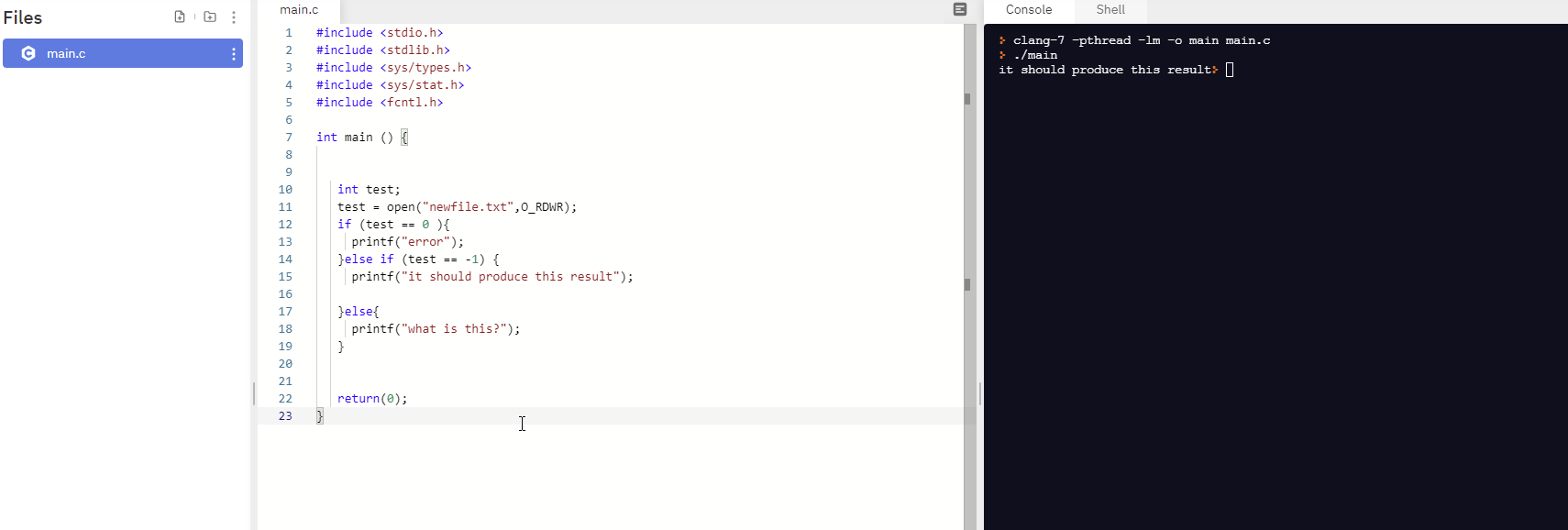
1. **perror()**

a subroutine that describes the system call errors. Displays the last system call error.



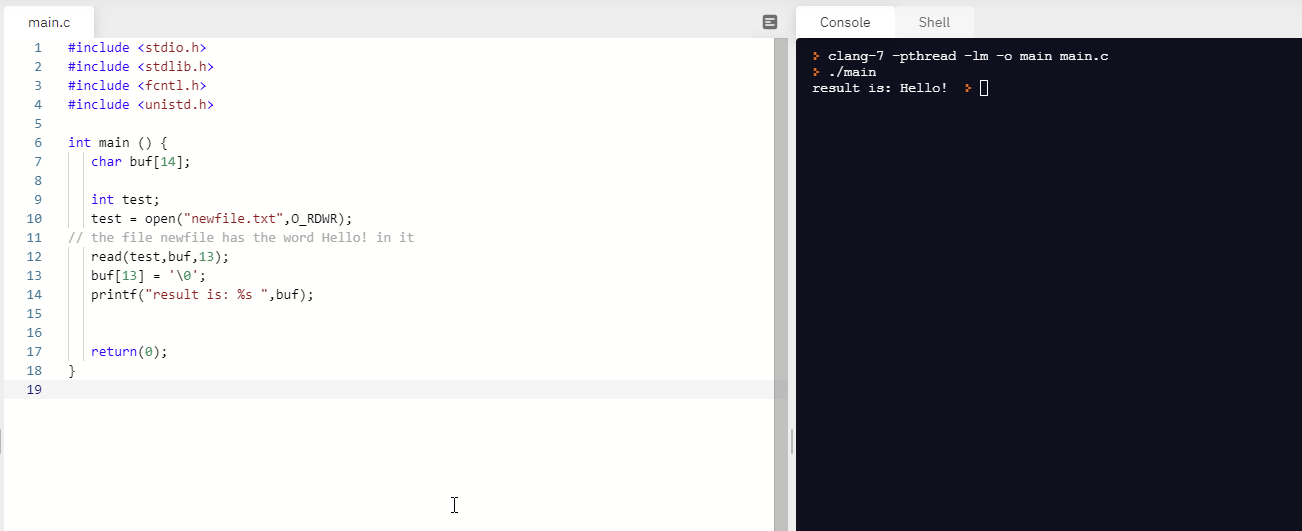
1. **open()**

opens or creates a file.



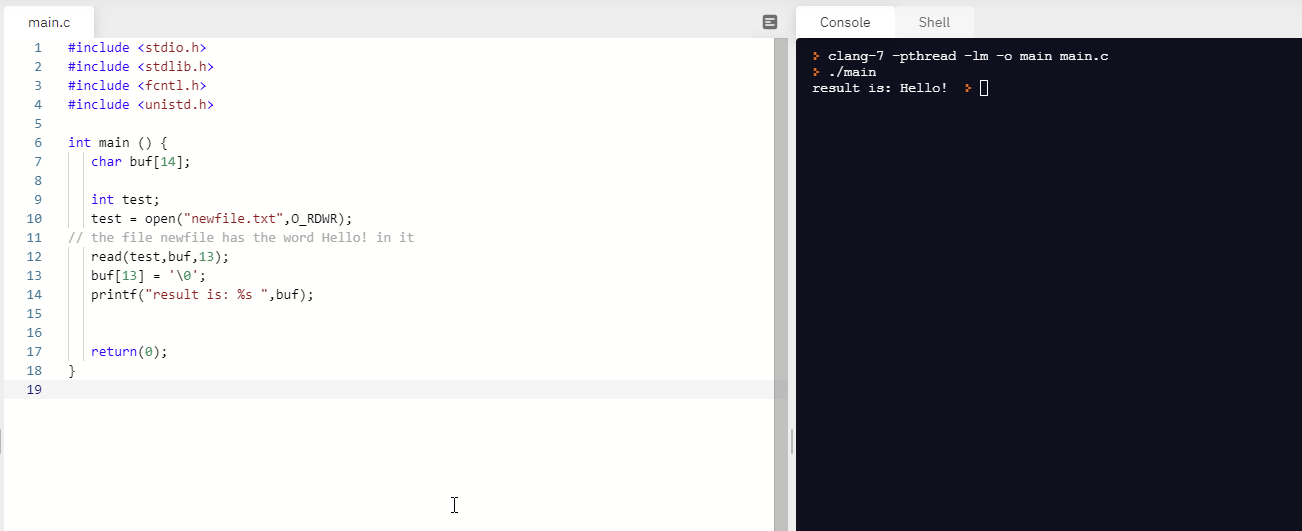
1. **read()**

reads the byte representation of a file into a buffer.

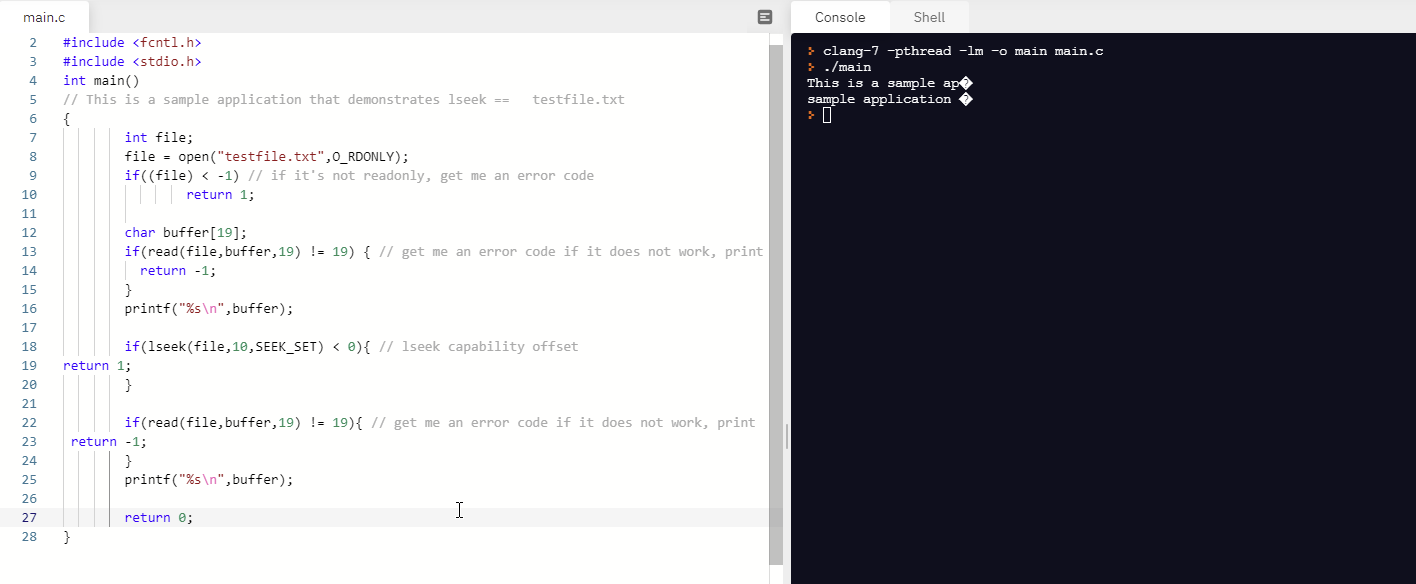


1. **write()**

writes byte from a buffer to the file.

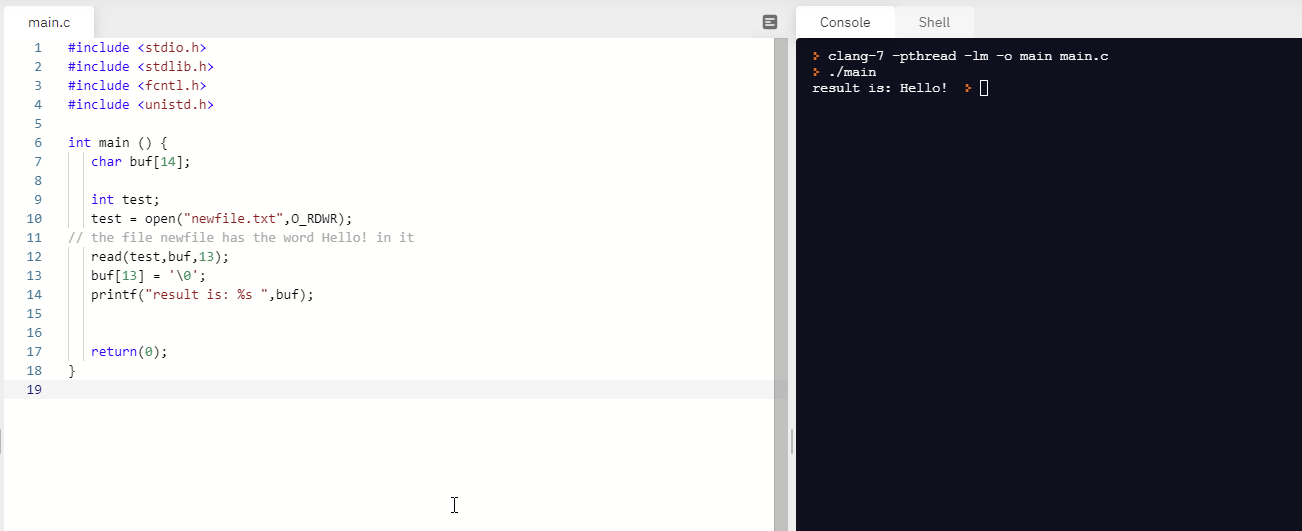


1. **lseek()**

moves a offset that is particular into a file 

1. **close()**

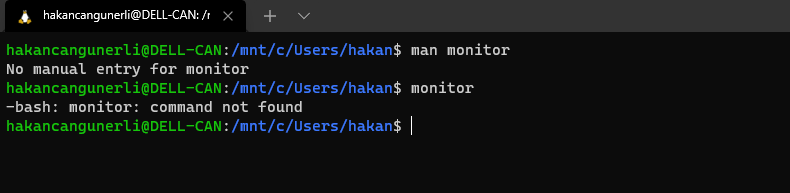
close the file.



1. **monitor()**

scans the files specified, displays information about the modification of those files.

// THIS DOES NOT EXIST WITHIN MY SHELL OR GNU C



1. **chown()**

changing the owner status or the group status of a file

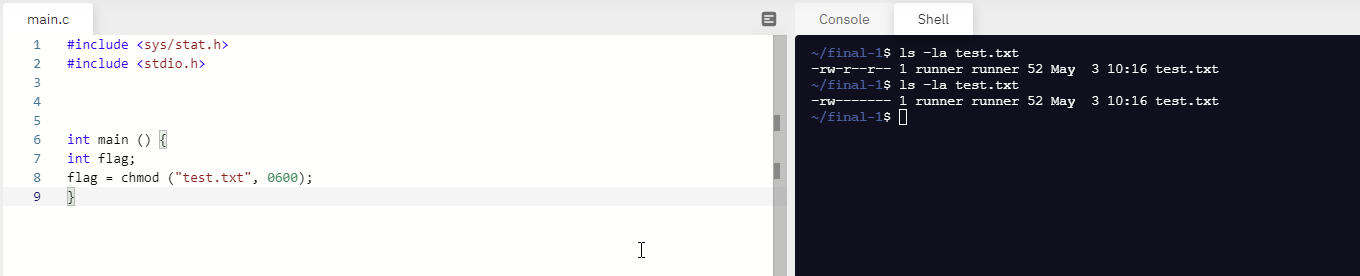


1. **fchown()**

pretty much like chown, except that it needs to be the file descriptor definition as supposed to the path.

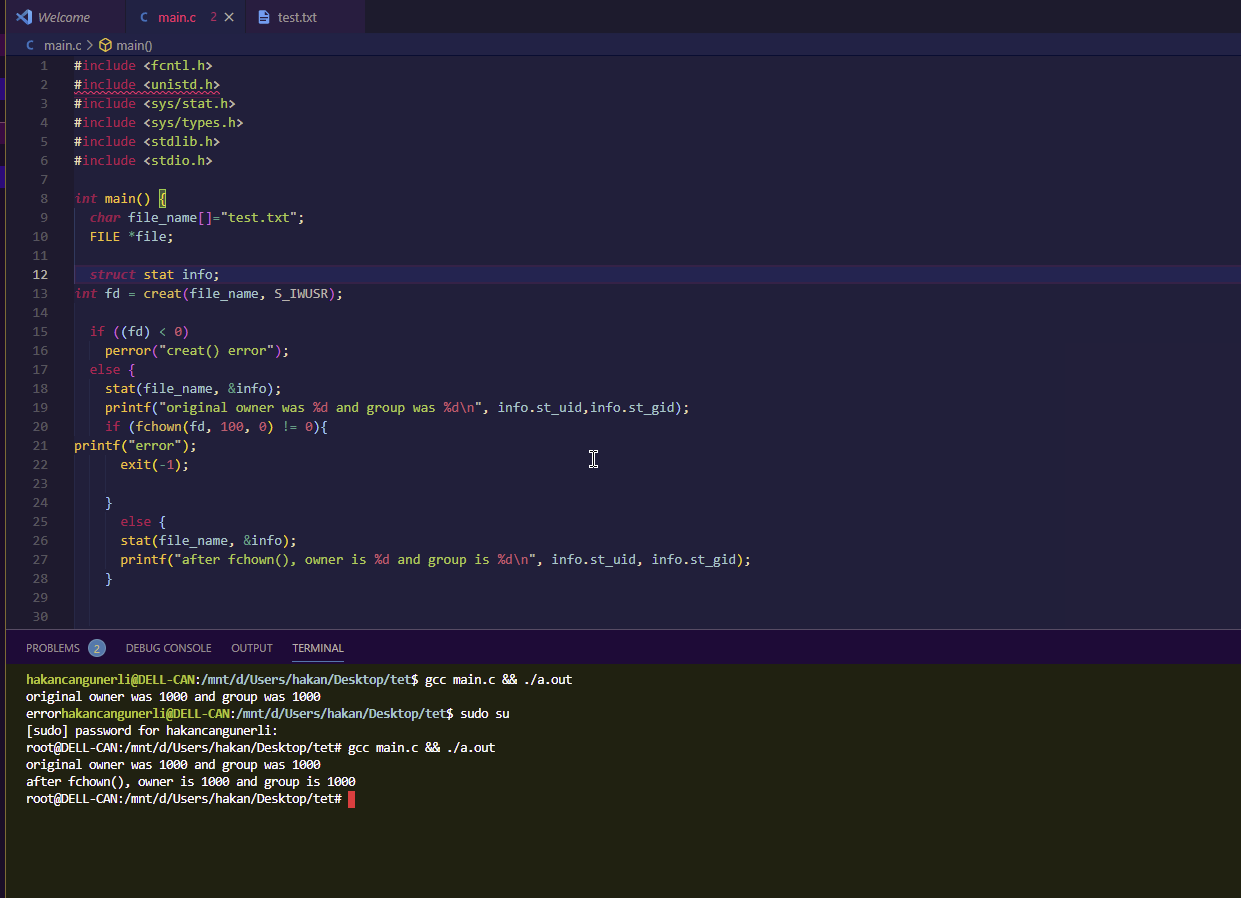
1. **chmod()**

changing the access permissions of a file

****

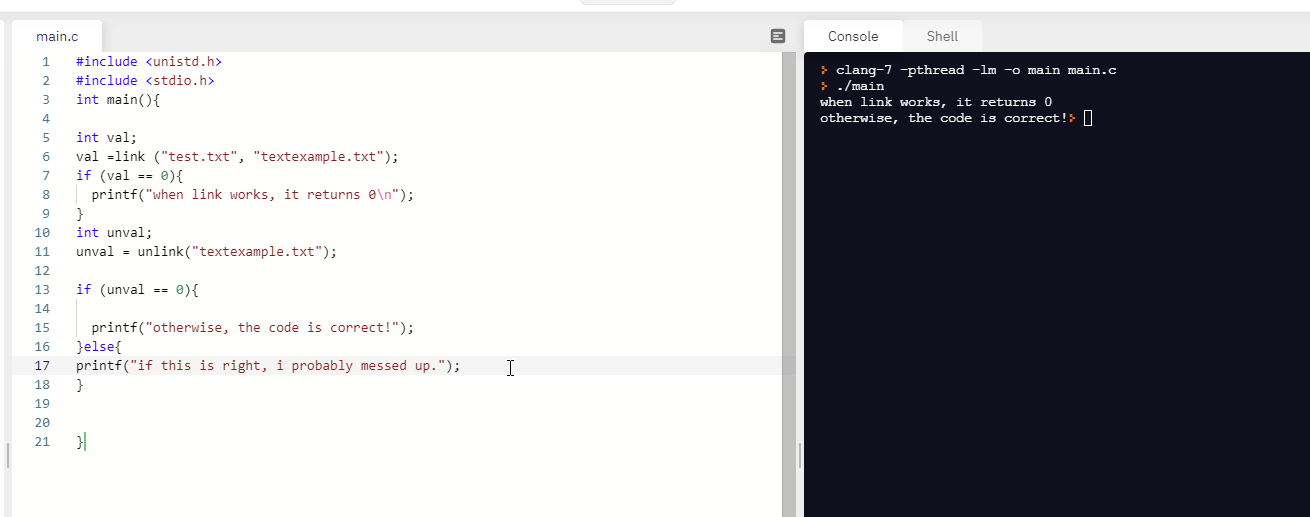
1. **fchmod()**

pretty much like chmod, except that it needs to be the file descriptor definition as supposed to the path.

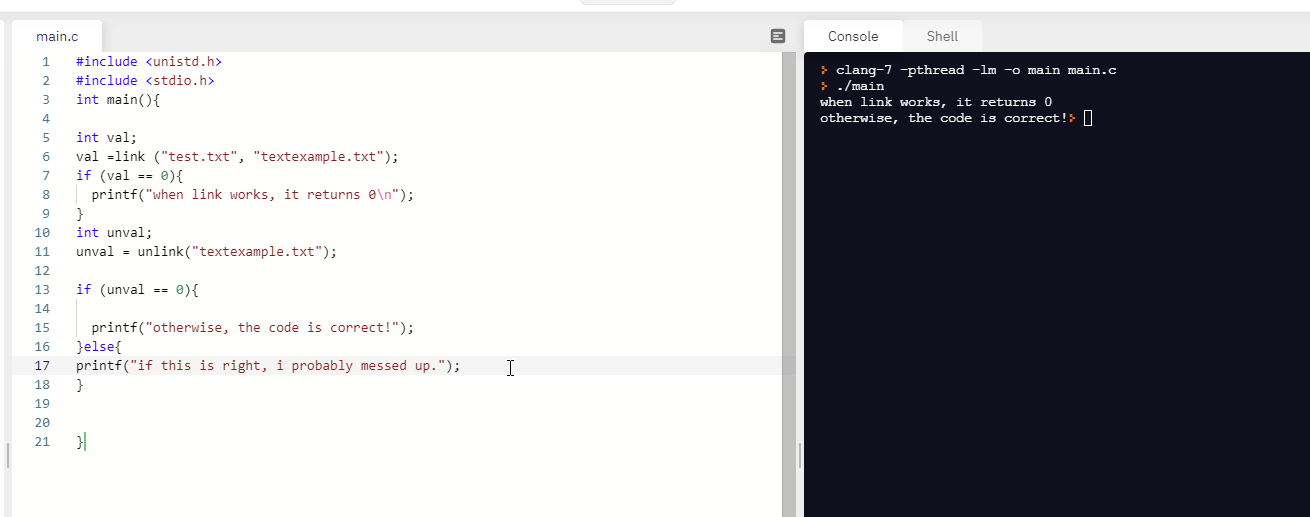


1. **link()**

creating a link between files.

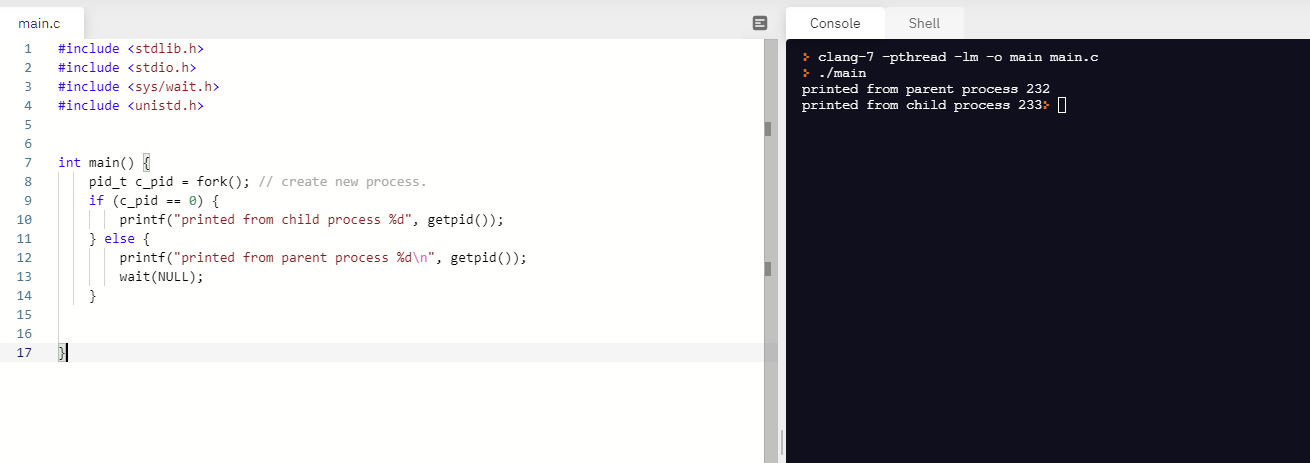


1. **unlink()**

killing the link between files.  


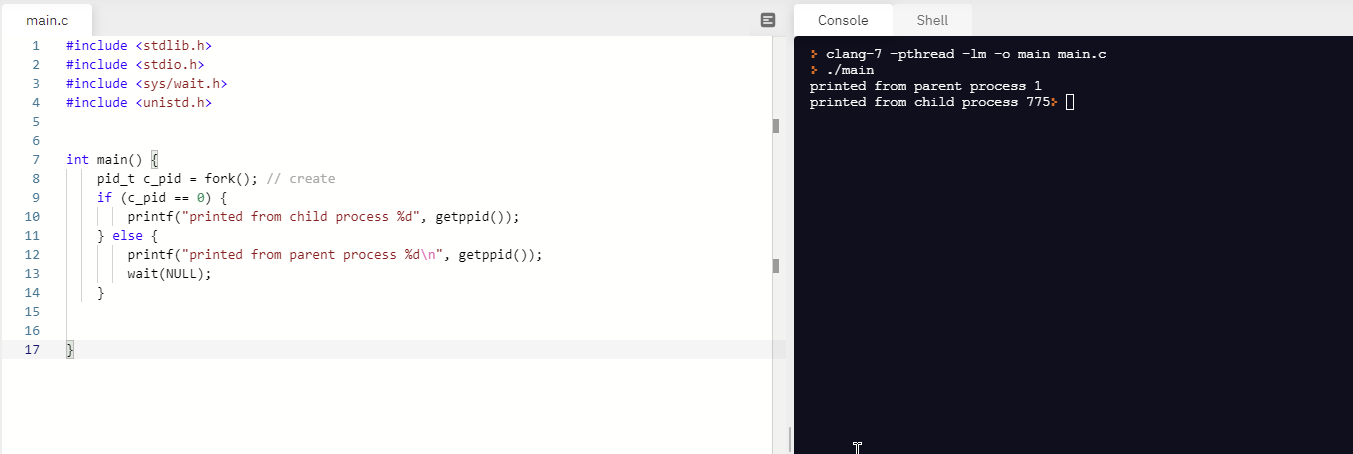
1. **getpid()**

get the id of a process. This is for calling function.



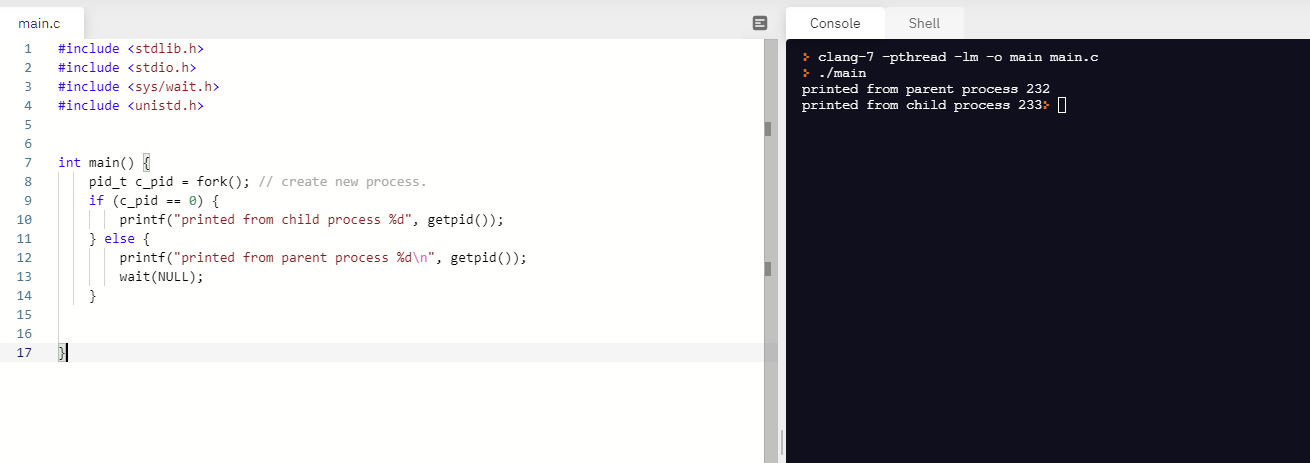
1. **getppid()**

return the process id of the parent.



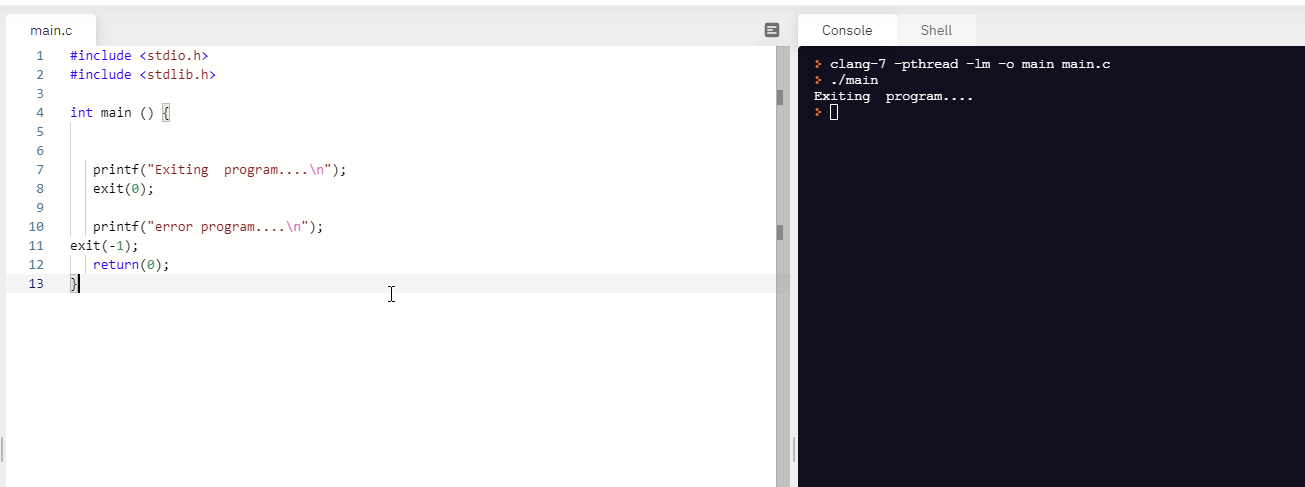
**p. fork()**

calls a process child. Duplicates the calling process

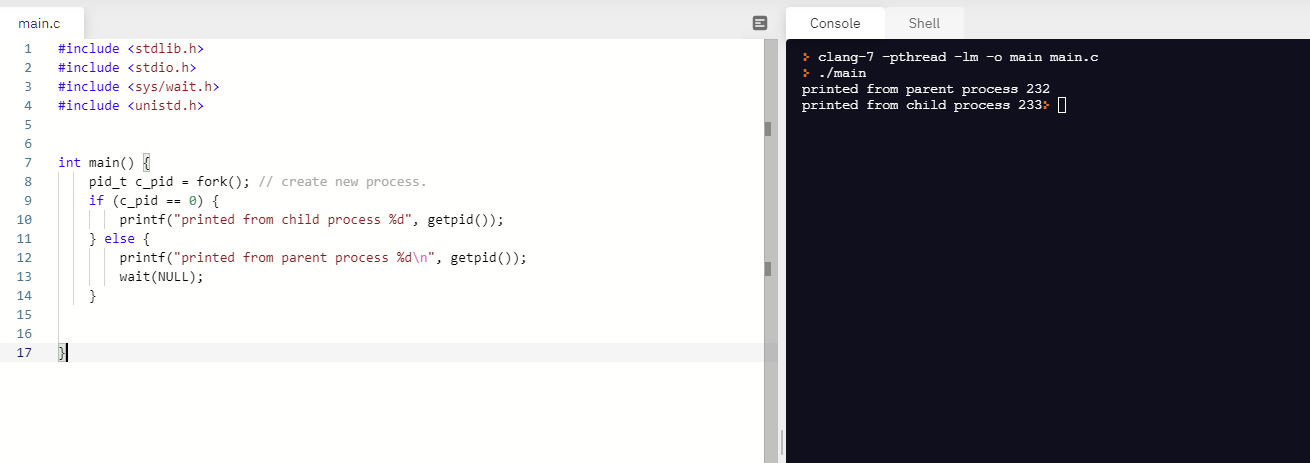


1. **exit()**

exit. Terminate the process.

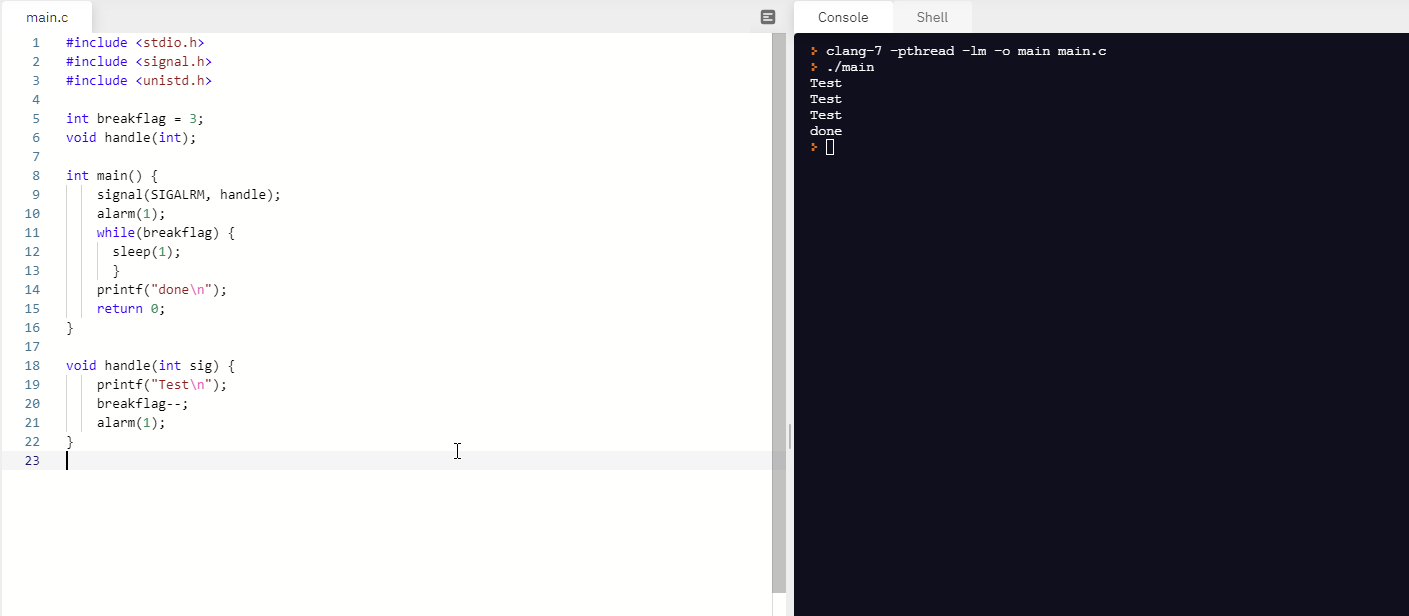
****

1. **wait()**

wait for a process to change. 

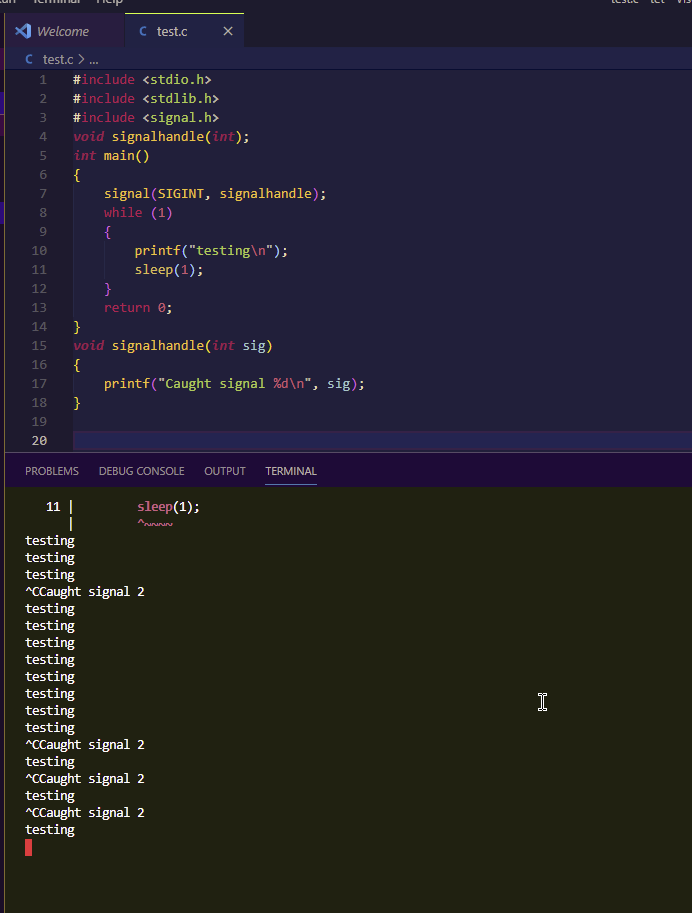
1. **alarm()**

set an alarm clock for a signal.



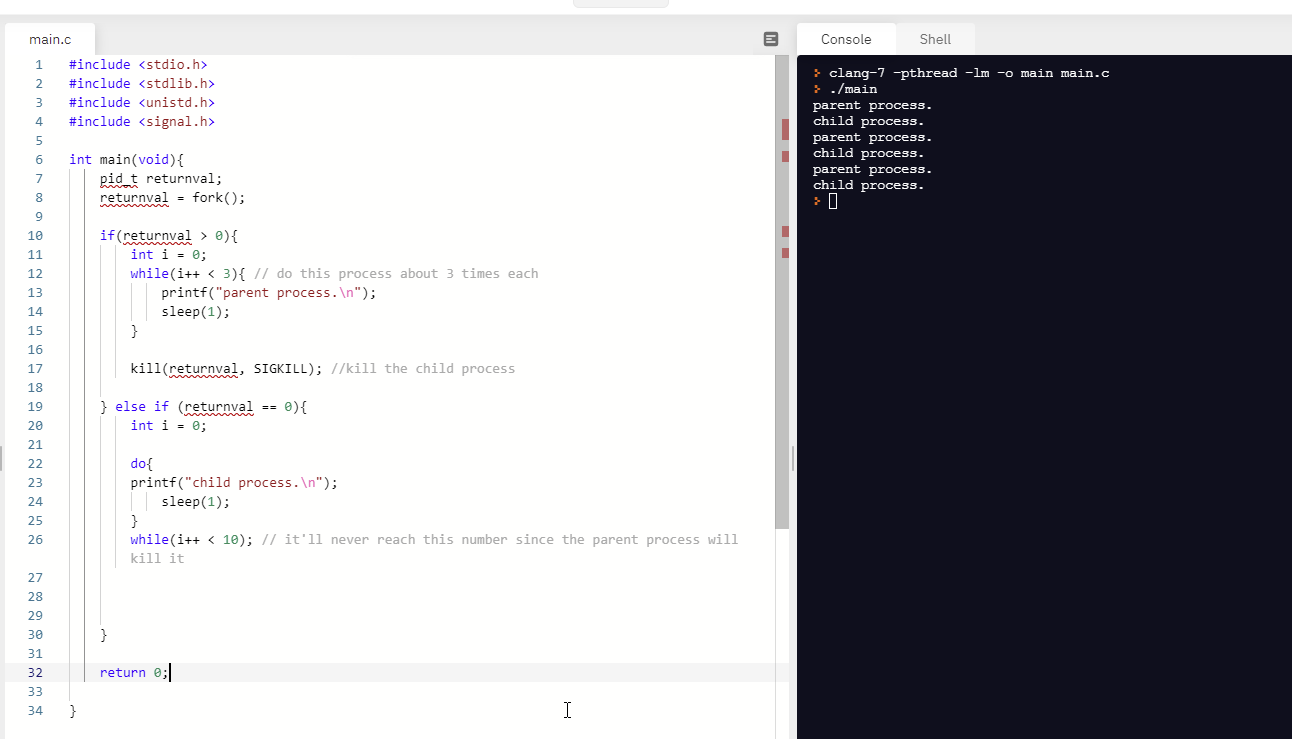
1. **signal()**

signal handler.



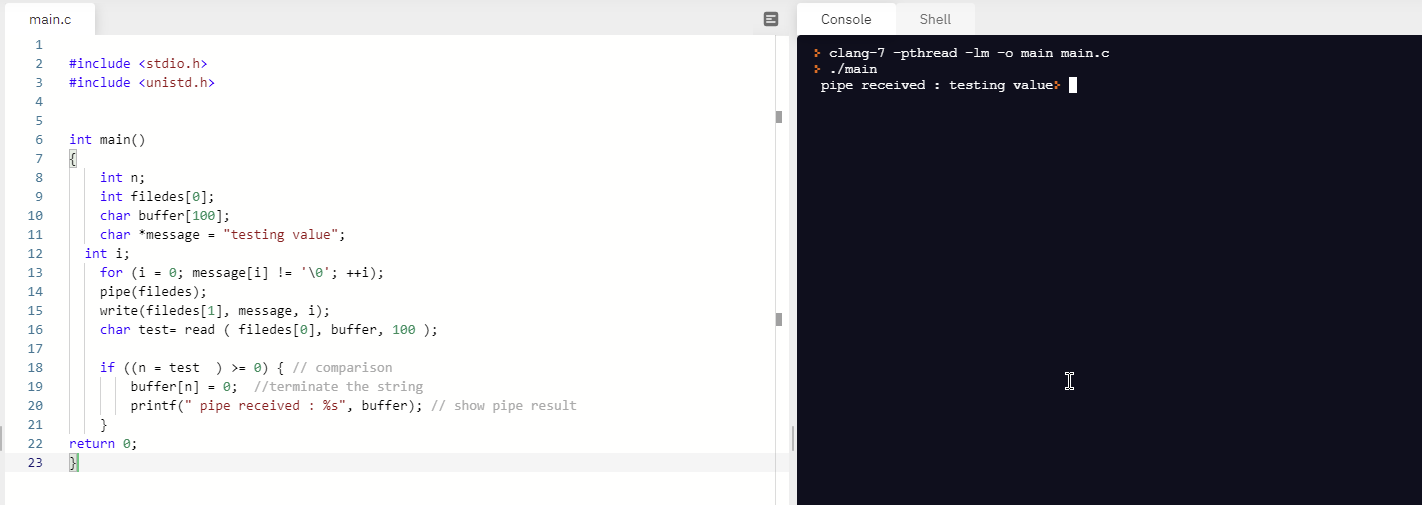
1. **kill()**

killing a process.



1. **pipe()**

connection between two processes.



1. **scp() (also referred to as secure copy)**

openSSH file copy. Copy between localhost and remote host.

I’m not able to implement this due to host issues.