

Projekti 2: Unix Shell

wish-ohjelma

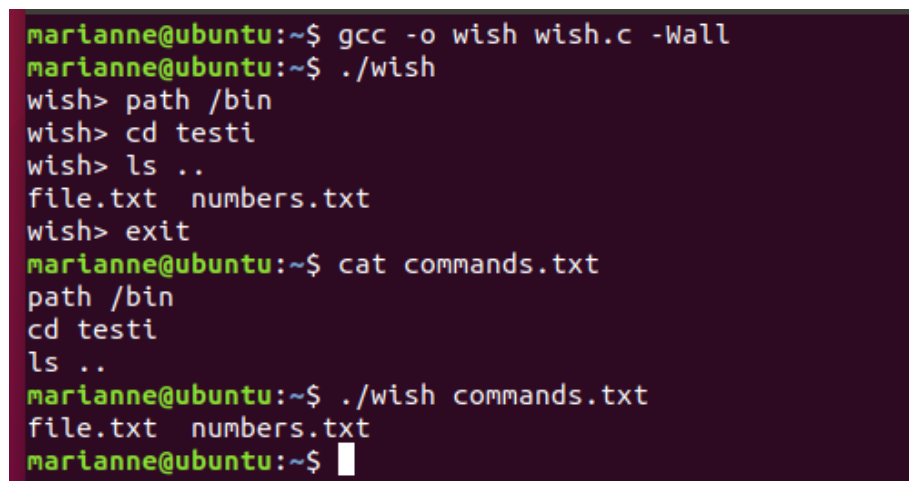
Ohjelman toiminta

Ohjelman tiedostonimi: wish.c

Ohjelman kääntäminen: **gcc -o wish wish.c -Wall**

./wish käskyllä ohjelma käynnistyy, ja pyytää käyttäjältä komentoja. Komennot toteutetaan ohjelman määritelmän mukaisesti: cd, path ja exit on toteutettu erikseen, muille luodaan prosessi ja ohjelma ajaa ne normaalisti. Toiminta näkyy kuvassa 1.

./wish commands.txt käskyllä ohjelma avaa parametrinä annetun tiedoston ja ottaa käskyt tiedostosta rivi kerrallaan. Kun tiedoston käskyt on käyty läpi, ohjelma loppuu. Toiminta näkyy kuvassa 1.



```
marianne@ubuntu:~$ gcc -o wish wish.c -Wall
marianne@ubuntu:~$ ./wish
wish> path /bin
wish> cd testi
wish> ls ..
file.txt  numbers.txt
wish> exit
marianne@ubuntu:~$ cat commands.txt
path /bin
cd testi
ls ..
marianne@ubuntu:~$ ./wish commands.txt
file.txt  numbers.txt
marianne@ubuntu:~$
```

Kuva 1. Ohjelman kääntäminen ja toiminta ilman parametreja, sekä yhdellä parametrilla.

Toteutetut käskyt

cd: Käsky ottaa yhden parametrin, ja vaihtaa käytössä olevan hakemiston parametrin mukaiseen. Jos parametrin mukaista hakemistoa ei löydy, ohjelma antaa virheilmoituksen ja pyytää uuden käskyn.

path: Ottaa yhden parametrin, ja määrittää shellin etsintä-polun parametrin mukaisesti. Oletuspolku on /bin.

exit: Lopettaa ohjelman ajon.

Muut käskyt: Ohjelma parsii käskyn, ja aloittaa lapsi-prosessin, jossa annettu käsky ajetaan. Käskyn ajon jälkeen ohjelma kysyy käyttäjältä uutta käskyä, tai ottaa uuden käskyn annetusta tiedostosta.

Redirection

Käskyjen uudelleenohjaus toimii, mutta ilman twistejä, eli output-käskyllä vastaus tulostuu ruudulle, ei mihinkään tiedostoon.

Parallel commands

Ohjelma ei tue tätä toimintoa, koska kehittäjä ei tajunnut tästä ominaisuudesta mitään.

Lähteet

C-ohjelmoinnin perusteiden ohjelmointiopasta, "C-kieli ja käytännön ohjelmointi osa 1, versio 2"

Fork() ja execv() käskyt:

<https://pages.cs.wisc.edu/~remzi/OSTEP/cpu-api.pdf>