Solution to (a):

CHILD_1   PARENT   CHILD_2   CHILD   PARENT   GRANDCHILD

PARENT column:
pipe(fds[])
fork()
fork()
wait()
wait()
pid(child_1)
pid(child_2)
finish()

CHILD_1 column:
close(fds[0])
close(1)
dup(fds[1])
exec("ls -l")
finish()

CHILD_2 column:
close(fds[A])
close(D)
dup(fds[0])
exec("WC")
finish()

CHILD column:
pipe(fd1)
pipe(fd2)
fork()
close(fd2[1])
pidchild
wait
read(fd2, pid)
pid(child)
pid(grandchild)
finish()

PARENT column (2nd):
fork()
close(fd2[0])
close(fd2[1])
close(fd[0])
close(1)
dup(fd[1])
exec("ls-l")
finish

GRANDCHILD column:
close(fd[0])
pid = getpid()
write(fd2, pid)
close(fd[1]
close(0)
dup(fd[0])
exec("WC")
finish
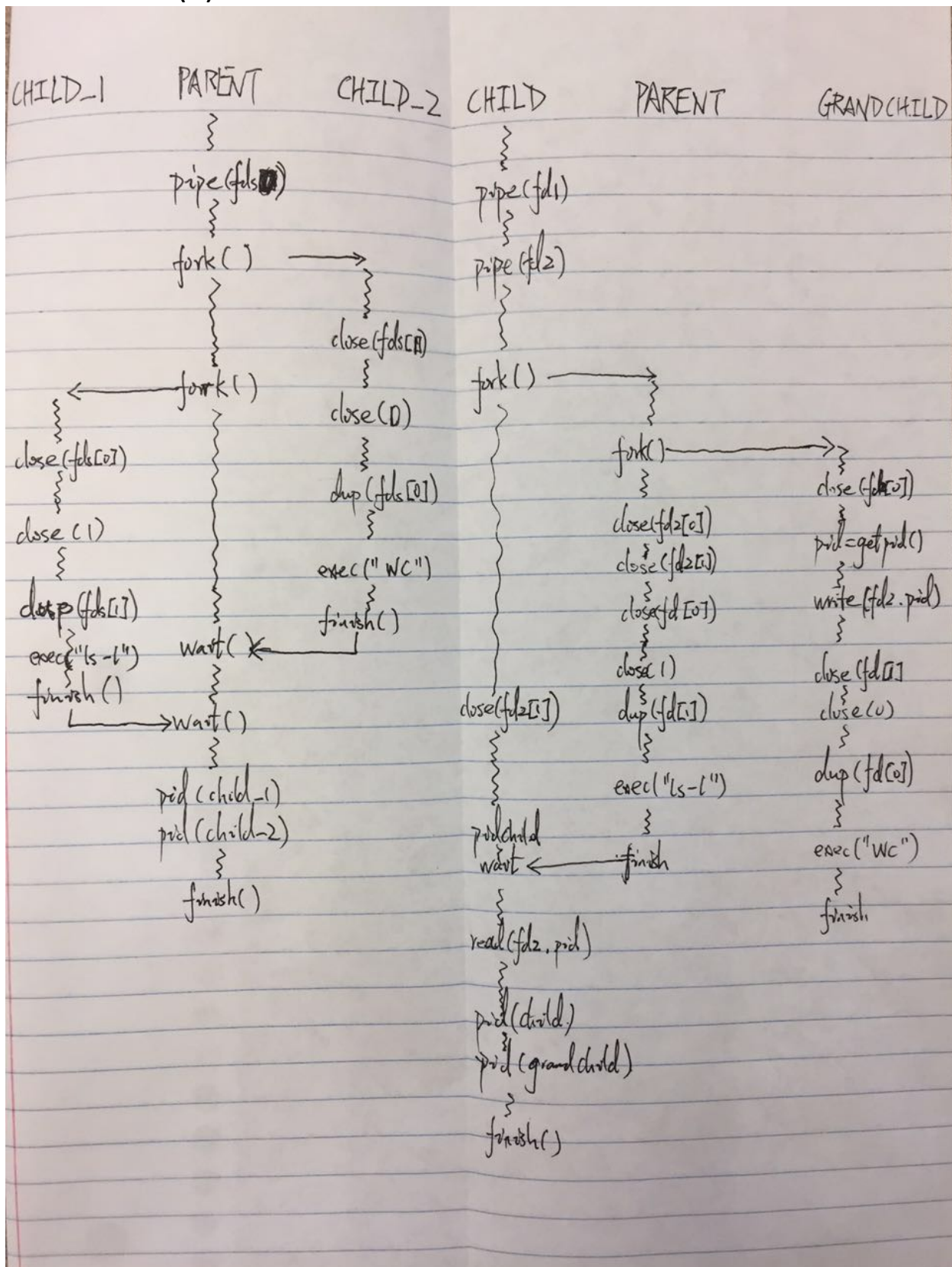
Solution to (b):

Option 1 is preferable, see reasons below,
In option 1, we only need 1 pipe for the communication between processes
child_1 and child_2, although syscall wait() needs to be called twice.
However, in option 2, 2 pipes are need. And another operation to pass the result
to parent process is also more complicated than option 1.

Solution to (c):

See in q3.c