#include <string.h>

#include <unistd.h>

#include <sys/wait.h>

void ft\_putchar\_fd2(char c)

{

write(2, &c, 1);

}

void ft\_putstr\_fd2(char \*str)

{

while (\*str)

ft\_putchar\_fd2(\*str++);

}

int ft\_puterror(char \*str, char \*arg)

{

ft\_putstr\_fd2(str);

if (arg)

ft\_putstr\_fd2(arg);

ft\_putchar\_fd2('\n');

return (1);

}

int ft\_exec(char \*\*argv, char \*\*env, int i, int tempfd)

{

argv[i] = NULL;

dup2(tempfd, STDIN\_FILENO);

close (tempfd);

execve(argv[0], argv, env);

return (ft\_puterror("error: cannot execute ", argv[0]));

}

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

{

int tempfd;

int i;

int fd[2];

pid\_t pid;

(void)argc;

tempfd = dup(STDIN\_FILENO);

i = 0;

while (argv[i] && argv[i + 1])

{

argv = &argv[i + 1];

i = 0;

while (argv[i] && strcmp(argv[i], ";") && strcmp(argv[i], "|"))

i++;

if (strcmp(argv[0], "cd") == 0)

{

if (i != 2)

ft\_puterror("error: cd: bad arguments", NULL);

else if (chdir(argv[1]) != 0)

ft\_puterror("error: cd: cannot change directory to ", argv[1]);

}

else if (i != 0 && (argv[i] == NULL || strcmp(argv[i], ";") == 0))

{

pid = fork();

if (pid == -1)

ft\_puterror("error: fatal", NULL);

else if (pid == 0)

{

if (ft\_exec(argv, env, i, tempfd) == 1)

return (1);

}

else

{

close (tempfd);

while (waitpid(-1, NULL, WUNTRACED) != -1)

;

tempfd = dup(STDIN\_FILENO);

}

}

else if (i != 0 && strcmp(argv[i], "|") == 0)

{

if (pipe(fd) == -1)

ft\_puterror("error: fatal", NULL);

pid = fork();

if (pid == -1)

ft\_puterror("error: fatal", NULL);

else if (pid == 0)

{

dup2(fd[1], STDOUT\_FILENO);

close (fd[0]);

close (fd[1]);

if (ft\_exec(argv, env, i, tempfd) == 1)

return (1);

}

else

{

close (tempfd);

close(fd[1]);

tempfd = fd[0];

}

}

}

close (tempfd);

return (0);

}