

# Käyttöjärjestelmät ja systeemiohjelmointi

Olga Vinogradova-Lebedeva  
System Programming

16. kesäkuuta 2024

## 1 Project 1

### 1.0.1 Yleiskatsaus

Tämä ohjelma sisältää funktioita, jotka tarkistavat, ovatko kaksi tiedostoa linkitettyjä, sitten lukevat syötteen tiedostosta ja kirjoittavat sen toiseen tiedostoon tai standard-tulosteeseen sekä lukevat standard-syötteestä ja kirjoittavat standard-tulosteeseen. Ohjelman tarkoitus on kääntää tiedoston syöterivien järjestys.

## 1.1 Tiedostot ja Funktiot

### 1.1.1 filesarelinked

Tarkistaa, ovatko kaksi tiedostoa kovasti linkitettyjä (eli niillä on sama inode-numero ja laite-ID).

**Parametrit:**

- `const char *file1`: Ensimmäisen tiedoston polku.
- `const char *file2`: Toisen tiedoston polku.

**Palauttaa:**

- 1 jos tiedostot ovat linkitettyjä.
- 0 jos tiedostot eivät ole linkitettyjä.

### 1.1.2 readfilewritefile

Lukea rivit tiedostosta, kääntää niiden järjestys ja kirjoittaa ne toiseen tiedostoon.

**Parametrit:**

- `const char *filenamer`: Syöttötiedoston polku.
- `const char *filenamew`: Ulostulotiedoston polku.

### 1.1.3 readfilewriteoutput

Lukea rivit tiedostosta, kääntää niiden järjestys ja kirjoittaa ne standart-tulosteeseen.

**Parametrit:**

- `const char *filename`: Syöttötiedoston polku.

### 1.1.4 readstdinwritestdout

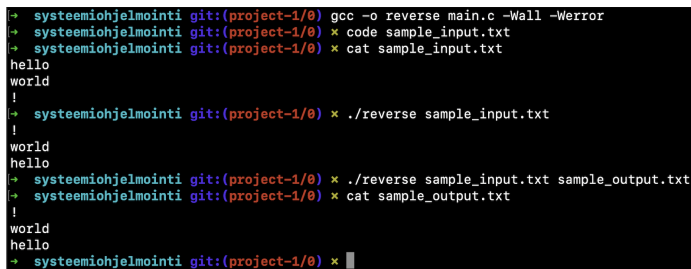
Lukea rivit standard-syötteestä, kääntää niiden järjestys ja kirjoittaa ne standard-tulosteeseen.

### 1.1.5 main

Parsia komentorivin argumentit ja määrittää, mitä funktiota kutsutaan argumenttien määrän perusteella.

**Parametrit:**

- `int argc`: Argumenttien määrä.
- `char *argv[]`: Argumenttivektori.



```
* systeemiohjelmointi git:(project-1/0) gcc -o reverse main.c -Wall -Werror
* systeemiohjelmointi git:(project-1/0) x code sample_input.txt
* systeemiohjelmointi git:(project-1/0) x cat sample_input.txt
hello
world
!
* systeemiohjelmointi git:(project-1/0) x ./reverse sample_input.txt
!
world
hello
* systeemiohjelmointi git:(project-1/0) x ./reverse sample_input.txt sample_output.txt
* systeemiohjelmointi git:(project-1/0) x cat sample_output.txt
!
world
hello
* systeemiohjelmointi git:(project-1/0) x
```

Kuva 1: Reverse käyttö

## 1.2 Kääntäminen ja Suorittaminen

Koodin kääntäminen:

```
gcc -o reverse main.c -Wall -Werror
```

Ohjelman suorittaminen:

- Ilman argumentteja (lukee standard-syötteestä, kirjoittaa standard-tulosteeseen):  
`./reverse`
- Yhdellä argumentilla (lukee tiedostosta, kirjoittaa standard-tulosteeseen):  
`./reverse input.txt`
- Kahdella argumentilla (lukee yhdestä tiedostosta, kirjoittaa toiseen):  
`./reverse input.txt output.txt`

### 1.3 Virheen käsittely

Ohjelma käsittelee virheitä tiedosto-operaatioiden, muistinvarauksen ja virheellisen käytön osalta. Se tulostaa asianmukaiset virheilmoitukset `stderr`-virtaan ja lopettaa tilakoodilla 1.

A terminal window with a black background and white text. The prompt is 'initial-reverse git:(master) \*'. The user has entered 'sh test-reverse.sh'. The output shows seven tests, all passing, listed as 'test 1: passed' through 'test 7: passed'. The prompt is now 'initial-reverse git:(master) \*' with a cursor.

```
→ initial-reverse git:(master) * sh test-reverse.sh
test 1: passed
test 2: passed
test 3: passed
test 4: passed
test 5: passed
test 6: passed
test 7: passed
→ initial-reverse git:(master) *
```

Kuva 2: Reverse testit

### 1.4 Testit

Ohjelma on läpäissyt kaikki Ostep-projektin testit (2)