

Jesse Mäkelä

jesse.j.makela@student.lut.fi

Project 1: Warmup to C and Unix programming

Features

- `./reverse` — read from stdin, write to stdout (exits using 0)
- `./reverse input.txt` — read from `input.txt`, write to stdout
- `./reverse input.txt output.txt` — read `input.txt`, write reversed lines to `output.txt`

Build

```Bash

```
gcc -std=c11 -Wall -o reverse reverse.c
```

## Usage example:

```
• essej@JessePC:~/kaysys$ cd ./projekti1/
• essej@JessePC:~/kaysys/projekti1$ gcc -std=c11 -Wall -o reverse reverse.c
• essej@JessePC:~/kaysys/projekti1$./reverse
hello
world
θ
world
hello
• essej@JessePC:~/kaysys/projekti1$./reverse input.txt
a file
is
this
hello
• essej@JessePC:~/kaysys/projekti1$./reverse input.txt output.txt
❖ essej@JessePC:~/kaysys/projekti1$
```

Input and output files:

| ≡ output.txt ×                                                         | ... | ≡ input.txt ×                                                    | ... |
|------------------------------------------------------------------------|-----|------------------------------------------------------------------|-----|
| projekti1 > ≡ output.txt<br>1 a file<br>2 is<br>3 this<br>4 hello<br>5 | ... | projekti1 > ≡ input.txt<br>1 hello<br>2 this<br>3 is<br>4 a file | ... |
|                                                                        |     |                                                                  |     |

# Project 2: Unix Utilities

Jesse Mäkelä

[jesse.j.makela@student.lut.fi](mailto:jesse.j.makela@student.lut.fi)

## Features

| Feature  | What does                                                                            | Usage example (bash)                                              |
|----------|--------------------------------------------------------------------------------------|-------------------------------------------------------------------|
| my-cat   | A simplified version of the `cat` utility that reads and displays file contents.     | ./my-cat test1.txt                                                |
| my-grep  | A simplified version of the `grep` utility that searches for text patterns in files. | ./my-grep foo test2.txt                                           |
| my-zip   | A file compression utility using run-length encoding.                                | ./my-zip test3.txt > compressed.z                                 |
| my-unzip | A file decompression utility for files compressed with my-zip.                       | ./my-unzip compressed.z<br>./my-unzip compressed.z > original.txt |

## Build

This project has Makefile

To build:

```
make
```

To clean:

```
make clean
```

Usage example:

```
• essej@JessePC:~/kaysys/projekti2$ make
 gcc -Wall -Werror -o my-cat my-cat.c
 gcc -Wall -Werror -o my-grep my-grep.c
 gcc -Wall -Werror -o my-zip my-zip.c
 gcc -Wall -Werror -o my-unzip my-unzip.c
• essej@JessePC:~/kaysys/projekti2$./my-cat test1.txt
Hello World
This is a test file
It has multiple lines
• essej@JessePC:~/kaysys/projekti2$./my-grep foo test2.txt
foo bar
This line has foo in it
foolish test
• essej@JessePC:~/kaysys/projekti2$./my-zip test3.txt > compressed.z
• essej@JessePC:~/kaysys/projekti2$./my-unzip compressed.z
aaaaaaaaabbbb
• essej@JessePC:~/kaysys/projekti2$./my-unzip compressed.z > original.txt
• essej@JessePC:~/kaysys/projekti2$ make clean
 rm -f my-cat my-grep my-zip my-unzip *.z *.o
✧ essej@JessePC:~/kaysys/projekti2$
```

| ≡ test1.txt X                                                                                     | ... | ≡ test2.txt X                                                                                                               |
|---------------------------------------------------------------------------------------------------|-----|-----------------------------------------------------------------------------------------------------------------------------|
| projekti2 > ≡ test1.txt<br>1 Hello World<br>2 This is a test file<br>3 It has multiple lines<br>4 |     | projekti2 > ≡ test2.txt<br>1 foo bar<br>2 This line has foo in it<br>3 Another line without the word<br>4 foolish test<br>5 |

| ≡ test3.txt X                                    | ... | ≡ original.txt U X                                  |
|--------------------------------------------------|-----|-----------------------------------------------------|
| projekti2 > ≡ test3.txt<br>1 aaaaaaaaaabbbb<br>2 |     | projekti2 > ≡ original.txt<br>1 aaaaaaaaaabbbb<br>2 |

# Project 3: Unix Shell

This project implements a simple Unix shell called 'wish' in C. The shell supports interactive and batch modes for command execution, built-in commands, external command execution, output redirection, parallel command execution, and detailed error handling.

## Build

This project has Makefile

To build:

```
make
```

To clean:

```
make clean
```

## Features Implemented

| Command                | What does                                        |
|------------------------|--------------------------------------------------|
| exit                   | Exit the shell (no arguments allowed)            |
| cd <directory>         | Change directory (exactly one argument required) |
| path <dir1> <dir2> ... | Change directory (exactly one argument required) |

## Advanced Features

Output redirection using '>'

Parallel execution using '&'

## Usage

Interactive mode:

```
./wish
```

```
wish> ls
wish> echo Hello World
wish> exit
```

Batch mode:

```
./wish commands.txt
```

## Usage examples

### Interactive Mode & Basic Commands

```
● essej@JessePC:~/kaysys/projekti3$./wish
wish> ls
Makefile README.md unix_shell.c wish
wish> echo hello from shell
hello from shell
wish> pwd
/home/essej/kaysys/projekti3
wish> exit
```

### Built-in Commands

```
● essej@JessePC:~/kaysys/projekti3$./wish
wish> path /bin /usr/bin
wish> cd /tmp
wish> pwd
/tmp
wish> cd /home/essej/kaysys/projekti3
wish> pwd
/home/essej/kaysys/projekti3
wish> exit
```

## Output Redirection

```
● essej@JessePC:~/kaysys/projekti3$./wish
wish> echo This is redirected output > test.txt
wish> cat test.txt
This is redirected output
[
wish> ls -la > listing.txt
wish> cat listing.txt
total 52
drwxr-xr-x 2 essej essej 4096 Dec 15 16:37 .
drwxr-xr-x 6 essej essej 4096 Dec 14 12:10 ..
-rw-r--r-- 1 essej essej 407 Dec 9 10:59 Makefile
-rw-r--r-- 1 essej essej 2078 Dec 15 16:16 README.md
-rw-r--r-- 1 essej essej 0 Dec 15 16:37 listing.txt
-rw-r--r-- 1 essej essej 26 Dec 15 16:37 test.txt
-rw-r--r-- 1 essej essej 8931 Dec 15 16:06 unix_shell.c
-rwxr-xr-x 1 essej essej 17864 Dec 15 16:32 wish
wish> exit
```

## Parallel Commands

```
esseej@JessePC:~/kaysys/projekti3$./wish
wish> echo Task 1 & echo Task 2 & echo Task 3
Task 1
Task 2
Task 3
wish> ls & pwd & echo Done
/home/esseej/kaysys/projekti3
Done
Makefile README.md listing.txt test.txt unix_shell.c wish
wish> exit
```