Operációs rendszerek

I.óra

***Jogosultságok:***

-Alapjogok: -Szereplők:

-r -u(user)

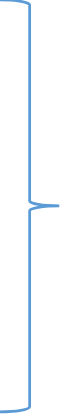
-w -g(group)

-x -o(other)

-a(all)

chmod\_\_(mit)\_\_(mire?)

-/rwx/rw-/r—

[r:4]

[w:2] (Ezek az értékek mindig ezek

[x:1]

|  | fájlok | könyvtárak |
| --- | --- | --- |
| írás | tartalma módosítható | fájlok hozhatóak létre vagy törölhetőek le benne |
| olvasás | tartalma megtekinthető | a benne lévő fájlok ki listázhatóak |
| futtatás | a parancsértelmező tudja futtatni | a benne lévő fájlokat el tudjuk érni/hozzáférünk |

Vannak speciális jogosultságok:

|  | Fájlok | Könyvtárak |
| --- | --- | --- |
| SUID(Set User ID)  [SUID:4] | első x helyén áll  (u+s-sel adható meg)  mindig a tulaj jogaival fut le | semmit sem csinál |
| SGID(Set Group ID)  [SGID:2] | második x helyén áll  (g+s-sel adható meg)  mindig a csoport tulaj jogaival fut le | az itt létrejövő fájlok a csoport tulajdonába kerülnek |
| Sticky bit  [Sticky bit:1] | harmadik x helyén áll  (o+t-vel lehet elérni)  sima fájlok esetén semmit sem tesz | csak a saját tulajdonban lévő fájlokat lehet törölni. |

Elérési utak:

|  | elhelyezkedése: | jele: |
| --- | --- | --- |
| Abszolút | a gyökér könyvtártól kezdve a teljes elérési út | mindig „/”-rel kezdődik |
| Relatív | az aktuális könyvtárhoz viszonyított út | minden ami nem „/”-rel kezdődik |

Behelyettesítések:

-fájlbehelyettesítés:metakarakterek vannak benne

pl:

| „\*” | tetszőleges számú tetszőleges karakter |
| --- | --- |
| „?” | 1db tetszőleges karaktert jelent |
| -[abcd] vagy [a-d]vagy[a-d A-D]  számok esetén  [0-9] | a felsoroltakból egy karakter |
| [^abcd] | a felsoroltak közül bármi lehet |
| [[:alpha:]] | [a-z A-Z] |
| [[:digit:]] | [0-9] |
| [[:alnum:]] | [a-z A-Z 0-9] |
| [[:lower:]] | [a-z] |
| [[:upper:]] | [A-Z] |

Globbing /létrehozásnál!!!!!

{a,b,c,d} vagy {a..d}

-Változó behelyettesítés:

ha van egy változó akkor $-el tudom kérni,hogy az értékét helyettesítse be

pl: $USER

$HOME

$PWD

$PATH

Parancs behelyettesítés:

pl:

Ha a parancs szövegét akarom letárolni akkor x=$(date)- ez esetben letárolja a dátumot

Aritmetikai behelyettesítések:

pl x=3+6

x$((3+6))

Osztás esetén ha „%” -al osztunk akkor 5%3=2(maradékos osztás)

5/3=1(egész osztás)

II.óra

Fájlkezelő parancsok:

1.Mozgatás:”mv”

mv\_\_(mit)\_\_(hova)

pl: File1 File2 Konyvtar2 vagy

konyvtar1 konyvtar2 vagy

file konyvtar/új név🡺ez tulajdonképpen átnevezés is

2.Másolás:”cp”

cp\_(mit)\_\_(hova)

minden ugyanaz kivéve a konyvtar1 konyvtar2

a könyvtár esetén

r konyvtar1 konyvtar2

3. Könyvtár létrehozás:”mkdir”

mkdir Könyvtár1 Könyvtár2 …

4.Törlés:”rm”

rm file1 file2 file3

Mappa törlése esetén:

rm r mappa

rmdir🡺üres mappa törlése

FELADATOK:

1.(8 MAPPA LÉTREHOZÁSA)

- touch mystery\_chapter{1..8}.odf

2.(12 FÁJL LÉTREHOZÁSA)

tv\_season1/2\_episode1-6.ogg

-touch tv\_season{1,2}\_episode{1..6}.ogg

3.(2 MAPPA LÉTREHOZÁSA)

-mkdir/home/hallgato/videók/season{1..2}

4.(A SEASON 1 EPIZÓDJAIT A SEASON 1 MAPPÁBA KELL ÁTHELYEZNI, A MÁSODIKÉT PEDIG A MÁSODIKBA)

-mv tv\_season1\_episode[1..6].ogg Videók/season1

-mv tv\_season2\_episode[1..6].ogg Videók/season2

5 .(DOKUMENTUMOK /MY\_BESTSELLER/CHAPTER LÉTREHOZÁSA)

mkdir –p Dokumentumok/my\_bestseller/chapter

6.(A MY BESTSELLERBE LÉTREHOZNI EGY EDITOR,PLOT\_CHANGE ÉS EGY VACATION MAPPÁT)

-mkdir Dokumentumok/my\_bestseller/editor,plot\_change,vacation

7.(A HOME KÖNYVTÁRBAN LÉVŐ MYSTERY CHAPTER FÁJLOKAT MOZGASSUK A CHAPTERBE)

mv/home/hallgato/mystery\_chapter\*/home/hallgato/Dokumentumok/my\_bestseller/chapter/

vagy!

mv~/myster\_chapter\*.

8.(A CHAPTER 1-2-T AZ EDITORBA KELL TENNI A 7-8-AT A VACATIONBA )

-mv mystery\_chapter[1-2].odf../editor

-mv mystery\_chapter[7-8].odf../vacation

9.(A MÁSODIK ÉVAD ELSŐ EPIZÓDJÁT MÁSOLJUK A VACATION MAPPÁBA, A MYSTERY CHAPTER 5.6 RÉSZÉT PEDIG A PLOT CHANGEBE)

-cp tv\_season2\_episode1.ogg~/Dokumentumok/my\_bestseller/vacation

-cp myster\_chapter[56].odf./plot\_change

10.

mystery\_chapter5\_(mai dátum).odf

-cp mystery\_chapter.odf mystery\_chapter5\_$(date+%f).odf

11.

mystery\_chapter5\_(timestamp).odf

cp mystery\_chapter.odf mystery\_chapter5\_$(date+%s)

12.

mystery\_chapter5\_(felhasználó név).odf

-cp mystery\_chapter5.odf

mystery\_chapter5\_$USER.odf

date+%f🡪év-hó-nap

date+%s🡪timestamp

III.óra

Fájlrendszerek:Egy tároló eszközön vagy annak egy részén van egy kialakított struktúra,aminek a célja a fájlok tárolására,szervezésére szolgál.

\*Alap fájlműveletek:

-létrehozás

-törlés

-bővítés

-csonkítás

-hozzáférés…stb

\*Jogosultságok:

-tömörítés

-titkosítás

-naplózás

-ext(extended file system)/leggyakoribb fájlrendszer

:-Reiser FS(ha nagyon sok pici fájllal kell dolgozni)

:-MiXFS(ha nagyon nagy fájlokkal kell dolgozni)

\*INODE TÁBLÁZAT:

inode:adatszerkezet, ami egy fájlról tárolja a fontos adatokat

inode szám:azonosító, mely egy partíción belül egyedi

ls –i-vel tudjuk megnézni

* Mérete
* Típusa
* Helye(mutatók):/lehet egyszeres,kétszeres és háromszoros
* Jogosultságok
* tulajdonos(felhasználó és hogy melyik csoport tulajdonosa)

\*3 idő bélyegző

* létrehozás
* módosítás
* hozzáférés

Stat paranccsal lehet megnézni mi van benne

* hardlink számláló
* fájl neve nincs benne!!!

könyvtár:/példa

| Név | inode szám |
| --- | --- |
| alma.txt | 12345 |
| körte.txt | 98765 |
| szilva.txt | 12345 |

Linkelés:(2Féle)

* hardlink:egy már létező inodehoz új fájlnevet rendelünk, ha ez a szám 0,akkor lehet törölni

-létrehozása:

ln réginév újnév

* soft link(szimboolikus link/symbolic link)

hasonlít a parancsikonhoz

-új fájlt hoz létre új inode-dal,ami rá mutat egy másik fájlra

ln –s –sel hozható létre

ln –s réginév újnév

A hardlink mindig működni fog, de a soft link „eltörhet”

\*\*Átirányítások:

\*bemenet(stdin)

\*kimenet(stdout)

\*hibakimenet(stderr)

Ha valahova ki akarom írni pl:parancs >ki.txt (ez „>”felül írja a fáj tartalmát)

ha azt akarom hogy ne vesszen el a fájl tartalma csak adjon hozzá akkor parancs>>ki.txt(ez „>>” hozzáfűz)

parancs<be.txt

hibakimenet esetén:

parancs 2>hiba.txt

Lehet két parancsot összekötni (pipe line)

p1|p2(ez a pipe line)

IV.óra

Eddig tanultak:

* cd
* ls
* echo
* touch
* PWD
* wich
* chmod
* mv
* cp
* rm
* rmdir
* mkdir
* mk
* ln

„cat”parancs

„cut”parancs –d, -f

„grep” „minta fájl” azokat a sorokat adja vissza amiben megtalálja a mintát

pl: grep ’sherlock’big.txt

etc/psswd

minden sor 7 elemből áll

| név | x(titkosított jelszó) | UID | GID | komment | home könyvtár | login shell |
| --- | --- | --- | --- | --- | --- | --- |

cut –d:-f1,3/etc/psswd!!

A sor elemeiből az elsőt és a harmadikat írja ki !!

ez esetben a nevet és a User id-t

^ bármelyik sornál keresni fogja a szót a sor elején

PL :^A

’!$’ minden sor végén a felkiáltó jelet fogja keresni

üres sor keresés ’^$’ esetén

\*head

\*tac(cat visszafelé:az utolsó sor lesz az első és az első az utolsó)

\*sort

\*ls –l|sort –k5 -n

-fájlméret szerint sorba rakja a fájlokat

\*ls –l|sort –k5 –n –r

-csökkenőbe iratja ki a tartalmakat

\*ls –l|sort –k5 –n –r|head –n3

-a három legnagyobb sort fogja kiírni

wc (hány sor,szó,karakter van egy fájlban)

wc -l (line:sor szám)

wc –w (szavak száma)

wc –c (karakterek száma)

tail: az utolsó pár sort írja ki

ls –l|tail –n +2

az ls második sorától a végéig kiirja , az első sort pedig levágja

„Szűrők”: szöveges adat feldolgozást végeznek

gyimes.inf.unideb.hu/KMITT/B19/

cat file|parancs==parancs<file

cut(feldarabolni tudunk vele)

V.óra

gedit wordfreq.sh

./wordfreq.sh fájl1.txt fájl2.txt fájl3.txt

Bármilyen kiterjesztésű shell script lehet

Ha azt akarom, hogy bashban lehessen futtatni

Akkor az sh.ban az első sor

#!bin/bash

cat$\* |tr –c a-z A-Z ’\n’ | tr-s ’\n’ | tr A-Z a-z | sort| uniq –c| sort –n –r |head

man tr(2 karakter halmazt kell megadni, ha az elsőben talál olyan karaktert ami egyezik,akkor kicseréli a másik mappában lévőével!!

@ Ez így működik

tr abcd ABCD

tr- s /az ismétlődésektől szabadít meg

tr –c/ ez a tagadása lesz mindennek

$( azt jelenti,hogy az összes dolgon/mappán fusson amit megadunk)

VI. óra

ps –aux vagy aux

pstree/meg lehet vele nézni a viszonyokat

ps-aux|grep’bash’

top

htop/ugyanaz csak színesben

Szignálok:

man 7 signal

-jelzések amit a futó folyamatoknak küldünk kill paranccsal küldünk signalokat

kill paranccsal küldünk signalokat

kill –l-el ki lehet listázni minden signalt

Pl:

SIGINT:interrupt <- Ctrl+c

SIGNOUT:quit<-dump-ot készít

SIGKILL:kill<-nem felülírható!!

SIGTERM:terminnate<.alapértelmezett

SIGSTOP:stop<-ideiglenes felfüggesztés(lehet folytatni)

Ctrl+z a billentyűzet kombinációja, continueval lehet folytatni

kill/mit/ /kinek/

Prioritás kezelés:

-szám<- az alacsonyabb a fontosabb

(3 tagból adódik össze)

alap+futtatási prioritás+niceness

futtatási prioritás=CPU közben folyamatosan nő

Niceness=Ezt mi tudjuk kézzel állítani

niceness(-20-tól +19-ig mehet)

Sima felhasználó csak pozitív nicet adhat míg a root negatívat is

nice –n / egy érték pl10/parancs

renice( már futó folyamaton lehet változtatni)

renice –n/egy érték/PID (\*PROCESSID)

Sima felhasználó itt is csak növelheti és a root csökkentheti

a root bárkiét bármire állíthatja

A háttérben futtatás

parancs&

Pl:./vegtelen.sh&

VIII.óra

gzip(zippelni lehet vele fájlokat

gunzip(unzippelni lehet vele)

gzip fájl->fájl.gz

pl:

gzip pelda3.txt

->pelda3.txt.gz

a folyamat esetén az eredeti állomány eltűnik

gunzip fájl ->fájl

pl:

gunzip pelda3.txt.gz

->pelda3.txt

ha több fájlt adunk meg a gzipnek akkor mindből lesz egy gzip

több állományból egyet akarok létrehozni🡪tar

\*tar –c :létrehozás

\*tar –v:beszédes mód

\*tar –f:fájl név

\*tar –t:listázás

\*tar –r:hozzáfűzés(már létre van hozva de még hozzátesz)

\*tar –x:kicsomagolás

\*tar –z :tömörítés

pl:

\*tar –cvf szoveges\_fajlok.tar\*.txt

-általában így szoktuk létrehozni

-listázása:

\*tar –tvf szoveges\_fajlok.tar

-ha valamit hozzá akarunk adni:

\*tar –rvf szoveges\_fajlok.tar helloworld.sh

-kicsomagolás:

\*tar –rvf szoveges\_fajlok.tar

-tömörítés

\*tar –cvfz orai\_munka.tar.gz\*txt\*.s

-kitömörítés

\*tar –xvfz orai\_munka.tar.gz

Időzített futtatás:

\*egyszeri: „at” paranccsal

pl:

\*\*at now +2 minute

\*\*atq (milyen parancsok várnak futtatásra)

\*\*tty (a terminált azonosítja)

-ha holnapra időzítem de az nekem nem jó akkor atrm-mel tudom removelni (Kivenni a sorból)

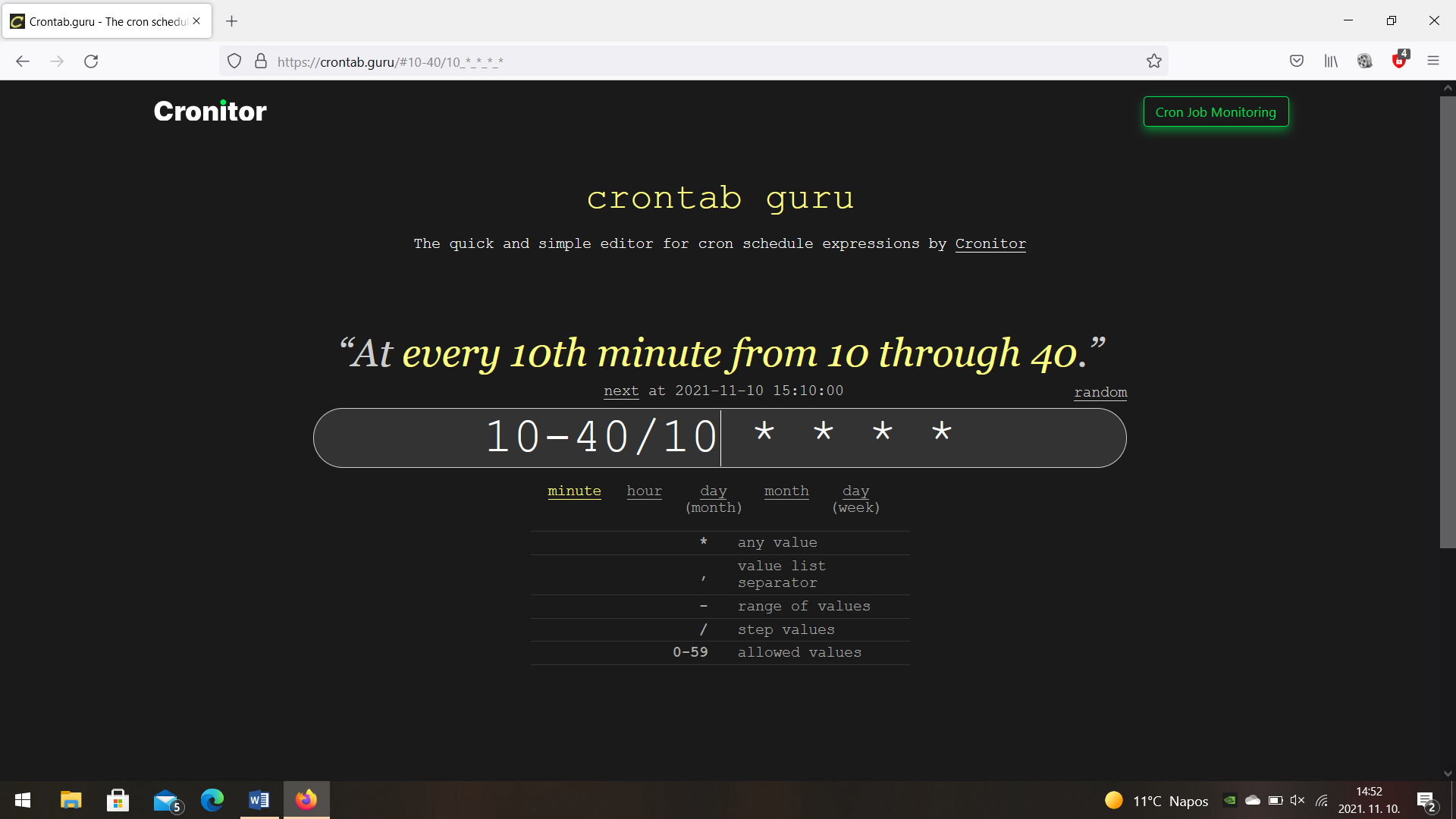
\*\*batch(nem időhöz fogja kötni , hanem a gép terheléséhez/van egy határ aminél nem fog lefutni)

\*ismételt időzítés: „cron” paranccsal

mindenkinek akinek van jogosultsága, azoknak van egy fiókjuk

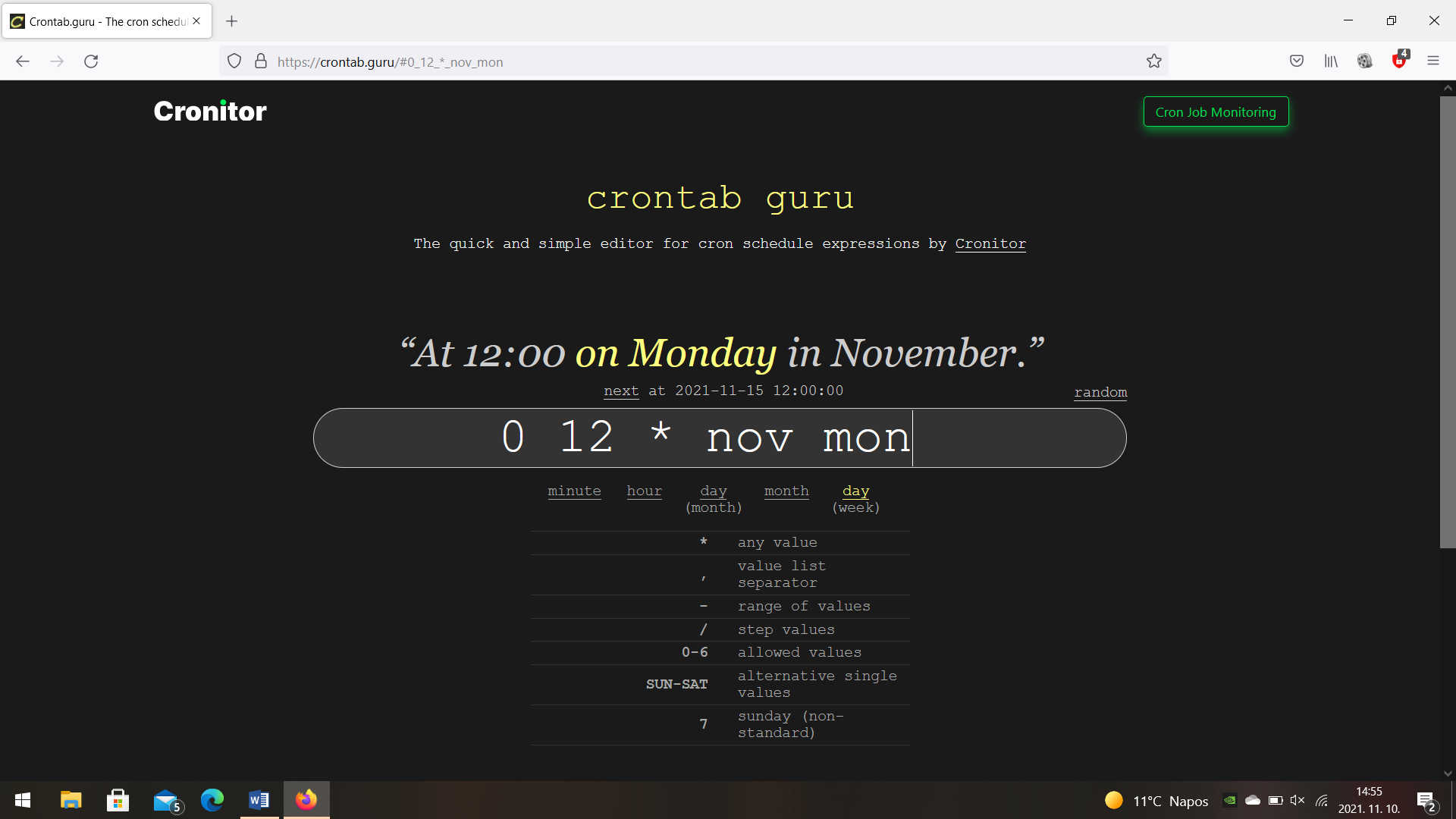
crontab –e (szerkeztés esetén)

pl:: minden óra 10 perc és negyven perc között 10 percenként fog lefutni



vagy pl

november minden hétfőjén délben fut le.



programon belül

\*\*\*\*\* echo hello>/dev/pts/0 -> ez minden percben kiírja hogy hello

grafikus helyzeteknél mindig ki kell írni egy grafikus felületre

Display=:0

programon belül

\*\*\*\*\* expor Display=:0 && firefox

parancs1 && parancs2

(lefut az első, ha az első hiba nélkül fut le akkor lefut a második is)

parancs1|| parancs2

(lefut az első parancs, ha az első hibás, akkor lefut a második)

Feladatok:

1. (MINDEN NAP ÉJFÉLKOR A HOME KÖNYVTÁRAMBÓL TÖRÖLJÜNK MINDEN FÁJLT AMINEK TMP VAGY TEMP A KITERJESZTÉSE)

0 0 1-31 1-12 0-6 rm\*.tmp \*.temp

vagy

00\*\*\* rm\*.tmp\*.temp

1. (Karácsonykor az asztalon jöjjön létre egy karácsony.txt fájl,amibe írjuk bele,hogy boldog karácsonyt)

002412\* touch karacsony.txt | echo „Boldog Karácsonyt”>Asztal/karacsony.txt

ez is elég

002412\* echo „Boldog Karácsonyt”>Asztal/karacsony.txt

1. ( Szilveszterkor megnyitja a böngészőt és betölt egy belő kiválasztott youtube videót)

0011\* export Display=:0 && firefox <https://youtube.com/(amit> szeretnék)

Feladatok:

1 feladat

Minden héten szerdán 15:00 kor a home könyvtárban levő sh és txt kiterjesztésű fájlokat tegye egy olyan fájlba aminek a neve „orai\_munka\_(dátum).tar.gz tömörített fájlba

\*\*tar –czf orai\_munka\_$(date+ %F).tar.gz

2 feladat

Minden órában írja ki a 10 legtöbb cpu időt használt processz nevét írja ki egy cpu\_(timestamp).txt fájlba

\*ps –aux| tail –n +2 | tr- s ’ ’| cut –d ’’ –f10,11| sort –n | tail

3. Feladat

crontab –e

\*0\*\*\*\* ps –aux |tail –n+2 | tr –s ’’ | cut –d ’’ –f10,11| sort –n | tail >cpu\_$(date +\%s.txt)

4 feladat

Írjuk ki annak a felhasználónak a nevét ,akinek a legtöbb futó processze van!

\*\*ps –aux | cut -d’’ –f1 | sort –r | uniq –c | sort –n | tr –s ’’ | cut –d ’’ –f3 | tail –n –1

../echo Már $(date %H) óra van / --🡪 (14:00 esetén)

Már 14 óra van.

Mire számíthatunk a zhban ?? :

(2.dik zh)

- szűrők

- jogok

- pipeline , átirányítás

- processz kezelés

-ps –aux

-top

-htop

-pstree

-kill használata

-prioritás kezelés (niceness): -nice,renice

./vegtelen.sh

Háttérben futás esetén

./vegtelen.sh & (a végére írt és jel a háttérbe futtatást eredményezi)

Archiválás :

+gzip(betömörítés) egyesével tömörít , az eredetit törli

+gunzip(kitömörít) egyesével

+tar /kapcsolók

* -cvf (újat létrehozni) ez egy új , még nem létező
* - cvzf(tömörít is)
* -tvf(felsorolja a tartalmat)
* -xvf(kicsomagolás)
* -xvzf(zipes dolgot csomagol ki)
* -rvf (hozzáfűzés ) létezőhöz tesz

időzített futtatás:

--egyszeri

at paranccsal ( megadunk egy időpontot és hogy mit akarunk akkor futtatni)

atq(kilistázza , hogy mik várnak most futtatásra)

atqm(ebből a listából tudunk törölni)

batch( at szerint műkődik, nem időponthoz hanem terheltséghet köti a futtatást)

--ismétlődő időzített futtatás

crontab

Az online felület -🡪 gyires.inf.unideb.hu/KMITT/b19

5-8 ig az anyagok

w (kiírja , hogy hány felhasználó van bejelentkezve

who(kik vannak bejelentkezve)

whoami(ki van bejelentkezve?)

echo $USER

id(mi a felhasználó user id, group id .. információk) ha megadunk egy felhasználót akkor arról fogja leírni az információkat

group(milyen csoportokban van a felhasználó

user( ad egy listát,hogy kik vannak bejelentkezve)

Adott valamilyen fájl, és abból mondjunk meg valami infót

cat fájl | stb.

feladatok:

Az aktuális mappában lévő fájlról adjon valamit

1. ls –l( vagy egyéb kapcsoló|..)

Ha a bejelentkezett felhasználókból indulunk

w |..

who|..

Írjuk ki az összes felhasználót aki .. etc/passwd ből

…| tail –n +2 | ..

Ha a fejlécet akarom levágni

minden oszlopot egy szóköz válasszon el

ls –l | tr –s ’’ | cut –d ’ ’ –f5

cat/etc/passwd(minden sor a felhasználók információit tartalmazza)

soronként 7 mező

felhasználó:jelszó:user id:group id: komment mező: home könyvtár: login shell

bash(emberi felhasználású)

Kik azok a felhasználók akiknek a loginshellje bash ?

cat etc/passwd/ | grep ’bash$’ | cut –d ’: -f1

\*crontab esetén a \*/2 az a két óránként ha második csillag esetén használjuk

Ha csak kill 2020 van akkor termet küld a kill vagyis kill term 2020 lesz!!!

cat pelda.txt | head –n 100| tail –n 10 (91-től 100-ig írja ki a sorokat)

melyik az aktuális könyvtárban található legnagyobb fájl ?

(cut oszlopokat vág)

sort- sorokat néz az ötös kapcsoló az ötödik sor szerint rendez

ls –l | sort –k 5 –n | tail –n 1

Az aktuális könyvtárban hány almappa van ?

Csak egy számot írjon ki

ls –l |grep ’^d’ | wc –l

Mik a mappák ? ( ls –l | tr –s ’ ’ | cut –d ’ ’ –f1)

Crontabos feladat

Minden héten péntek este 8-kor készítsen egy tanulnivalo.tar.gz tömörített fájlt,amiben legyen benne a home könyvtárban levő összes olyan fájl aminek a nevében benne van a jegyzet szó !

020\*\*5 tar .cvzf tanulnivalo.tar.gz\*jegyzet\*