილების სია თქვენი პირადი დირექტორიის list\_conf.txt ფაილში და ამავდროულად, ეს სია ეკრანზეც გამოიტანეთ. ოპერაცია შეასრულეთ შელში ერთი ბრძანებით!
36. დაამატეთ პირად დირექტორიაში არსებული ყველა txt გაფართოების ფაილების სია /tmp/list.txt ფაილში და ამავდროულად ეს სია ეკრანზეც გამოიტანეთ. ოპერაცია შეასრულეთ შელში ერთი ბრძანებით!
37. /etc დირექტორიის ყველა ფაილის სახელი გადაამისამართეთ ფაილ liste\_etc.txt-ში და კონფიგურაციის ყველა ფაილი კი liste\_etc\_conf.txt-ში. ოპერაცია შეასრულეთ შელში ერთი ბრძანებით!
38. ვთქვათ გაქვთ 3 ფაილი (n1.txt, n2.txt, n3.txt), რომლებშიც ჩაწერილია შემთხვევითი რიცხვები. გამოითვალეთ შემდეგი არითმეტიკული გამოსახულება ამ ფაილებში მითითებული რიცხვებით:  $(n1 + (n2 * n3)) / n2$
39. გამოიტანეთ  $\pi$ -ს მნიშვნელობა მძიმის შემდეგ 1000 ციფრის სიზუსტით.
40. გამოიტანეთ 12345 ხარისხად 6789 მნიშვნელობა.
41. შექმენით ცვლადი File, რომლისაც მნიშვნელობად გარკვეულ არითმეტიკულ გამოსახულებას მიანიჭებთ (მაგალითად  $12+3-5$ ). შექმენით ფაილი იმ დასახელებით, რასაც ამ არითმეტიკული გამოსახულების გამოთვლის შედეგად მიიღებთ. ოპერაცია შეასრულეთ შელში ერთი ბრძანებით!
42. შექმენით ცვლადი File, რომლისაც მნიშვნელობად გარკვეულ არითმეტიკულ გამოსახულებას მიანიჭებთ (მაგალითად  $12+3-5$ ). გახსენით ფაილი დასარედაქტირებლად იმ დასახელებით, რაც ამ არითმეტიკული გამოსახულების გამოთვლის შედეგად მიიღებთ, შეიტანეთ რამდენიმე ხაზი და აკოპირეთ ეს შიგთავსი სხვა ფაილში, რომლის დასახელებას „1“ -ით მეტია მიღებულ არითმეტიკულ შედეგზე. ოპერაცია შეასრულეთ შელში ერთი ბრძანებით!
43. გააფერადეთ შელის მოსაწვევი (Shell Prompt string).

44. შეცვალეთ შელის მოსაწვევი შემდეგნაირად: „I'm in ... (მიმდინარე დირექტორიის სახელი), Now it is ... (მიმდინარე დრო), this is ... (ბრძანების ნომერი)th command“

45. შექმენით შემდეგი ტიპის ფერადი shell მოწვევა. « My processes number is - N \$ ».

46. რას შეასრულებს შელის შემდეგი ბრძანება:

```
$ cat <ls> cat
```

- 1) „ls“ ფაილის შიგთავსს გადასცემს „cat“ ბრძანებას სტანდარტულ შესასვლელზე
- 2) გამოიტანს „<ls>“ და „cat“ ფაილების შიგთავსს ეკრანზე ერთიმეორეს მიყოლებით
- 3) „ls“ ფაილის შიგთავსს გადააწერს „cat“ დასახელების ფაილს
- 4) სინტაქსურად არასწორი ჩანაწერია
- 5) „ls“ ბრძანების შედეგს გადაიტანს „cat“ ფაილში

47. რას შეასრულებს შელის შემდეგი ბრძანება:

```
$ cat <cat> ls
```

- 6) „ls“ ფაილის შიგთავსს გადასცემს „cat“ ბრძანებას სტანდარტულ შესასვლელზე
- 7) გამოიტანს „<cat>“ და „ls“ ფაილების შიგთავსს ეკრანზე ერთიმეორეს მიყოლებით
- 8) „cat“ ფაილის შიგთავსს გადააწერს „ls“ დასახელების ფაილს
- 9) სინტაქსურად არასწორი ჩანაწერია
- 10) „cat“ ბრძანების შედეგს გადაიტანს „ls“ ფაილში

48. რას შეასრულებს შელის შემდეგი ბრძანება:

```
$ cat <<ls>> sort
```

```
> pwd  
> mkdir  
> ls  
$
```

- 1) „<<ls>>“ ფაილის შიგთავსს გადასცემს „sort“ ბრძანებას სტანდარტულ შესასვლელზე
- 2) გამოიტანს „<<ls>>“ და „cat“ ფაილების შიგთავსს ეკრანზე ერთიმეორეს მიყოლებით
- 3) „sort“ ფაილის შიგთავსს ბოლოში დაამატებს კლავიატურიდან შეტანილ პირველ 2 ხაზს.
- 4) სინტაქსურად არასწორი ჩანაწერია
- 5) „sort“ ფაილის შიგთავსს ბოლოში დაამატებს „ls“ ბრძანების შედეგს

49. რას შეასრულებს შელის შემდეგი ბრძანება:

```
$ wc file.txt | tee wc
```

- 1) დაითვლის file.txt ფაილში ახალ ხაზზე გადასვლის, სიტყვისა და ბაიტების რაოდენობას და აკოპირებს მას wc ფაილში
- 2) გახსნის file.txt ფაილსა და wc ფაილს wc ტექსტურ რედაქტორში

- 3) დაითვლის `file.txt` ფაილში ახალ ხაზზე გადასვლის, სიტყვისა და ბაიტების რაოდენობას და გახსნის `wc` ფაილს `tee` ბრძანებით
- 4) სინტაქსურად არასწორი ჩანაწერია
- 5) დაითვლის `file.txt` ფაილში ახალ ხაზზე გადასვლის, სიტყვისა და ბაიტების რაოდენობას და გადასცემს მას `tee`-სა და `wc` ბრძანებას სტანდარტულ შესასვლელზე

50. რას შეასრულებს შელის შემდეგი ბრძანება:

```
$ wc tee | tee wc
```

- 1) დაითვლის `tee` ფაილში ახალ ხაზზე გადასვლის, სიტყვისა და ბაიტების რაოდენობას და აკოპირებს მას `wc` ფაილში
- 2) გახსნის `tee` ფაილსა და `wc` ფაილს `wc` ტექსტურ რედაქტორში
- 3) დაითვლის `tee` ფაილში ახალ ხაზზე გადასვლის, სიტყვისა და ბაიტების რაოდენობას და გახსნის `wc` ფაილს `tee` ბრძანებით
- 4) სინტაქსურად არასწორი ჩანაწერია
- 5) დაითვლის `tee` ფაილში ახალ ხაზზე გადასვლის, სიტყვისა და ბაიტების რაოდენობას და გადასცემს მას `tee`-სა და `wc` ბრძანებას სტანდარტულ შესასვლელზე

51. განსხვავდება თუ არა ამ ორი ბრძანების შედეგი:

```
$ command 2>&1 >file
```

```
$ command >file 2>&1
```

- 1) პირველი ბრძანება `command`-ის სტანდარტულ გამოსასვლელს გადაიტანს `file`-ში, ხოლო მეცდომას ეკრანზე, მეორე კი `command`-ის სტანდარტულ და შეცდომების გამოსასვლელს გადაიტანს `file`-ში.
- 2) პირველი ბრძანება `command`-ის სტანდარტულ და შეცდომების გამოსასვლელს გადაიტანს `file`-ში, მეორე კი `command`-ის სტანდარტულ გამოსასვლელს გადაიტანს `file`-ში, ხოლო მეცდომას ეკრანზე.
- 3) ორივე ბრძანება `command`-ის სტანდარტულ და შეცდომების გამოსასვლელს გადაიტანს `file`-ში.
- 4) სინტაქსურად არასწორი ჩანაწერია პირველი ბრძანება.
- 5) სინტაქსურად არასწორი ჩანაწერია მეორე ბრძანება.

52. გახსენით ტერმინალი და გაუშვით შემდეგი ბრძანებები:

```
$ VAR1="გამარჯობა"
```

```
$ VAR2="საქართველო"
```

```
$ export VAR1
```

```
$ bash
```

```
$ echo $VAR1 $VAR2
```

```
$ exit
```

```
$ echo $VAR1 $VAR2
```

„echo“ ბრძანება ერთნაირ შედეგს გამოიტანს ორივე შემთხვევაში? თუ კი რატომ? თუ არა რატომ?

53. გახსენით ტერმინალი და გაუშვით შემდეგი ბრძანებები:

```
$ VAR1="გამარჯობა"
$ VAR2="საქართველო"
$ export VAR1
$ echo $VAR1 $VAR2
$ unset VAR1
$ echo $VAR1 $VAR2
„echo“ ბრძანება ერთნაირ შედეგს გამოიტანს ორივე შემთხვევაში? თუ
კი რატომ? თუ არა რატომ?
```

54. გახსენით ტერმინალი და გაუშვით შემდეგი ბრძანებები:

```
$ VAR1="გამარჯობა"
$ VAR2="საქართველო"
$ export VAR1 VAR2
$ echo $VAR1 $VAR2
$ export -n VAR1
$ bash
$ echo $VAR1 $VAR2
$ exit
$ echo $VAR1 $VAR2
„echo“ ბრძანება ერთნაირ შედეგს გამოიტანს სამივე შემთხვევაში? თუ
კი რატომ? თუ არა რატომ?
```

55. გახსენით ტერმინალი და შექმენით ცვლადები შემდეგი მნიშვნელობებით:

```
$ VAR1="გამარჯობა"
$ VAR2="საქართველო"
```

რა შედეგი გამოვა ეკრანზე თითოეული ბრძანების შესრულებისას:

- 1) echo \$VAR1\$VAR2
- 2) echo \$VAR1VAR2
- 3) echo \${VAR1}\${VAR2}
- 4) echo \$VAR13\$VAR24
- 5) echo \${VAR1}\${VAR2}4
- 6) echo \$VAR1\${VAR2}
- 7) echo \$VAR1{VAR2}

56. აკოპირეთ /bin დირექტორიიდან \$HOME/bin-ში ls ბრძანება და გაუშვით.

57. გადაანაცვლეთ cp და mv ბრძანებების ეფექტები (ყურადღება მიაქციეთ იმას, რომ თქვენ არ გაქვთ ადმინისტრატორის უფლებები!)

58. დააცარიელეთ \$HOME/tmp დირექტორია. წაშალეთ დამალული ფაილებიც. (ყურადღება მიაქციეთ იმას, რომ დირექტორიაში არ უნდა შეხვდეთ . და .. დასახელების ქვედირექტორიებს).

59. რომელი ბრძანება დააცარიელებს \$HOME/tmpfile ფაილს და გაანულებს მის ზომას (მრავალი პასუხი):

- 1) cat /dev/null > \$HOME/tmpfile
- 2) cp /dev/null > \$HOME/tmpfile

- 3) `echo "" > $HOME/tmpfile`
- 4) `echo > $HOME/tmpfile`
- 5) `echo -n > $HOME/tmpfile`
- 6) `true > $HOME/tmpfile`
- 7) `> $HOME/tmpfile`
- 8) `truncate -s0`

60. ვთქვათ თქვენს სისტემაში გაქვთ ორი ბირთვის ფაილი: `/boot/vmlinuz-3.16.0-4-amd64` და `/boot/vmlinuz-4.9.0-4-amd64`. დაიმახსოვრეთ მათ შორის განსხვავება `vmlinuz3.16-4.9` ფაილში და ერთი ბირთვის დაკარგვის შემთხვევაში აღადგინეთ ის.
61. ბრძანება « `cmp` », რომლის დანიშნულებაცაა ორი სხვადასხვა ფაილის ერთმანეთთან შედარება, განხორციელდება წარმატებით ან წარუმატებლად. წარმოიდგინეთ ორივე შესაძლებლობა და მოიყვანეთ ამის მტკიცებულება.
62. დაყავით `toto` ფაილი ისე, რომ დანაყოფი ფაილები შეიცავდნენ `toto` ფაილის 10-10 ხაზს. დანაყოფების სახელის სუფიქსი იყოს `toto_`
63. `toto` ფაილი დაყავით `N` თანაბარ ნაწილად პრეფიქსით: `toto_part` და ციფრული სუფიქსებით.
64. `toto` ფაილი დაყავით ისე, რომ თითოეული დანაყოფის ფაილი შეიცავდეს `toto` ფაილის 5 სტრიქონს.
65. აიღეთ `bmp` გაფართოების (ან დაუმუშავებელი სურათის ერთ-ერთი ფაილი) და დაყავით `N` ნაწილად. კერძოზე გამოიტანეთ:
  - სურათის მხოლოდ პირველი 2/3 ნაწილი
  - სურათის მხოლოდ პირველი 1/10 ნაწილი
  - წანაცვლეთ დანაყოფები `bmp` გაფართოების ფაილში (მე-2 დანაყოფი მე-3 ში გადაიტანეთ, მე-3 მე-4-ში, ა.შ. და ბოლო მე-2-ში. პირველ ნაწილს არ შეეხოთ!). ნახეთ და შედეგს მიიღებთ.
66. 6 მბ ზომის `mp3` ფაილი დაყავით ისე, რომ მისი ნაწილები ტრანსპორტირებადი იყოს დისკეტების მეშვეობით (ზომა - 1.44 MB). გადაწერის შემდეგ, ააწყეთ საწყისი ფაილი.
67. მოძებნეთ ყველა `html` ფაილი თქვენს პირად დირექტორიაში და შეინახეთ მათი დასახელებები `fichier.html`.
68. თქვენი პირადი დირექტორიიდან უკანა პლანზე გაუშვით ყველა `html` ფორმატის ფაილის ძიება ისე რომ ისინი გადაამისამართოთ ფაილ `$HOME/fichier.html`-ში.
69. მოძებნეთ ყველა ფაილი, რომლის სახელიც იწყება „T“ ასო-ნიშნით ან ყველა დირექტორია, რომელის სახელიც მთავრდება „N“ ასო-ნიშნით.
70. მოძებნეთ 2 დღის წინ შეცვლილი ყველა ფაილი, რომელთა ზომა 23-დან 145 ბაიტამდეა. თან წარმოადგინეთ ვიზუალური მტკიცებულება კერძოზე.
71. მოძებნეთ ყველა ფაილი, რომელიც მოდიფიცირებულ იქნა 1 დღის წინ და ამავდროულად `toto` ფაილის მოდიფიკაციის შემდეგ.



72. მოძებნეთ ყველა ფაილი, რომელიც შეიცავს 45 დღის წინ და toto ფაილის მოდიფიკაციის შემდგომ.
73. მოძებნეთ ყველა ფაილი, რომელიც შეიცავს ბოლო 45 დღეში და toto ფაილის მოდიფიკაციის შემდგომ.
74. მოძებნეთ 45 ბოლო წუთის განმავლობაში შეცვლილი ფაილები.
75. მოძებნეთ ყველა ფაილი, რომელიც იწყება ასო « a » -ზე და რომლებიც დირექტორიის ტიპის ფაილებია ან ფაილები, რომლებიც ბოლოვდებიან ასო « k » ასოზე და რომლებიც არ არიან დირექტორიის ტიპის ფაილები.
76. აჩვენეთ განსხვავება `find . -size n, -n, +n` შორის.
77. მოძებნეთ ყველა ფაილი, რომელთაც აქვთ მხოლოდ და მხოლოდ წვდომის შემდეგი უფლებები: წაკითხვა მფლობელისთვის, ისევე როგორც ჯგუფისთვის და წაკითხვა/მოდიფიკაცია სხვა მომხმარებლებისთვის.
78. მოძებნეთ ყველა ფაილი, რომელთაც აქვთ ჩამოთვლილთაგან ერთ-ერთი უფლება მაინც: წაკითხვა მფლობელისთვის, ისევე როგორც ჯგუფისთვის, და წაკითხვა/მოდიფიკაცია სხვა მომხმარებლებისთვის.
79. მოძებნეთ ყველა ფაილი, რომელთაც აქვთ მხოლოდ და მხოლოდ შემდეგი უფლებები: წაკითხვა მესაკუთრეებისთვის, ისევე როგორც ჯგუფისთვის, წაკითხვა/მოდიფიკაცია სხვა მომხმარებლებისთვის და გადაადგილეთ ისინი დირექტორიაში: `$HOME/Problems`.
80. მოძებნეთ ყველა ფაილი, რომელთაც აქვთ შემდეგი უფლებები: წაკითხვა ყველასთვის, წერა მესაკუთრისთვის და ჯგუფისთვის ან სხვა მომხმარებლისთვის, და
81. მოძებნეთ ყველა დირექტორია `sticky bit` წვდომის უფლებით.
82. მოძებნეთ ყველა ფაილი თქვენს პერსონალურ დირექტორიაში, რომელთაც წვდომის არანაირი უფლება არ აქვთ.
83. მოძებნეთ ყველა ფაილი თქვენს პერსონალურ დირექტორიაში, რომელთაც აქვთ აქტიური `SUID bit`.
84. მოძებნეთ ყველა ჩვეულებრივი ფაილი თქვენს პერსონალურ დირექტორიაში, რომელთაც აქვთ უფლება: 777 და გარდაქმენით 644.
85. მოძებნეთ ყველა ფაილი `mp4` ან `avi` გაფართოებით (დიდი და პატარა ასოებით) `/home` დირექტორიიდან და წაშალეთ ისინი.
86. მოძებნეთ ყველა ფაილი `mp4` ან `avi` გაფართოებით (დიდი და პატარა ასოებით), რომელთა ზომა 100 MO-ზე მეტია დირექტორია `/home`-დან და წაშალეთ ისინი.
87. თქვენს პერსონალურ დირექტორიაში წაშალეთ ყველა ცარიელი დირექტორია.
88. მოძებნეთ ყველა ფაილი, რომლებიც წაკითხულ იქნენ ბოლო ორი საათის განმავლობაში.
89. მოძებნეთ კონფიგურაციის ყველა ფაილი (`.conf`) `/etc` დირექტორიაში და არა ქვედირექტორიებში.

90. მოძებნეთ ის ფაილი, რომელიც ყველაზე გვიან წაიკითხეს, და ამავდროულად მისი შიგთავსი შეცვალეს `toto` ფაილი შემდეგ და გახენით ის `vim`-ში.
91. აჩვენეთ მიმდინარე მომხმარებლის კუთვნილი პროცესები.
92. იპოვეთ `syslog` პროცესის PID.
93. შეადარეთ ერთმანეთს `top`-ის და `root`-ის მიერ გაშვებული პროცესების რაოდენობა.
94. პროცესორის გამოყენებადობის მიხედვით გაფილტრეთ ყველა პროცესი.
95. მეხსიერების გამოყენებადობის მიხედვით გაფილტრეთ პროცესები.
96. რეალურ დროში მეხსიერებიდან გაფილტრეთ პირველი 10 მუშა პროცესი ისე, რომ განახლების პერიოდი იყოს 1 წამი.
97. `ps -aux` განსხვავებულ შედეგს იძლევა ვიდრე `"ps aux"` ?
98. აჩვენეთ პროცესების სია მომხმარებლის მიერ განსაზღვრულ ფორმატში: კერძოდ - PID ნომერი, მფლობელის სახელი და ბრძანების სახელი.
99. გამოიტანეთ იმ პროცესების სია, რომელიც შეიცავს რომელიმე კონკრეტულ გამოსახულებას სახელში (მაგალითად `chrome`).
100. დახურეთ `gedit`-ის ყველა პროცესი ერთბაშად.
101. დახურეთ ყველა პროცესი ერთბაშად, რომლის სახელიც იწყება „chr“-ით.
102. გაუშვით `top` ბრძანება (`batch` რეჟიმში) და შეინახეთ პირველი 10 პროცესი ფაილებში `10p.txt` შემდეგი დამუშავებისთვის.
103. გაუშვით `top` ბრძანება (`batch` რეჟიმში) და შეინახეთ პირველი 5 იტერაციის შედეგად გამოტანილი პროცესების სია ფაილში `5p.txt` შემდეგი დამუშავებისთვის.
104. გაუშვით `top` ბრძანება (`batch` რეჟიმში) მხოლოდ მუშა პროცესების გამოსატანად და შეინახეთ პირველი 10 იტერაციის შედეგად გამოტანილი პროცესების სია ფაილში `10p.txt` შემდეგი დამუშავებისთვის.
105. უკანა პლანზე გაუშვით ბრძანება `tee`. გადაიტანეთ ეს პროცესის წინა პლანზე, შემდეგ კი ხელახლა განახორციელეთ ამ ბრძანების უკანა პლანზე გადატანა და გაშვება (ბრძანებებით: `kill`, `bg` და `fg`)
106. აჩვენეთ `firefox` -ის PID (მხოლოდ და მხოლოდ PID), თუ ის გაშვებულია და მოკალით ის, თუ არაა გაშვებული, არ გამოიტანოთ შეცდომა ეკრანზე.
107. აჩვენეთ მიმდინარე დროის წამების, წუთებისა და საათების ჯამი.
108. აჩვენეთ კვირის რომელი დღე იყო 01/01/01.
109. აჩვენეთ კვირის დღე, რომელიც იქნება 2 წელიწადსა, 5 თვესა და 6 დღეში.
110. აჩვენეთ კვირის დღე რომელიც იყო 2 წლის, 5 თვისა და 6 დღის წინ.
111. აჩვენეთ კვირის დღე რომელიც იქნება 2 წლის, 5 თვისა და 6 დღის შემდეგ.

112. აჩვენეთ კვირის დღე, რომელიც იქნება 2 წელიწადში, 5 თვის წინ და 6 დღის შემდეგ.
113. აჩვენეთ UNIX-ის შექმნის შემდგომ განვლილი წამების რაოდენობა.
114. ვთქვათ toto ფაილის ბოლო მოდიფიკაციის თარიღი გამოსახულია წამებში UNIX-ის ეპოქიდან ვთქვათ 1702831970.
115. ვთქვათ toto ფაილის ბოლო მოდიფიკაციის თარიღია 2023-12-17 20:51:27. გამოსახეთ ეს თარიღი წამებში გასული UNIX-ის ეპოქიდან.
116. გამოიტანეთ toto ფაილის პირველი 15 ხაზი.
117. გამოიტანეთ toto ფაილის ბოლო 15 ხაზი.
118. გამოიტანეთ toto ფაილის მე-5-დან მე-15 ხაზამდე.
119. გამოიტანეთ toto ფაილის მესამე ხაზიდან ბოლომდე ყველა ხაზი.
120. გამოიტანეთ toto ფაილის მე-5 სვეტი.
121. გამოიტანეთ toto ფაილის 2-5 სვეტის 5-10 სტრიქონი და შედეგი გადაამისამართეთ შემდეგი დასახელების ფაილისკენ: toto\_YYMMDDHHMM (სადაც, YYMMDDHHMM არის მიმდინარე წელითვედღესაათიწუთი)
122. გამოიტანეთ toto ფაილის შემდეგი სვეტები: მეხუთე სვეტიდან მესამე სვეტამდე, სადაც სვეტებს შორის გამოყოფია : სიმბოლო.
123. გამოიტანეთ toto ფაილის ყველა სვეტი მეხუთე სვეტამდე, სადაც სვეტებს შორის გამოყოფია \$ სიმბოლო.
124. აჩვენეთ toto ფაილის ყველა სტრიქონი, რომელიც შეიცავს დიდ ასოებს უკულმა (ბოლოდან პირველისკენ)
125. აჩვენეთ toto ფაილის ყველა სტრიქონი, რომლებიც მხოლოდ ციფრებს შეიცავს.
126. დავეშვათ, toto ფაილი შეიცავს მხოლოდ ციფრებსა და ლათინურ ასოებს (დიდს და პატარას). აჩვენეთ ყველა სტრიქონი, რომელიც შეიცავს მხოლოდ და მხოლოდ ციფრებს. ამის შემდეგ ყველა სტრიქონი აჩვენეთ შებრუნებით.
127. შექმენით toto ფაილი რომელიც 15 სტრიქონიან ტექსტს შეიცავს. განახორციელეთ ტექსტის ზოგიერთი ელემენტის მოდიფიკაცია: სიტყვა « est » ჩაანაცვლეთ სიტყვით « était » ყველგან, წაშალეთ მეორე, 7-9 და 12 სტრიქონი და ტექსტის დანარჩენი ნაწილი გადაიტანეთ ფაილ titi-ში.
128. შექმენით toto ფაილი, რომელიც შეიცავს 15 სტრიქონიან ტექსტს. მთელ ტექსტში სიტყვა « était » ჩაანაცვლეთ « est »-ით და მეორე მხრივ, ყველა პატარა ასო ჩაანაცვლეთ დიდი ასოებით, გარდა შემდეგი ასოებისა: « e », « s » « t ».
129. იპოვეთ და ეკრანზე გამოიტანეთ განმეორებული ხაზების რაოდენობა toto ფაილიდან.

130. იპოვეთ და ეკრანზე გამოიტანეთ `toto` ფაილში არსებული ასო-ნიშნების რაოდენობა.
131. განსაზღვრეთ თქვენს პერსონალურ დირექტორიაში არსებული ფაილების რაოდენობა.
132. დავუშვათ, რომ გაქვთ დირექტორია: `$HOME/etudiant/`, რომელიც მრავალი ფაილია შენახული. თითოეულ ფაილში თითოეულ სტრიქონზე წარმოდგენილია სტუდენტების სახელი, გვარი, დაბადების თარიღი და ადგილი. ყველა ეს ინფორმაცია შეკრიბეთ ერთ ფაილში – `etudiant`, რომელშიც შევა თითოეული სტუდენტის სახელი, გვარი, დაბადების თარიღი და ადგილი ყოველ სტრიქონზე.
133. დავუშვათ, რომ გაქვთ 4 ფაილი: `nom` (სტუდენტების გვარებს შეიცავს ყოველ სტრიქონზე) `prenom` (სტუდენტების სახელებს შეიცავს ყოველ სტრიქონზე), `date` (სტუდენტების დაბადების თარიღს შეიცავს ყოველ სტრიქონზე), `lieu` (სტუდენტების დაბადების ადგილს შეიცავს ყოველ სტრიქონზე). ყველა ეს ინფორმაცია შეკრიბეთ ერთ ფაილში – `etudiant`, რომელშიც შევა ქვემოთ მოხსენიებული დეტალები: ყოველ სტრიქონზე თითოეული სტუდენტის გვარი, სახელი, თარიღი და დაბადების ადგილით.
134. დავუშვათ, რომ გაქვთ 3 ფაილი: `number.txt`, რომელიც შეიცავს თანმიმდევრულ რიცხვებს ხაზ-ხაზ, `surname.txt`, რომელიც შეიცავს სტუდენტების გვარებს ხაზ-ხაზ და `name.txt`, რომელიც შეიცავს სტუდენტების სახელებს ხაზ-ხაზ. შეკრიბეთ ეს ინფორმაცია ერთ ფაილში `students.txt`, რომელიც ყოველ ხაზზე შეიცავს რიცხვს, სტუდენტის სახელს, იგივე რიცხვს, სტუდენტის გვარს.
135. დავუშვათ, რომ გაქვთ 3 ფაილი: `number.txt`, რომელიც შეიცავს თანმიმდევრულ რიცხვებს ხაზ-ხაზ, `surname.txt`, რომელიც შეიცავს სტუდენტების გვარებს ხაზ-ხაზ და `name.txt`, რომელიც შეიცავს სტუდენტების სახელებს ხაზ-ხაზ. შეკრიბეთ ეს ინფორმაცია ერთ ფაილში `students.txt`, რომელიც ყოველ ხაზზე შეიცავს რიცხვს, სტუდენტის სახელს, სტუდენტის გვარს, სადაც რიცხვსა და სახელს შორის გამყოფი სიმბოლოა `:`, ხოლო სახელსა და გვარს შორის გამყოფი სიმბოლოა `„`.
136. ვთქვათ, მოცემული გაქვთ ერთი ფაილი `students.txt`, სადაც მოცემულია `N` სტუდენტის მონაცემები (სახელი, გვარი, მისამართი) შემდეგი ფორმატით:

```

student_name1
student_surname1
student_address1
student_name2
student_surname2
student_address2
...
student_nameN
student_surnameN
student_addressN

```

შეინახეთ ეს მონაცემები `students2.txt`-ში შემდეგნაირად

```
student_name1_student_surname1,student_address1
student_name2_student_surname2,student_address2
...
student_nameN_student_surnameN,student_addressN
```

სადაც პირველი და მეორე სვეტს შორის გამყოფი სიმბოლოა ‘,’ ხოლო მეორე და მესამე სვეტს შორის გამყოფი სიმბოლოა ‘:’:  
ოპერაცია შეასრულეთ ერთი ბრძანებით!

137. შექმენით D1 და D2 დასახელების დირექტორიები თქვენს პირად დირექტორიაში და მათში გადააკოპირეთ ყველა ის ბრძანება, რომელსაც გააქტიურებული აქვს SGID ან/და SUID ბიტი.
138. მოძენეთ ის ფაილი, რომელიც ყველაზე გვიან ნახეს თქვენს პირად დირექტორიაში, და ამავდროულად, მისი შიგთავსი შეცვალეს toto ფაილი შემდეგ, და გახსენით ის vim ტექსტურ რედაქტორში readonly რეჟიმში და თან კურსორი გადაადგილეთ მესამე სტრიქონზე ავტომატურად. ოპერაცია შეასრულეთ ერთი ბრძანებით!
139. შექმენით ტექსტური ფაილი foobar.txt (მაგალითისთვის სამხაზიანი). დასაწყისსა და ბოლოში ჩაამატეთ ე.წ. Bell სიმბოლო, ხოლო ყოველი ხაზის შუაში (თვალის ზომით) შეიტანეთ Carriage Return სიმბოლო. დარწმუნდით, რომ cat ბრძანებით მისი შიგთავსი გამოტანისას არ გამოისახება სრული ინფორმაცია. იპოვეთ გამოსავალი, რომ ნახოთ სრული შიგთავსი და იდენტიფიკაცია გაუკეთეთ მასში არსებულ უხილავ სიმბოლოებს.
140. ვთქვათ გაქვთ ორი ფაილი f1.txt და f2.txt, რომელთაც აქვთ პირველი ველი (სვეტი) საერთო. შეურიეთ ერთმანეთს ამ ორი ფაილის შიგთავსი ისე, რომ საერთო ველი ერთხელ გამოჩნდეს და არა ორჯერ.
141. ვთქვათ გაქვთ ორი ფაილი f1.txt და f2.txt, რომელთაც აქვთ საერთო ველი (სვეტი), f1.txt-ში მე-2, ხოლო f2.txt-ში მე-3. შეურიეთ ერთმანეთს ამ ორი ფაილის შიგთავსი ისე, რომ საერთო ველი ერთხელ გამოჩნდეს ხაზის დასაწყისში.
142. აჩვენეთ jpeg და png ფორმატის ყველა ფაილის ზომები (დიდი და პატარა ასოებს მნიშვნელობა აქ მიანიჭოთ) თქვენი პერსონალური დირექტორიიდან.
143. აჩვენეთ თქვენი პერსონალური დირექტორიის ყველა დირექტორიის ზომა ინდივიდუალურად და აგრეთვე საერთო ზომა.
144. აჩვენეთ ფაილები თქვენი პირადი დირექტორიიდან ერთ სვეტად.
145. შექმენით თქვენს პირად დირექტორიაში არსებული ფაილების არქივი დასახელებით archive\_YMMDD.tar
146. შექმენით თქვენს პირად დირექტორიაში არსებული ფაილების შეკუმშული (gzip-ით) არქივი დასახელებით archive\_YMMDD.tgz
147. შექმენით თქვენს პირად დირექტორიაში არსებული ფაილების შეკუმშული (bzip2-ით) არქივი დასახელებით archive\_YMMDD.tbz2
148. შემდეგი დღის ღამის 03 საათზე შექმენით არქივის ფაილი: archive\_YMMDD.tar, რომელშიც შევა \$HOME/Documents დირექტორიის

- ყველა ფაილი. ამის შემდგომ, 5 საათსა და 15 წუთში არქივში დაამატეთ \$HOME/tmp დირექტორია.
149. კვირის ერთ დღეს შექმენით არქივის ფაილი: archive.tar, რომელშიც შევა პერსონალური დირექტორიის ყველა ფაილი. მომდევნო დღეს, 21:00 სთ-ზე, წაშალეთ არქივიდან დირექტორია \$HOME/tmp და არქივი შეკუმშეთ Lempel-Ziv-ის ალგორითმით.
150. განახორციელეთ /etc/passwd ფაილის კოპირება ყოველ დღე 20:30-ზე \$HOME/tmp/passwd-ში, გარდა შაბათ-კვირისა.
151. ყოველი თვის პირველ დღეს 15:48-ზე გააკეთეთ თქვენი პერსონალური დირექტორიის სარეზერვო ასლი /tmp-ში და დაარქვით ამ არქივს სახელი შემდეგი ნიმუშის მიხედვით: YYMMDDHHMM.tar (სადაც, YYMMDDHHMM არის მიმდინარე წელითვედღესაათიწუთი)
152. იანვრის თვეში ყოველი შაბათი-კვირის ღამის საათებში (00:00-06:00) ყოველი 10 წუთის ინტერვალში დათვალეთ XYZ მომხმარებლის მიერ გაშვებული პროცესების რაოდენობა და დააგროვეთ ეს მონაცემები /tmp/jan-ფაილში.
153. 25 დეკემბრიდან 2 იანვრამდე ყოველი დღე-ღამის ყოველ 15 წუთში დათვალეთ სისტემაში გაშვებული პროცესების სრული რაოდენობა და დააგროვეთ ეს მონაცემები /tmp/hollydays-ფაილში.
154. 25 დეკემბრიდან 2 იანვრამდე ყოველი დღე-ღამის ყოველ 3 საათში გამოთვალეთ XYZ მომხმარებლის პირადი დირექტორიის ზომა და დააგროვეთ ეს მონაცემები /tmp/XZYsize-ფაილში.
155. შექმენით თქვენს პირად დირექტორიაში არსებული ფაილების არქივი დასახელებით archive\_YYMMDDHHMMSS.tar. გადაიტანეთ \$HOME/dir დირექტორიაში, გადართ და იქ ამოარქივებთ მხოლოდ txt გაფართოების ფაილები.
156. მომავალი წლის 3 თებერვალს 14:48 წუთზე გაიგეთ / დირექტორიის დაკავებულობა და ჩაწერეთ \$HOME/df\_info ფაილში.
157. მომავალი წლიდან ყოველი თვის 3 რიცხვში 14:48 წუთზე გაიგეთ / დირექტორიის დაკავებულობა და ჩაამატეთ ეს ინფორმაცია \$HOME/df\_info ფაილში.
158. აჩვენეთ განსხვავება არქივის ფაილების მთლიან ზომასა და თავად არქივის ფაილის ზომას შორის.
159. მომავალი წლიდან დაწყებული აიძულეთ user1, user2 და user3 მომხმარებლები შეცვალონ თავიანთი პაროლი თუ ისინი შედიან sudo ჯგუფში.
160. მომავალი წლიდან ყოველი თვის შაბათ-კვირას ღამის 2, 4, 7 საათზე გამოთვალეთ თქვენი პირადი დირექტორიის ზომა და ჩაწერეთ \$HOME/size\_info ფაილში.
161. შექმენით ფაილი toto \$HOME/tmp-ში, რომლის მხოლოდ წაკითხვა შეგიძლიათ.
162. შექმენით ფაილი toto \$HOME/tmp-ში, რომლის წაკითხვას და მოდიფიცირებას შეძლებთ, მაგრამ ვერ შეძლებთ მის წაშლას.
163. შექმენით ფაილი toto \$HOME/tmp-ში, რომლის წაკითხვა და წაშლა შეგიძლებათ, მაგრამ არა მისი მოდიფიცირება.

164. შექმენით ფაილი `toto $HOME/titi/`-ში იმგვარად, რომ თქვენთვის შეუძლებელი იყოს `titi`-ს შიგთავსის ნახვა, მაგრამ შეგეძლოთ `toto`-ს შიგთავსის ნახვა.
165. მიეცით უფლება სხვა მომხმარებლებს წაშალონ თქვენი საკუთარი ფაილები თქვენი პირადი დირექტორიიდან.
166. მიეცით უფლება სხვა მომხმარებლებს წაშალონ თავისი კუთვნილი ფაილები თქვენი პირადი დირექტორიიდან.
167. გახსენით ის ტექსტური ფაილი `vi`-ის მეშვეობით, რომელიც უკანასკნელად იქნა მოდიფიცირებული თქვენს პერსონალურ დირექტორიაში.
168. რამდენი დაფარული ფაილი გაქვთ თქვენს პირად დირექტორიაში?
169. რამდენი დაფარული დირექტორია გაქვთ თქვენს პირად დირექტორიაში?
170. რამდენი დაფარული ფაილი გაქვთ (ჩვეულებრივი ტიპის ფაილები, დირექტორიები) თქვენს პირად დირექტორიაში?
171. თქვენი პერსონალური დირექტორიიდან აჩვენეთ ფაილები შესაბამისი სიმბოლოს დამატებით (ამ სიმბოლოებიდან რომელიმე « \*/=>@| ») (ყოველი შესვლისას) და გააფერადეთ ამავე დროს (ექსპლიციტურად)
172. აჩვენეთ ყველა დაფარული ფაილი თქვენი პერსონალური დირექტორიიდან და დაამატეთ ეს სია ფაილ `f_ca.txt`-ში.
173. გახსენით `script.sh` ფაილი `vim`-ის მეშვეობით მხოლოდ კითხვით რეჟიმში.
174. გახსენით `script.sh` ფაილი `vi`-ს მეშვეობით და ამავედროულად კურსორი გადაადგილეთ მათე სტრიქონზე ავტომატურად.
175. გაფილტრეთ ფაილ `/etc/passwd`-ის შიგთავსი მესამე სვეტის მიხედვით.
176. გამოიტანეთ ეკრანზე `conf` გაფართოების ფაილების სია `/etc` დირექტორიიდან და დააკოპირეთ ნაპოვნი ფაილები USB (`/media/usb`) -ზე.
177. მოძებნეთ `conf` გაფართოების ყველა ფაილი `/etc/` დირექტორიიდან და დააარქივეთ ისინი `conf.tar` ფაილში თქვენს პირად დირექტორიაში.
178. რა ზომას (ბაიტებში) იკავებს თქვენი პირადი დირექტორია?
179. აჩვენეთ ბაიტებში ფაილური სისტემების მიერ დაკავებული სივრცე. (არ გამოიტანოთ ვირტუალური ფაილური სისტემები)
180. ყველა (`$HOME/bin/*`) სკრიპტები დააკოპირეთ USB (`/media/usb`)-ში, ისე რომ შეინარჩუნოთ შემდეგი ატრიბუტები: ფორმა, საკუთრება, დროის მაჩვენებლები.
- 181.

Grep ბრძანებით:

182. გამოიტანეთ `toto` ფაილის ყველა ის სტრიქონი, რომელიც შეიცავს `Linux` გამოსახულებას.

183. გამოიტანეთ *toto* და *titi* ფაილის ყველა სტრიქონი, რომელიც შეიცავს *Linux* გამოსახულებას.
184. გამოიტანეთ *toto* ფაილის ყველა სტრიქონი, რომელიც შეიცავს *Linux* გამოსახულებას დიდი ან/და პატარა ასოებით.
185. გამოიტანეთ *toto* ფაილის ყველა სტრიქონი, რომელიც შეიცავს *Linux* ან *linux* გამოსახულებას.
186. რამდენი სტრიქონი შეიცავს *Linux* და *linux* გამოსახულებებს *toto* ფაილში.
187. გამოიტანეთ *toto* ფაილის ყველა სტრიქონი, რომელიც შეიცავს *Linux* ან *linux* გამოსახულებებს და ასევე 2 სტრიქონს მათ წინ.
188. გამოიტანეთ *toto* ფაილის ყველა სტრიქონი, რომელიც შეიცავს *Linux* ან *linux* გამოსახულებებს და ასევე 2 სტრიქონს მათ შემდეგ.
189. გამოიტანეთ *toto* ფაილის ყველა სტრიქონი, დანომრილი სახით, რომელიც შეიცავს *Linux* ან *linux* გამოსახულებებს და ასევე 2 სტრიქონს მათ წინ და შემდეგ.
190. გამოიტანეთ თქვენი პირადი დერექტორიიდან (იგულისხმება ქვე-დირექტორიებიც) იმ ფაილების სია, რომელიც შეიცავს *Linux* ან *linux* გამოსახულებებს.
191. გამოიტანეთ *toto* ფაილის ყველა სტრიქონი, რომელიც შეიცავს სიტყვას: *Linux* ან *linux*.
192. გამოიტანეთ *toto* ფაილის ყველა სტრიქონი, რომელიც შეიცავს გამოსახულებას: « .\* ».
193. გამოიტანეთ *toto* ფაილის ყველა სტრიქონი, რომელიც შეიცავს გამოსახულებას: « .\* » ან « .| ».
194. გამოიტანეთ *toto* ფაილის ყველა სტრიქონი, რომელიც იწყება გამოსახულებით: BTU.
195. გამოიტანეთ *toto* ფაილის ყველა სტრიქონი, რომელიც მთავრდება გამოსახულებით: FFI.
196. გამოიტანეთ *toto* ფაილის მხოლოდ ანბანურრიცხვითი სტრიქონები.
197. გამოიტანეთ *toto* ფაილის მხოლოდ რიცხვითი სტრიქონები.
198. გამოიტანეთ *toto* ფაილის ის სტრიქონები, რომლებიც მხოლოდ *N* სიმბოლოსგან შედგება.
199. გამოიტანეთ *toto* ფაილის სტრიქონები, რომლებიც შეიცავენ შემდეგ IP მისამართებს : 192.168.2.105, 217.147.231.59.
200. გამოიტანეთ *toto* ფაილის სტრიქონები, რომლებიც შეიცავენ მობილური ტელეფონების ნომრებს საქართველოში (+9955xxxxxxxxx) .
201. გამოიტანეთ ფაილ *.bash\_profile* -ის სტრიქონები, რომლებიც მხოლოდ კომენტარს შეიცავენ (სტრიქონის დასაწყისში გამოტოვება არ ითვლება)



202. გამოიტანეთ სია სისტემის ყველა იმ მომხმარებლის, რომელთა სახელები მთავრდება ერთი ან რამდენიმე რიცხვით.
  203. გამოიტანეთ მოცემული ფაილის არა ცარიელი სტრიქონები (გამოტოვება და ტაბულაცია არ ითვლება)
  204. გამოიტანეთ მოცემული ფაილის არა ცარიელი სტრიქონები (გამოტოვება და ტაბულაცია ითვლება)
  205. რამდენი მილის ტიპის ფაილი გაქვთ თქვენს პირად დირექტორიაში?
  206. რამდენი ჩველებრივი ტიპის ფაილი და დირექტორია გაქვთ თქვენს პირად დირექტორიაში ?
  207. რამდენი დაფარული ფაილი გაქვთ თქვენს პირად დირექტორიაში?
  208. რამდენი დაფარული დირექტორია გაქვთ თქვენს პერსონალურ დირექტორიაში?
  209. რამდენი დაფარული ფაილი და დირექტორია გაქვთ თქვენს პერსონალურ დირექტორიაში?
  210. გამოიტანეთ ყველა დაფარული ფაილი თქვენი პერსონალური დირექტორიიდან და ეს სია დაამატეთ `f_ca.txt` ფაილს.
  211. გამოიტანეთ `toto` ფაილის სტრიქონები, რომლებიც სულ მცირე ერთხელ მაინც შეიცავენ `'ifg'` გამოსახულებას.
  212. გამოიტანეთ `toto` ფაილის სტრიქონები, რომლებიც სულ მცირე სამჯერ მაინც შეიცავენ `'ifg'` გამოსახულებას.
  213. გამოიტანეთ სტრიქონები, რომლებიც შეიცავენ სიტყვებს « `le` » ან « `la` » (დიდი ან/და პატარა ასოებით) და გამოტანისას ფერით გამოკვეთეთ.
  214. გამოიტანეთ სტრიქონები, რომლებიც შეიცავენ მხოლოდ ნამდვილ რიცხვს.
- გამოიტანეთ `toto` ფაილის ყველა სტრიქონი, რომლებიც შეიცავენ ნამდვილ IP მისამართებს.

Sed ბრძანებით:

215. შეცვალეთ ყველა შემხვედრი `/bin/sh /bin/bash`-ით მოცემულ ფაილში.
216. გამოიტანეთ ფაილ `/etc/passwd`-ის 5 დან 9-მდე სტრიქონები.
217. წაშალეთ `/etc/passwd` ფაილის 5-დან 9-მდე სტრიქონები.
218. წაშალეთ `/etc/passwd` ფაილის პირველი 5 სტრიქონი.
219. გამოიტანეთ `/etc/passwd` ფაილის შიგთავსი ისე, რომ თითოეულ სტრიქონში მეორე შემხვედრი « `:` » ჩაანაცვლოთ « `!` »-ით.
220. გამოიტანეთ `/etc/passwd` ფაილის შიგთავსი ისე, რომ თითოეულ სტრიქონში მესამე შემხვედრის ჩათვლით ყველა « `:` » ჩაანაცვლოთ « `!` »-ით.

221. გამოიტანეთ `/etc/passwd` ფაილის შიგთავსი მომხმარებლის სახელის ველის გარეშე.
222. გამოიტანეთ `/etc/passwd` ფაილის პირველი სვეტი.
223. გამოიტანეთ `/etc/passwd` ფაილის შიგთავსი GID ველის გარეშე.
224. მოცემული ფაილის თითოეული სტრიქონის დასაწყისში დაამატეთ სიტყვა « `début` »
225. გამოიტანეთ `/etc/passwd` ფაილიდან მომხმარებლის სახელისა და მათი UID-ის ველები.
226. გამოიტანეთ სისტემის მხოლოდ იმ მომხმარებელთა სახელები, რომელთა ინტერპრეტატორი არის: `/bin/bash`
227. დაითვალოთ თქვენი პირადი დირექტორიის დირექტორიები (ქვე-დირექტორიები) .
228. წამალეთ `toto` ფაილის ყველა სტრიქონი, რომლებიც შეიცავენ რიცხვით სიმბოლოებს.
229. წამალეთ `toto` ფაილის ყველა რიცხვითი სიმბოლო.
230. შეცვალეთ სიტყვა `Windows` სიტყვით `Linux` `toto` ფაილის 3-დან 7-მდე სტრიქონებში.
231. შეცვალეთ სიტყვა `Windows` `Linux`-ით `toto` ფაილის სტრიქონებში გარდა 3-დან 7-მდე სტრიქონებისა.
232. წამალეთ `toto` ფაილის პირველი სტრიქონი.
233. წამალეთ `toto` ფაილის მესამე სტრიქონი.
234. წამალეთ `toto` ფაილის ბოლო სტრიქონი.
235. წამალეთ `toto` ფაილის სტრიქონები 5-დან 9-მდე.
236. წამალეთ `toto` ფაილის მეხუთეს გარდა ყველა სტრიქონი.
237. წამალეთ `toto` ფაილის ყველა სტრიქონი გარდა 5-დან 9-მდე სტრიქონისა.
238. წამალეთ `toto` ფაილის პირველი და ბოლო სტრიქონი.
239. `toto` ფაილის ყველა სტრიქონზე წამალეთ პირველი N სიმბოლო.
240. `toto` ფაილის ყველა სტრიქონზე წამალეთ ბოლო N სიმბოლო.
241. მე-N სიმბოლო წამალეთ `toto` ფაილის ყველა სტრიქონზე.
242. `toto` ფაილის ყველა სტრიქონზე წამალეთ პირველი და ბოლო სიმბოლოები.
243. `/etc` დირექტორიაში (ქვედირექტორიებშიც) დაითვალოთ ჩვეულებრივი ფაილები.
244. `/etc` დირექტორიაში (ქვედირექტორიებშიც) დაითვალოთ `.conf` გაფართოების ფაილები.

245. დაითვალეთ *toto* ფაილის ყველა ცარიელი სტრიქონები.
246. წაშალეთ *toto* ფაილის ყველა ის სტრიქონი, რომელიც ცარიელია ან მხოლოდ გამოტოვებებია.
247. წაშალეთ *toto* ფაილის სტრიქონები, რომლებიც მხოლოდ დიდი ასოები შედის.
248. წაშალეთ ყველა სტრიქონი, რომელიც არ შეიცავს გამოსახულებას *Linux toto* ფაილში.
249. წაშალეთ ყველა სტრიქონი გარდა იმ სტრიქონებისა, რომლებიც შეიცავენ გამოსახულებას *linux* (დიდი ან პატარა ასოებით) *toto* ფაილში.
250. წაშალეთ ყველა სტრიქონი, რომელიც არ შეიცავს გამოსახულებებს *Linux* ან *linux toto* ფაილში.
251. წაშალეთ ყველა სტრიქონი, რომელიც არ შეიცავს გამოსახულებებს *Linux*, *linux*, *Unix* ან *unix toto* ფაილში.
252. *toto* ფაილიდან წაშალეთ ყველა სტრიქონი, პირველი სტრიქონიდან დაწყებული იმ სტრიქონის ჩათვლით, რომელიც შეიცავს გამოსახულებას *Windows*.
253. *toto* ფაილიდან წაშალეთ ყველა სტრიქონი, დაწყებული იმ სტრიქონიდან, რომელიც შეიცავს გამოსახულებას *Windows* ბოლომდე.
254. წაშალეთ *toto* ფაილის ბოლო სტრიქონი, თუ ის შეიცავს გამოსახულებას *Windows* ან *windows*.
255. წაშალეთ *toto* ფაილის 5-9 სტრიქონები, თუ ისინი შეიცავენ გამოსახულებას *Windows* ან *windows*.
256. წაშალეთ სტრიქონები, რომლებიც შეიცავენ გამოსახულებას *Windows* და მათი მომდევნო სტრიქონიც.
257. წაშალეთ ახალ ხაზზე გადასვლის სიმბოლო *toto* ფაილში.
258. წაშალეთ *toto* ფაილის ყველა ლუწი სტრიქონი.
259. წაშალეთ *toto* ფაილის ყველა კენტი სტრიქონი.
260. წაშალეთ ყველა იმ სტრიქონის მომდევნო სტრიქონი, რომლებიც შეიცავენ გამოსახულებას *Windows*
261. წაშალეთ ყველა იმ სტრიქონის მომდევნო 2 სტრიქონი, რომლებიც შეიცავენ გამოსახულებას *Windows*
262. წაშალეთ ყველა იმ სტრიქონის მომდევნო მე-2 სტრიქონი, რომლებიც შეიცავენ გამოსახულებას *Windows*
263. გადაადგილეთ ყველა ლუწი სტრიქონი კენტი სტრიქონის ბოლოში (1 =>2, 3=>4 ...).
264. *toto* ფაილის თექვსმეტობით ფორმატის შიგთავსი გარდაქმენით დიდ ასოებად.
265. *toto* ფაილის თექვსმეტობით ფორმატის შიგთავსი გარდაქმენით დიდ ასოებად და შეინახეთ ფაილში *titi*.

266. გამოიტანეთ *toto* ფაილის სტრიქონების ნომრები, რომლებიც შეიცავენ გამოსახულებას *linux* (დიდი ან პატარა ასოები)
267. დაითვალოთ *toto* ფაილის სტრიქონები.
268. *toto* ფაილის პატარა ასოებიანი სიმბოლოები დააკონვერტირეთ დიდ ასოებში.
269. *toto* ფაილის დიდ ასოებიანი სიმბოლოები დააკონვერტირეთ პატარა ასოებში.
270. *toto* ფაილის თითოეული სტრიქონის შემდგომ ჩასვით ერთი ცარიელი სტრიქონი.
271. *toto* ფაილის თითოეული სტრიქონის შემდგომ ჩასვით ორი ცარიელი სტრიქონი.
272. *toto* ფაილის თითოეული სტრიქონის წინ ჩასვით ერთი ცარიელი სტრიქონი.
273. *toto* ფაილის თითოეული სტრიქონის წინ ჩასვით ორი ცარიელი სტრიქონი.
274. *toto* ფაილის თითოეული სტრიქონის წინ და შემდეგ ჩასვით ერთი ცარიელი სტრიქონი.
275. *toto* ფაილის თითოეული სტრიქონის წინ და შემდეგ, რომლებიც შეიცავენ გამოსახულებას *Linux*, ჩასვით ერთი ცარიელი სტრიქონი.
276. გამოიტანეთ *toto* ფაილის შიგთავსი შებრუნებით (იგივე ბრძანება *tac*)
277. *Toto* ფაილის შიგთავსი ჩასვით *titi* ფაილის ბოლოში.
278. ჩაწერეთ მიმდინარე თარიღი *toto* ფაილის პირველ სტრიქონამდე და საწყისი ფაილი შეინახეთ *toto.bak -ში*.
279. წაშალეთ გამოტოვებები ან ტაბულაციები თითოეული სტრიქონის დასაწყისში.
280. გამოიტანეთ ის სტრიქონი, რომელშიც მხოლოდ პირველი შემხვედრი გამოსახულება *Windows* შედის.
281. შეცვალეთ მხოლოდ პირველი შემხვედრი გამოსახულება *Windows Linux-*ით.
282. წაშალეთ ის სტრიქონი, რომელშიც მხოლოდ პირველი შემხვედრი გამოსახულება *Windows* შედის.
283. ყველა დიდ ასოიანი სიმბოლო ჩაწერეთ ფრჩხილებში.
284. ყველა სტრიქონი ჩასვით ორმაგ ფრჩხილში.
285. თუ სტრიქონი მთავრდება ბექსლექსით დაამატეთ მას მომდევნო სტრიქონი.