dtse

(https://profile.intra.42.fr)

(https://profile.intra.42.fr/searches)

(https://signin.intra.42.fr/users

Scale for project Fillit (/projects/fillit)

You should correct 2 students in this team

★

Git repository

```
vogsphere@vgs.42.us.org:intra/2016/activities/fillit/mhurd
```

Introduction

To ensure this evaluation goes smoothly, please respect the following set of rules :

- Please remain courteous, polite, respectful and constructive at all times during this exchange. The trust bond between the school's communauty and yourself depends on it.

- Should you notice any malfunctions within the submitted project, make sure you take the time to discuss those with the student (or group of students) being graded.

- Keep in mind that some subjects can be interpreted differently. If you come accross a situation where the student you're grading has interpreted the subject differently than you, try and judge fairly whether their interpretation is acceptable or not, and grade them accordingly. Our peer-evaluation system can only work if you both take it seriously.

Guidelines

- You may only evaluate whatever is in the GiT submission directory of the student you are grading.

- Make sure to check wether the GiT submission directory belongs to the student (or group) you're grading, and that it's the right project.

- Make sure no mischievous aliases have been used to trick you into correcting something that is not actually in the official submitted directory.

- Any script created to make this evaluation session easier - whether it was produced by you or the student being graded - must be checked rigorously in order to avoid bad surprises.

- If the student who is grading this project hasn't done the project him/herself yet, he/she must read the whole topic before starting the evaluation session.

- Use the flags available to you on this scale in order to report a submission directory that is empty, non-functional, that contains norm errors or a case of cheating, etc...
In this case, the evaluation session ends and the final grade is 0 (or -42, in case of cheating). However, unless the student has cheated, we advise you to go through the project together in order for the two (or more) of you to identify the problems that may have led for this project to fail, and avoid repeating those mistakes for future projects.

Go to

My projects (/)

All projects (/projects/graph)

List projects (/projects/list)

Your projects

Fillit (/projects/fillit)

profile.intra.42.fr)

rojects.intra.42.fr)

earning.intra.42.fr)

forum.intra.42.fr)

meta.intra.42.fr)

Attachments

test1 (/uploads/document/document/298/test1.prm) test7 (/uploads/document/document/299/test7.prm)

Subject (https://cdn.intra.42.fr/pdf/pdf/886/fillit.en.pdf) meta.intra.42.fr)

~~Sections~~

Preliminaries

/shop.intra.42.fr)

Setup

Please verify that :

- There are no libraries on the Git repository. There must be a script that fetches them when the work is compiled.
- The last available version of OpenGL is used. This is an OpenGL project.
- There MUST NOT be any graphical and sound assets on the Git repository. The students are allowed to fetch them separately.

Are all the above points correct ?

If not, the defense ends here.

✓ Yes ✕ No

Errors Handling

Here we will check whether all errors are handled

Number of parameters

Test the program without any parameters and with too many parameters.
The program must return an error message.

If it does not, the correction stops here.

Execute the program with a valid file as parameter. Does the program outputs as expected?
nIf it does not, the correction stops here.

✓ Yes ✕ No

Invalid piece

Execute the program with an invalid piece.
For example:
...#
..#.
.#..
#...
Try with a piece too big, a piece too small

Check whether the program displays an error message
If it doesn't, the correction stops here.

✓ Yes ✕ No

Invalid file

Execute the program with an invalid file, for example:
...#
...#
...#
...#
two empty lines
...#
..##
...#
Or an invalid file.

Check whether the program displays an error message
If it does not, the correction stops here.

✓ Yes ✕ No

Algorithm

Here we will check whether your algorithm has been correctly implemented

Everything is where it should be, as it should be?

Execute the program with a file formatted as follows:
#...
#...

```
#...
#...
*empty line*
....
....
..##
..##
```

Does the output matches this?

```
"###.
###.
#...
#...
```

If you get this instead:

```
"###.
###.
..#.
..#."
```

The algorithm isn't correctly implemented, the correction stops here.

✓ Yes ✕ No

Time

Here we will check your algorithm performance.

Simple test

Execute the program with the file test1.prm:
time ./fillit test1.prm
Does the result takes more than a second to be displayed?
In that case, answer no, and the correction stops

✓ Yes ✕ No

Advanced test

Execute the program with :
time ./fillit test7.prm
If the result takes more than 30 sec to be displayed => 0
20 to 30 sec => 1
10 to 20 => 2
5 to 10 => 3
1 to 5 => 4
Less than a sec => 5

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

5

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

Conclusion

Leave a comment on this correction

*** (required) Comment**

Finish correction