



Open your mind. LUT.

Lappeenranta University of Technology

Käyttöjärjestelmät ja systeemiohjelmointi

Project 2 UNIX SHELL

Lappeenrannan teknillinen yliopisto

Tietotekniikan koulutusohjelma

12.12.2018

CT30A3370 Käyttöjärjestelmät ja systeemiohjelmointi

Juho Kontiainen 0503209

Jesse Peltola 0523140

Sisällys

1. Johdanto.....	3
2. Komentotulkin toiminnallisuudet	3
2.1 Ohjelman toiminnasta.....	4
3. Ohjelman toimintojen esittelyä.....	4
4. Yhteenveto.....	6

1. Johdanto

Harjoitustyössä oli tarkoituksena luoda yksinkertainen komentotulkki -ohjelma (shell), johon käyttäjä pystyy syöttämään joko itse komentoja tai vaihtoehtoisesti komentojen suorittaminen käyttäjän valitsemasta tiedostosta.

Kaikki ohjelmakoodit löytyvät githubista: <https://github.com/JuhoKon/Project2>. Kaikki palautetut koodit kääntyvät optioilla -Wall ja -Werror, sekä toimivat kuten tässä dokumentaatiossa on selostettu.

2. Komentotulkin toiminnallisuudet

Tässä osiossa listataan ohjelman toiminnot sekä harjoitustyön kuvauksesta puuttuvat/vaillinaiset ominaisuudet.

Harjoitustyöstä löytyvät ominaisuudet:

- Interaktiivinen tila sekä batch-tila, jossa input luetaan tiedostosta. Batch-tilan tiedoston luvun jälkeen rakentamamme komentotulkki palaa interaktiiviseen tilaan.
- Luo uuden prosessin jokaiselle komennolle, toteutettu fork()-kutsulla.
- Ohjelmien ajaminen toteutettu execv-kutsulla.
- Kolme sisäänrakennettua funktiota; path, cd ja exit.
- Path asetettu `"/bin"`, kun ohjelma käynnistetään.
- cd -funktiolla pääsee liikkumaan hakemistoihin.
- Redirection toimii, jos tiedosto on jo olemassa, sen päälle kirjoitetaan.
- Erilaisia virheilmoituksia löytyy.

Harjoitustyöstä vaillinaisia/puuttuvia ominaisuuksia:

- Virheiden ohjaus tiedostoon ei ole toteutettu loppuun asti; ohjelma ei kirjoita kaikkia virheilmoituksia tiedostoon.
- Ohjelmalle voi antaa syötteenä useamman määrän `">"`-operaattoreita, eikä siitä tule virhettä. Myös operandin `">"` jälkeen voi olla useampi määrä argumentteja, ohjelma laskee vain ensimmäisen `">"` jälkeen olevan argumentin. Tämä johtuu siitä, miten toteutimme syötteen parsimisen.
- Rinnakkaiset komennot puuttuvat

2.1 Ohjelman toiminnasta

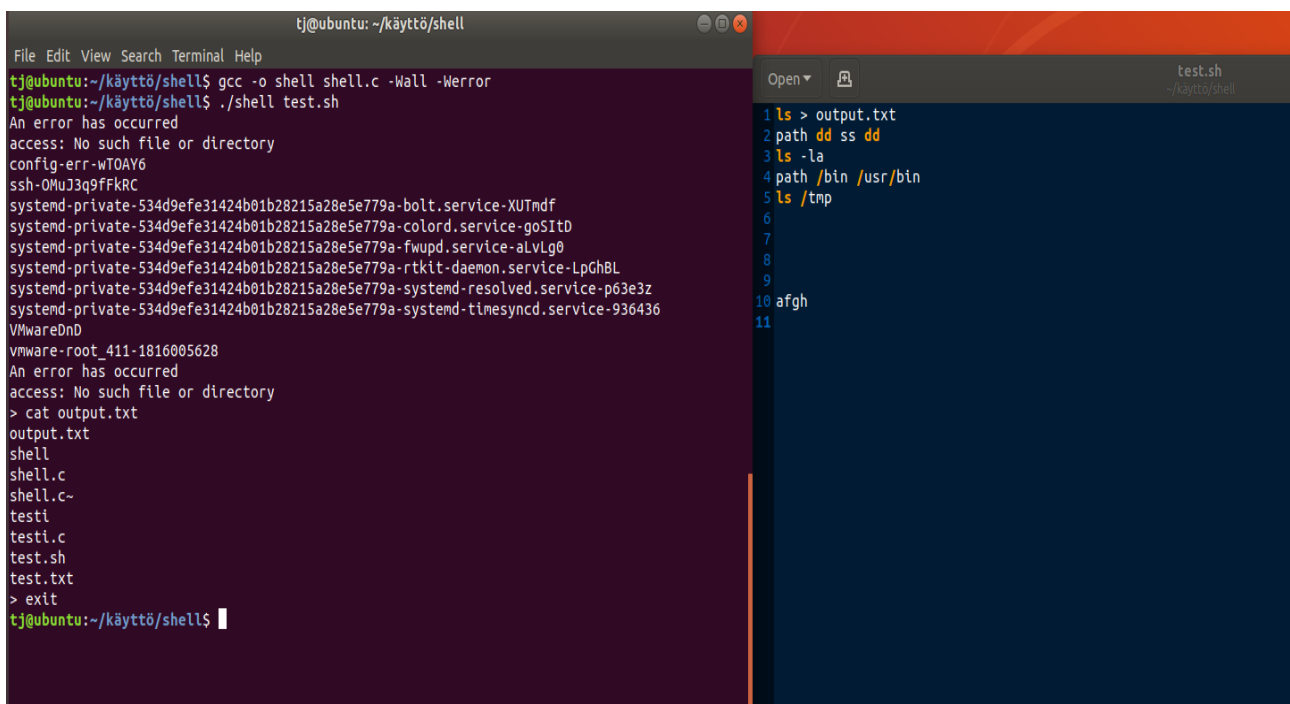
Ohjelma toimii siten, että main-funktiossa tarkistetaan, haluaako käyttäjä, että luetaan tiedostosta vai käsin syötettä, sen jälkeen kutsutaan loop-funktiota sen mukaan, onko meillä tiedosto, jota ajaa. Loop-funktiosta löytyy if-lauseke, jolla tarkistetaan onko tiedostoa jota lukea, jos on niin siirrytään batchluku-funktioon, muuten jatketaan ikilooppiin, josta poistutaan vasta kun käyttäjä antaa syötteeksi ”exit”.

Syötteen analysoinnin ja parsinnan toteutus on kahdessa funktiossa, parsefile sekä parse. Suoritukselle on kolme eri funktiota;

- presuoritus, jossa tarkistetaan tahdotaanko ajaa sisäänrakennettuja funktiota (exit,path tai cd) ja kutsutaan suoritus-funktiota jos tahdotaan suorittaa muita ohjelmia
- suoritus, jossa käydään path-lista läpi, ja tarkistetaan access()-kutsulla onko ohjelmaa olemassa, jos on, niin kutsutaan prosessi-funktiota
- prosessi, jossa itse ohjelman suoritus tapahtuu execv()-kutsulla

Ohjelman lähdekoodista käy ilmi tarkemmin, kuinka funktiot toimivat ja kommunikoivat keskenään. Myös lähteet ovat listattu lähdekoodiin.

3. Ohjelman toimintojen esittelyä



```
tj@ubuntu: ~/käyttö/shell
File Edit View Search Terminal Help
tj@ubuntu:~/käyttö/shell$ gcc -o shell shell.c -Wall -Werror
tj@ubuntu:~/käyttö/shell$ ./shell test.sh
An error has occurred
access: No such file or directory
config-err-wTOAY6
ssh-OMuJ3q9FFkRC
systemd-private-534d9efe31424b01b28215a28e5e779a-bolt.service-XUTndf
systemd-private-534d9efe31424b01b28215a28e5e779a-colord.service-goSItd
systemd-private-534d9efe31424b01b28215a28e5e779a-fwupd.service-aLVlg0
systemd-private-534d9efe31424b01b28215a28e5e779a-rtkit-daemon.service-LpGhBL
systemd-private-534d9efe31424b01b28215a28e5e779a-systemd-resolved.service-p63e3z
systemd-private-534d9efe31424b01b28215a28e5e779a-systemd-timesyncd.service-936436
VMwareDnD
vmware-root_411-1816005628
An error has occurred
access: No such file or directory
> cat output.txt
output.txt
shell
shell.c
shell.c-
testi
testi.c
test.sh
test.txt
> exit
tj@ubuntu:~/käyttö/shell$
```

Kuva 1

Kuvasta 1 (Kuva 1) nähdään, kuinka tiedostosta syötteiden luku toimii. Ensimmäisellä rivillä komento *ls* on ohjattu tiedostoon "output.txt". Seuravaksi *path* on vaihdettu siansaksaan, jolloin ei pitäisi pystyä suorittamaan mitään. Seuraavana on yritetty suorittaa komentoa *ls -la*, mutta komentoa ei löytynyt nykyisestä *path*:ista, minkä vuoksi virheilmoitus on tulostettu käyttäjälle. Sitten *path* on vaihdettu takaisin järkevään, ja seuraava komento *ls/tmp* toimiikin oikein. Viimeisenä on taas siansaksaa, jonka ohjelmamme tulkitsee virheeksi. Olen käsin syöttänyt komennon *cat output.txt* demonstroidakseni, että uudelleenohjaus toimii.

```
tj@ubuntu:~/käyttö/shell$ ./shell
> pwd
/home/tj/käyttö/shell
> ls
output.txt shell shell.c shell.c~ testi testi.c test.sh test.txt
> mkdir TestDir
> ls TestDir
> pwd
/home/tj/käyttö/shell
> cd TestDir
> pwd
/home/tj/käyttö/shell/TestDir
> path as lk kpo ih0 08y 7y h sctl j
> pwd
An error has occurred
access: No such file or directory
> path skjdf1 0j1f jk3 lkfj lksj klkl:___ /bin
> pwd
/home/tj/käyttö/shell/TestDir
> exit
tj@ubuntu:~/käyttö/shell$
```

Kuva 2 *cd*, *path* ja *exit*-funktioiden toimintaa.

```
tj@ubuntu:~/käyttö/shell$ ./shell
> cd a s d f
Too many arguments for command 'cd'.
> cd
Too few arguments for command 'cd'.
> exit
```

Kuva 3 *cd*-funktion virheilmoituksia.

```

tj@ubuntu:~/käyttö/shell$ ./shell
>
>
> ads      ff f
An error has occurred
access: No such file or directory
> l        s
An error has occurred
access: No such file or directory
> ls                                -la
total 84
drwxr-xr-x 3 tj tj  4096 Dec 11 16:43 .
drwxr-xr-x 4 tj tj  4096 Nov 29 17:04 ..
-rw-r--r-- 1 tj tj   65 Dec 11 16:28 output.txt
-rwxr-xr-x 1 tj tj 17896 Dec 11 16:43 shell
-rw-r--r-- 1 tj tj  9654 Dec 11 16:43 shell.c
-rw-r--r-- 1 tj tj  9642 Dec 11 16:11 shell.c~
drwxr-xr-x 2 tj tj  4096 Dec 11 16:40 TestDir
-rwxr-xr-x 1 tj tj  8304 Nov 29 14:23 testi
-rw-r--r-- 1 tj tj   89 Dec 11 16:13 testi.c
-rw-r--r-- 1 tj tj   74 Dec 11 16:27 test.sh
-rw-r--r-- 1 tj tj   74 Dec 11 16:25 test.txt

```

Kuva 4 Ohjelman toimintaa.

4. Yhteenveto

Ohjelman kaikki toiminnot ovat testattu omalla syötteellä sekä ajettu valgrind -muistidebuggaustyökalulla (komennolla `valgrind --leak-check=yes -v ./shell`). Muistivuotoja ei löytynyt ja ohjelma toimii oikein niiltä osin, mitä ominaisuuksia dokumentaatiossa on mainittu.