

National Technical University of Ukraine
“Igor Sikorsky Kyiv Polytechnic Institute”
Faculty of Informatics and Computer Science
Department of Information Systems and Technologies

LINUX

INSTRUCTIONS FOR PERFORMING LABORATORY WORKS FOR STUDENTS

Performed by:
student of group IM-14
Full name MEHMET KULUBECİOĞLU

Developed by: T
Maryna Khmeliuk

KYIV-2023

LABORATORY WORK #8

Topic

Data processing using the shell language. Scripts in Linux.

Task

scheduler in Linux

10. Write a script that reads the contents of the file passed as the first parameter

(in the file there is one integer in each line), checks the correctness of the format of

the contents of the file, if there are errors, it displays the messages and completes

the work, calculates the sum of all the numbers placed in the file, displays the received sum on the screen.

Aç

mehmet.sh

Kaydet

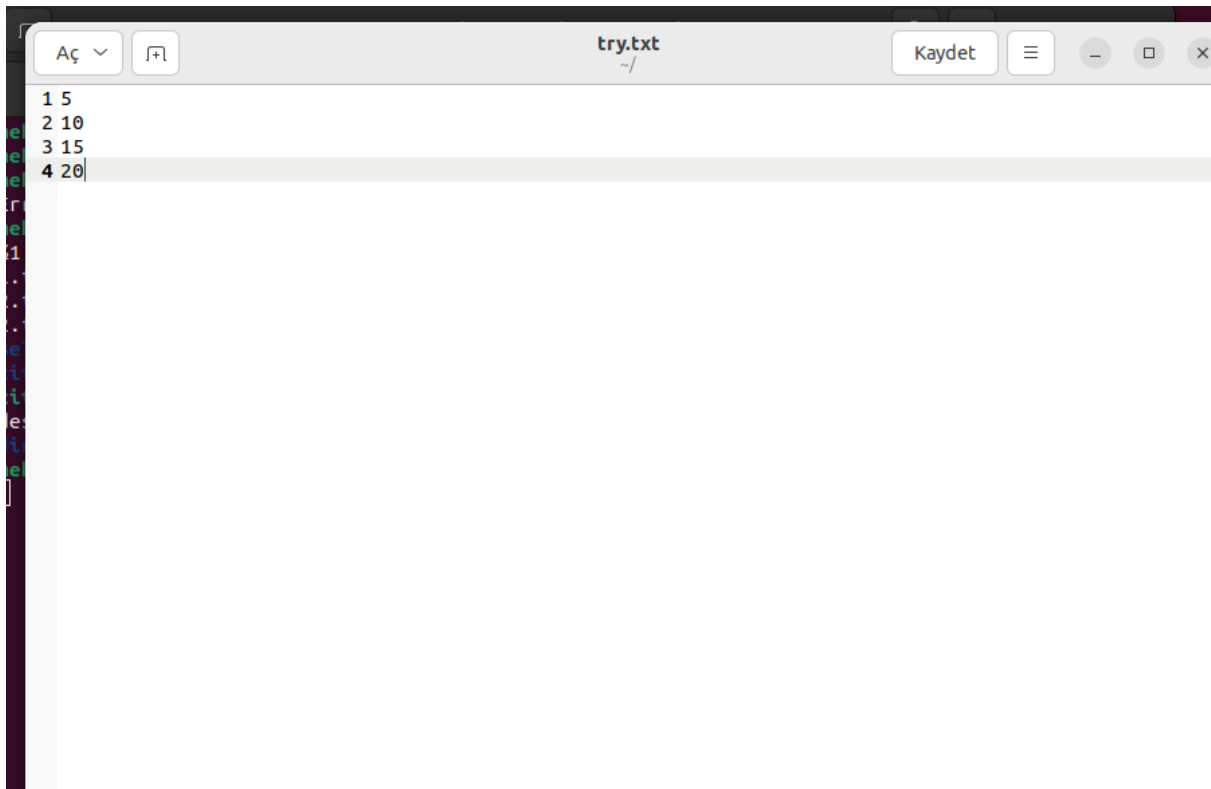
```
1 #!/bin/bash
2
3 # Check if a file is provided as a command-line argument
4 if [ "$#" -eq 0 ]; then
5     echo "Usage: $0 <filename>"
6     exit 1
7 fi
8
9 filename=$1
10
11 # Check if the file exists
12 if [ ! -e "$filename" ]; then
13     echo "Error: File '$filename' not found."
14     exit 1
15 fi
16
17 # Check if the file is empty
18 if [ ! -s "$filename" ]; then
19     echo "Error: File '$filename' is empty."
20     exit 1
21 fi
22
23 sum=0
24 error=false
25
26 # Read each line from the file and calculate the sum
27 while IFS= read -r line; do
28     # Check if the line contains a valid integer
29     if ! [[ "$line" =~ ^[0-9]+$ ]]; then
30         echo "Error: Invalid format in file. Line '$line' is not a valid integer."
31         error=true
32         break
33     fi
34
35     # Add the integer to the sum
36     sum=$((sum + line))
37 done < "$filename"
38
39 # Display the result
40 if [ "$error" = false ]; then
41     echo "Sum of numbers in file '$filename': $sum"
42 fi
43
```

sh

Girinti Genişliği: 8

Sat 7, Süt 3

ARY



```
mehmet@mehmet-VirtualBox:~$ ls
%1          directory      file2_sorted.txt  mehmet.sh      Resimler
1.txt       dosya             file2.txt         mehmet.txt     snap
2.tct       dosya1           Genel            Müzik          subdir
2.txt       dosya1_sorted.txt hellofile.txt     my_archive.tar.gz  Şablonlar
Belgeler    dosyaa           indirilenler     newdirectory    test.script.sh
city_1      example.txt      input.txt        newfile         test_script.sh
city_country.txt file             Masaüstü         new_pass.txt    try.txt
destination_file.txt file1_sorted.txt mehmet1.txt      new.txt         Videolar
dir1        file1.txt        mehmet2.txt      output.txt
```

```
mehmet@mehmet-VirtualBox:~$ chmod +x mehmet.sh
```

```
mehmet@mehmet-VirtualBox:~$ ./mehmet.sh try.txt
Sum of numbers in file 'try.txt': 50
mehmet@mehmet-VirtualBox:~$
```



```
mehmet@mehmet-VirtualBox: ~  
# Edit this file to introduce tasks to be run by cron.  
#  
# Each task to run has to be defined through a single line  
# indicating with different fields when the task will be run  
# and what command to run for the task  
#  
# To define the time you can provide concrete values for  
# minute (m), hour (h), day of month (dom), month (mon),  
# and day of week (dow) or use '*' in these fields (for 'any').  
#  
# Notice that tasks will be started based on the cron's system  
# daemon's notion of time and timezones.  
#  
# Output of the crontab jobs (including errors) is sent through  
# email to the user the crontab file belongs to (unless redirected).  
#  
# For example, you can run a backup of all your user accounts  
# at 5 a.m every week with:  
# 0 5 * * 1 tar -zcf /var/backups/home.tgz /home/  
#  
# For more information see the manual pages of crontab(5) and cron(8)  
#  
# m h dom mon dow   command  
  
~/tmp/crontab.8ClL6C/crontab" 23L, 889B
```