# Lab 4 — Input & Output Redirection

#include <stdio.h>

#include <unistd.h>

#include <stdlib.h>

#include <string.h>

#include <errno.h>

#include <sys/wait.h>

#include <sys/types.h>

#include <sys/stat.h>

#include <fcntl.h>

#include <stdio.h>

#include <errno.h>

extern char \*\*environ;

int main(int argc, char \*\*argv) {

int ret;

int status;

pid\_t pid = fork();

if(pid < 0) {

printf("Fork error: %s\n",strerror(errno));

exit(1);

printf("Wait: %d\n", wait(&status));

} else {

//open input for reading

int fin = open(argv[1], O\_RDONLY | O\_CREAT, 0777);

if (fin < 0) {

printf("Can't open input file: %s\n",strerror(errno));

exit(1);

}

dup2(fin, 0);

close(fin);

//open output for writing

int fout = open(argv[2], O\_WRONLY | O\_CREAT | O\_TRUNC, 0777);

if (fout < 0) {

printf("Can't open output file: %s\n",strerror(errno));

exit(1);

}

dup2(fout, 1);

close(fout);

ret = execve("copy", argv, environ);

if (ret < 0) {

printf("Execve failed: %s\n", strerror(errno));

exit(1);

}

}

wait(NULL);

exit(0);

}

# copy.c

#include <sys/types.h>

#include <sys/stat.h>

#include <fcntl.h>

#include <stdio.h>

#include <errno.h>

#include <string.h>

#include <stdlib.h>

#include <unistd.h>

int main(int argc, char \*\*argv) {

int fin = 0;

int fout = 1;

int n = 1;

char buffer[512];

int ret;

if(argc != 3) {

printf("Usage: lab1 infile outfile\n");

exit(1);

}

//get the process started

while(n > 0) {

n = read(fin, buffer, 512);

if (n < 0) {

printf("Error on read: %s\n",strerror(errno));

exit(1);

}

ret = write(fout, buffer, n);

if (ret < 0) {

printf("Erroc on write: %s\n",strerror(errno));

exit(1);

}

}

close(fin);

close(fout);

exit(0);

}

# Makefile

CFLAGS = -Wall -g

.PHONY: all clean

all: copy lab4

copy: copy.o

cc -o copy copy.o

lab4: lab4.o

cc -o lab4 lab4.o

clean:

rm copy copy.o lab4 lab4.o

# Output

