


- Check carefully that no malicious aliases were used to fool you and make you evaluate something that is not the content of the official repository.
  - To avoid any surprises and if applicable, review together any scripts used to facilitate the grading (scripts for testing or automation).
  - If you have not completed the assignment you are going to evaluate, you have to read the entire subject before starting the evaluation process.
  - Use the available flags to report an empty repository, a non-functioning program, a Norm error, cheating, and so forth. In these cases, the evaluation process ends and the final grade is 0, or -42 in case of cheating. However, except for cheating, students are strongly encouraged to review together the work that was turned in, to identify any mistakes that shouldn't be repeated in the future.
- 

## Attachments

 subject.pdf (<https://cdn.intra.42.fr/pdf/pdf/28501/en.subject.pdf>)

## Preliminaries

*If cheating is suspected, the evaluation stops here. Use the "Cheat" flag to report it. Take this decision calmly, wisely, and please, use this button with caution.*

---

### Preliminary tests

- Defense can only happen if the evaluated student or group is present. This way everybody learns by sharing knowledge.
- If no work has been submitted (or wrong files, wrong directory, or wrong filenames), the grade is 0, and the evaluation process ends.
- No Empty git (= nothing in git repository).
- No Norm error.
- Cheating (= -42).
- No compilation error or re-link.

If all of these points are valid, check yes and continues the correction.

If not use the appropriate flag at the end of the scale!

 Yes

 No

# General instructions

---

## General instructions

- The Makefile compiles both executables -> 1 point
- Server name is 'server' and show his PID at launch -> 2 point
- Client name is 'client' and launches asfollowing:  
'./client PID\_SERVEUR STRING\_TO\_PASS' -> 2 point

Rate it from 0 (failed) through 5 (excellent)



---

## Mandatory part

*This project is an introduction to signals Check the code and usage of signals only to communicate.*

---

### Transmission of messages

It's possible to pass on a message of any size

Received messages must be displayed by the server, and must be obviously corrects!

The server should never get stuck or print bad characters.

☐ Yes

☐ No

---

### Simple setup

- Server can receive multi-string without being relaunched  
-> 1 point
- Only one global per program, with justification (or no global)  
-> 1 point
- The communication is achieves ONLY with the signals SIGUSR1 and SIGUSR2  
-> 3 points

Received messages must be displayed by the server, and must be obviously corrects!

Rate it from 0 (failed) through 5 (excellent)



## Bonus

Evaluate the bonus part if, and only if, the mandatory part has been entirely and perfectly done, and the error management handles unexpected or bad usage. In case all the mandatory points were not passed during the defense, bonus points must be ignored.

### Unicode

Unicode characters are supported in the client and server.

☒ Yes

☐ No

### Acknowledgment

The client waits for acknowledgment from the server to send the next signal.

☒ Yes

☐ No

## Ratings

Don't forget to check the flag corresponding to the defense

☒ Ok

☐ Empty work

☐ Incomplete work

☐ Norme

☐ Cheat

☐ Crash

☐ Incomplete group

☐ Leaks

☐ Forbidden function

## Conclusion

Leave a comment on this evaluation

Finish evaluation

Privacy policy (<https://signin.intra.42.fr/legal/terms/5>)

Terms of use for video surveillance (<https://signin.intra.42.fr/legal/terms/1>)

Rules of procedure (<https://signin.intra.42.fr/legal/terms/4>)

Declaration on the use of cookies (<https://signin.intra.42.fr/legal/terms/2>)

General term of use of the site (<https://signin.intra.42.fr/legal/terms/6>)

Legal notices (<https://signin.intra.42.fr/legal/terms/3>)