

Студент (име и презиме):
Број индекса:
6. октобар 2025.

Факултет инжењерских наука
Практикум из рачунарских алата
Колоквијум

Задаци:

У свим задацима важи претпоставка да се наредбе/програми/скрипте извршавају у **bash** окружењу. Наредбе које извршавају корисници обавезно започињати са **\$**, а које извршава суперкорисник са **#**.

Задатак 1 – [по 2% за сваки тачан од десет одговора, односно 20% укупно]

[2%] — Приказати садржај датотеке `/proc/cpuinfo` користећи страничење (page by page приказ).

1 `$ less /proc/cpuinfo` решење

[2%] — Направити копију датотеке `important.txt` са називом `important_backup.txt`.

1 `$ cp important.txt important_backup.txt` решење

[2%] — Обрисати све датотеке у тренутном директоријуму које се завршавају са `.tmp`.

1 `$ rm *.tmp` решење

[2%] — Приказати садржај директоријума `/home` у детаљном формату, сортиран по величини.

1 `$ ls -lS /home` решење

[2%] — Извршити команду `ps aux` и филтрирати резултате тако да се приказују само процеси који садрже реч `python`.

1 `$ ps aux | grep python` решење

[2%] — Приказати последњи измењене датотеке у директоријуму `/var/log`.

1 `$ ls -lt /var/log` решење

[2%] — Приказати број редова, речи и карактера у датотеци `important_backup.txt`.

1 `$ wc important_backup.txt` решење

[2%] — Пронаћи све датотеке формата `pdf` унутар родитељског директоријума (посматрајући у односу на тренутни радни директоријум) и свих његових поддиректоријумима.

1 `$ find ../ -name "*.pdf" -type f` решење

[2%] — Направити директоријум `proba/dva/test` и сву хијерархију родитељских директоријума ако не постоје.

1 `$ mkdir -p proba/dva/test` решење

[2%] — Приказати тренутни радни директоријум.

1 `$ pwd` решење

Задатак 2 – [40%]

- [15%] — Написати **bash** скрипту која прима путању до директоријума као аргумент и генерише извештај о садржају тог директоријума. Потребно је да извешај садржи број датотека, број поддиректоријума и укупну величину свих датотека.

решење

```

1  #!/usr/bin/env bash
2  if [ $# -ne 1 ]; then
3      echo "Потребан је тачно један аргумент: путања до директоријума"
4      exit 1
5  fi
6  dir="$1"
7  if [ ! -d "$dir" ]; then
8      echo "Директоријум $dir не постоји"
9      exit 2
10 fi
11 # Број датотека (укључујући и у поддиректоријумима)
12 file_count=$(find "$dir" -type f | wc -l)
13 echo "Број датотека: $file_count"
14 # Број поддиректоријума (искључујући сам директоријум)
15 dir_count=$(find "$dir" -type d | wc -l)
16 dir_count=$((dir_count - 1))
17 echo "Број поддиректоријума: $dir_count"
18 # Укупна величина
19 total_size=$(du -sh "$dir" | cut -f1)
20 echo "Укупна величина: $total_size"

```

решење

- [25%] — Написати **bash** скрипту која прима путању до директоријума као аргумент и прави резервне копије свих **.txt** и **.sh** датотека унутар директоријума (потребно је разматрати и садржаје поддиректоријума). За **.txt** датотеке, потребно је направити две резервне копије: једну која на оригиналан назив додаје суфикс **_backup_DATUM** пре екстензије, где је **DATUM** тренутни датум у формату ГГГГ-ММ-ДД, и другу са додатком **.bak** иза оригиналног имена и екстензије. За **.sh** датотеке, направити резервну копију додавањем **.bak** на крај назива. Потребно је да скрипта испишује поруке о свакој направљеној копији и на крају укупан број новонасталих копија.

решење

```

1  #!/bin/bash
2  if [ $# -ne 1 ]; then
3      echo "Потребан је тачно један аргумент: путања до директоријума"
4      exit 1
5  fi
6  dir="$1"
7  if [ ! -d "$dir" ]; then
8      echo "Директоријум $dir не постоји"
9      exit 2
10 fi
11 datum=$(date +%Y-%m-%d)
12 count=0
13 files=$(find "$dir" -type f \( -name "*.txt" -o -name "*.sh" \))
14 for file in $files; do
15     ext="${file##*}"
16     if [ "$ext" = "txt" ]; then
17         base=$(basename "$file" .txt)
18         dirpart=$(dirname "$file")
19         backup_name="${base}_backup_${datum}.txt"
20         cp "$file" "$dirpart/$backup_name"
21         echo "Направљена копија: $dirpart/$backup_name"
22         ((count++))
23     cp "$file" "$file.bak"
24     echo "Направљена резервна копија: $file.bak"
25     ((count++))

```

```

26     elif [ "$ext" = "sh" ]; then
27         cp "$file" "$file.bak"
28         echo "Направљена резервна копија: $file.bak"
29         ((count++))
30     fi
31 done
32 if [ $count -eq 0 ]; then
33     echo "Нису пронађене .txt или .sh датотеке у директоријуму $dir"
34 else
35     echo "Укупно направљено $count резервних копија"
36 fi

```

решење

Задатак 3 – [20%]

[20%] — У директоријуму /библиотека налази се четири поддиректоријума књиге, часописи, дисертације и радови.

На пројекту раде 4 корисника: **милица**, **петар**, **јована** и **никола**. Направити наведене кориснике и подесити лозинку за сваког од њих. Поред наведених 4 корисника, на систему постоји и корисник **библиотекар** (администратор библиотеке).

Уредити дозволе на систему тако да буде задовољено:

- Сви корисници имају пуне привилегије над садржајем директоријума **радови**.
- Корисник **милица**, као главни истраживач има пуне привилегије над садржајем сва 4 директоријума.
- Корисник **петар** има право читања садржаја свих директоријума осим дисертације, али има право писања и извршавања само унутар директоријума **књиге**.
- Корисник **јована** има право читања садржаја свих директоријума осим дисертације, али има право писања и извршавања само унутар директоријума **часописи**.
- Корисник **никола** има право читања садржаја свих директоријума, али има право писања и извршавања само унутар директоријума **дисертације**.
- Власник свих директоријума је **библиотекар**.

Обезбедити да сви новонастали садржаји унутар поддиректоријума библиотеке припадају групи надлежној за одговарајући део библиотеке.

решење

```

1     # groupadd књиге
2     # groupadd часописи
3     # groupadd дисертације
4     # groupadd радови
5
6     # useradd милица
7     # usermod -a -G књиге,часописи,дисертације,радови милица
8     # passwd милица
9
10    # useradd петар
11    # usermod -a -G књиге,радови петар
12    # passwd петар
13
14    # useradd јована
15    # usermod -a -G часописи,радови јована
16    # passwd јована
17
18    # useradd никола
19    # usermod -a -G дисертације,радови никола
20    # passwd никола
21
22    # chown библиотекар:књиге /библиотека/књиге
23    # chown библиотекар:часописи /библиотека/часописи
24    # chown библиотекар:дисертације /библиотека/дисертације

```

```

25 # chown библиотекар:радови /библиотека/радови
26
27 # chmod 2777 /библиотека/радови
28 # chmod 2754 /библиотека/књиге
29 # chmod 2754 /библиотека/часописи
30 # chmod 2750 /библиотека/дисертације

```

решење _____

Задатак 4 – [по 2% за сваки тачан од десет одговора, односно 20% укупно]

[2%] — Иницијализовати нови git репозиторијум у тренутном директоријуму.

```

1 $ git init

```

решење _____

[2%] — Додати све датотеке из тренутног директоријума у индекс и направити почетни commit са поруком Почетни комит.

```

1 $ git add .
2 $ git commit -m "Почетни комит"

```

решење _____

[2%] — Направити нову грану под називом развој и пребацили се на њу.

```

1 $ git checkout -b развој

```

решење _____

[2%] — Направити нову датотеку README.md, додати је у индекс и направити commit са поруком Додао README.md.

```

1 $ touch README.md
2 $ git add README.md
3 $ git commit -m "Додао README.md"

```

решење _____

[2%] — Приказати статус репозиторијума и последњи commit на тренутној грани.

```

1 $ git status
2 $ git log -1

```

решење _____

[2%] — Приказати све постојеће гране у репозиторијуму, укључујући и удаљене гране.

```

1 $ git branch -a

```

решење _____

[2%] — Приказати разлике између тренутне гране и претходног commit.

```

1 $ git diff HEAD~1

```

решење _____

[2%] — Отказати последњи комит (задржавањем измена у индексу).

```

1 $ git reset --soft HEAD~1

```

решење _____

[2%] — Поново направити комит са задржаним изменама и поруком README.md - исправљена верзија.

```

1 $ git commit -m "README.md - исправљена верзија"

```

решење _____

[2%] — Вратити се на главну грану (master) и додати измене са гране развој.

```

1 $ git checkout master
2 $ git merge развој

```

решење _____