- To avoid any surprises, carefully check that both the correcting and the corrected students have reviewed the possible scripts used to facilitate the grading.

- If the correcting student has not completed that particular project yet, it is mandatory for this student to read the entire subject prior to starting the defence.

- Use the flags available on this scale to signal an empty repository, non-functioning program, a norm error, cheating etc. In these cases, the grading is over and the final grade is 0 (or -42 in case of cheating). However, with the exception of cheating, you are encouraged to continue to discuss your work (even if you have not finished it) in order to identify any issues that may have caused this failure and avoid repeating the same mistake in the future.

Attachments

Subiect (https://cdn.intra.42.fr/pdf/pdf/1000/ft_select.ro.pdf)

Introduction aux termcaps (https://cdn.intra.42.fr/pdf/pdf/1204/ft_select.fr.pdf)

Subject (https://cdn.intra.42.fr/pdf/pdf/965/ft_select.en.pdf)

Sujet (https://cdn.intra.42.fr/pdf/pdf/844/ft_select.fr.pdf)

Preliminaries

Basics

If one of those points are true then the defence is ended :
- Nothing turned in

- No author file

- Invalid author file

- Norm error

- Cheat

☑ Yes ☒ No

Features

As always, if there is a segfault/buserror/etc... it's the end of the defence.

Noncanonical mode

When ft_select is executed with a list of arguments, the terminal goes into noncanonical mode check that:

- Characters pressed on the keyboard aren't displayed.

- The terminal doesn't buffurize keyboard inputs until pressing return.

☑ Yes ☒ No

Choices list

When ft_select is executed with a list of arguments, in a window that's big enough for the whole list, the list is displayed at least on one column. It's also acceptable for the list is displayed on several columns.

Yes No

Navigation

When the list is displayed, the cursor must be on the first element. It's then possible to move from one element to the other by pressing arrows on the keyboard. Navigation through the list must be "circular" whether the list is on one or more columns. "Circular" is open to interpretation. It could mean for instance that only the last element goes to the first one. Every behavior that remains consistant is acceptable if properly justified especially when involving a display of several columns. It is therefore totally acceptable to go from one column to another using the left and right arrows.

Yes No

Current choice and selection

The active choice (cursor's position) must be underlined. Pressing space must select the current choice by going reverse video mode. Consequently the current choice will be underlined and in reverse video mode.

Yes No

Early termination

If at any time during the execution we press esc, the program must exit without errors. The original terminal option should be restored and nothing should be returned to the shell.

Yes No

Shell return

When at least one element has been selected, pressing the return key will send the selected choices (separated by exactly one space if there is more than one choice), back to the shell. Evidently, nothing more than the choices should be sent back (think control characters).

Yes No

Resizing

Resizing the window while keeping it big enough to display the whole list, possibly by reorganising the list on several columns or lines, shouldn't prevent us from using the program. Resizing the window in such a manner that the whole list can't be displayed should be properly handled. By "properly" we mean that the program is still running and resizing the window back to a proper size should restore the list.

Yes No

Element removal

If delete or backspace keys are pressed, the active choice (cursor's position) must be removed from the list. If there is no more elements in the list, the program must behave as if esc key was pressed.

Yes No

Signals 1/2

Whatever way your program exits, the default terminal options must be restored. That includes every catchable signals (not including uncatchable signals, but it'll mean that your program wasn't well developed anyway). Try to kill the process with ctrl+c or any other outside ways you come up with.

Yes No

Signals 2/2

It is possible to interrupt your program by pressing ctrl+z and to restore it using the 'fg' shell command without experiencing any problems while using the shell over the interruption or when getting back to ft_select afterwards.

☑ Yes ☒ No

Environnement

If your program is launched with either an empty or an incomplete environment, the program must behave reasonnably. By reasonnable we mean that for example the program exits directly by giving an error message. In no case can your program have an undetermined behavior because of the environment.

☑ Yes ☒ No

Back quotes

Ft_select must properly work in the following cases (use any combination you want as soon as you try using back quotes):

- `ft_select choix1 choix2 choix3 choix4`
- more `ft_select *.c`
- rm `ft_select ~/*`

Yes No

Bonus / Penalty

A lot of bonuses possible

- Columns are going left and right when the window is too small
- After termination, delete what needs to be deleted, the prompt and the cursor appear one line under the one used to run the program. Run 'top' to understand.
- A beautiful interface (up to the corrector to decide)
- If the choices are files names, colorize the list according to the extensions (a bit like ls -G on OSX.

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

Penalty

Not really a penalty: don't remove points for memory leaks or forgotten return values. It's dumb and useless:

- if the student already has a good grade, he will keep a correct grade.
- if he has a low grade, it will stay low. However, if you find memory leaks or fails on return values, "play" with them to try to crash their program. It will show how to be rigorous in his work.
- limit maxproc 25 (limit the nomber of process, in case of zombie it won't be able to execute commands).
- limit vmemoryuse X (where X is a limit, in bytes, a little more higher than the initial size of the process to check memory leaks).
- limit descriptors X (where X is the limit fd that will be opened, in the case where the program don't close opened files/pipes)

From 5 [everything's good] to 0 [lots of issues].
There will not be any associated points, it will not lower the grade.

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

Ratings

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

✔ Ok

Empty work | Incomplete work | No author file | Invalid compilation | Norme

Cheat | Crash | Leaks

Conclusion

Leave a comment on this evaluation

Finish evaluation