SCALE FOR PROJECT RTV1 (/PROJECTS/RTV1)

You should evaluate 2 students in this team

★

Git repository

vogsphere@vogsphere.21-school.ru:intra/2019/activities/rt 📋

Introduction

Please respect the following rules:

- Remain polite, courteous, respectful and constructive
throughout the correction process. The well-being of the community
depends on it.

- Identify with the person (or the group) graded the eventual
dysfunctions of the work. Take the time to discuss
and debate the problems you have identified.

- You must consider that there might be some difference in how your
peers might have understood the project's instructions and the
scope of its functionalities. Always keep an open mind and grade
him/her as honestly as possible. The pedagogy is valid only and
only if peer-evaluation is conducted seriously.

Guidelines

- Only grade the work that is in the student or group's
GiT repository.

- Double-check that the GiT repository belongs to the student
or the group. Ensure that the work is for the relevant project
and also check that "git clone" is used in an empty folder.

- Check carefully that no malicious aliases was used to fool you
and make you evaluate something other than the content of the
official repository.

- 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

🗋 Subject (https://cdn.intra.42.fr/pdf/pdf/1866/rtv1.en.pdf)

Mandatory part

Reminder : Remember that for the duration of the defence, no segfault, no other unexpected, premature, uncontrolled or unexpected termination of the program, else the final grade is 0. Use the appropriate flag. You should never have to edit any file except the configuration file if it exists. Otherwise the defence is over and the final grade will be 0. Those rules are actives thoughout the whole defence.

Author file

Check that the author file is at the root of the repository
and formatted as explained in the subject. If not defence is
over and the final grade will be 0.

⊘ Yes   ✕ No

Configuration file

Check that you can configure the point of view and the simples objects by a
configuration file or by passing parameters to the program.
If not defence is over and the final grade will be 0.

⊘ Yes   ✕ No

Display's technical component

In this section we'll evaluate the display's technical component. Run the program and execute the following 3 tests. If at least one fails, no points will be awarded for this section. Move to the next one.

- A window must open when launching the program and stay open during the whole program's execution.

- Hide either part of or the whole window with another one or the screen's borders, minimize the window and maximize it back. In every cases, the window's content must remain consistant.

- Ask the student to show you the code prooving that expose was managed properly and doesn't involve a re-calculation of the whole image or even only a part of it.

⊘ Yes   ✕ No

The 4 Basic Shapes

In this section we'll evaluate the 4 basic shapes. Run the program and execute the following 4 tests. If at least one fails, no points will be awarded for this section. Move to the next one.

- Place a sphere at the following coordinates {0, 0, 0}, the camera facing the sphere, and display the rendered image. The sphere should be visible and displayed without glitch.

- Place a plane with a null z value, the camera facing the plan, and display the rendered image. The plane should be visible and displayed without glitch.

- Place a cone's center at the following coordinates {0, 0, 0} extending along the y axis, the camera facing the cone, and display the image rendered. The cone should be visible and displayed without glitch.

- Place a cylinder extending along the y axis, the camera facing the cylinder, and display the image rendered. The cylinder should be visible and displayed without glitch.

⊘ Yes   ✕ No

Translations and rotations

In this section we'll evaluate that rotation and translation transformations can be applied on scene's objects. Run the program and execute the following 2 tests. If at least one fails, no points will be awarded for this section. Move to the next one.

- Place two spheres at the following coordinates {0, 0, 0}, the camera facing those spheres. Then put a translation on one of the two spheres oriented in a direction parallel to the camera's, of a greater distance than the sphere's diameter and display the rendered image. Both spheres should be visible and displayed without glitch.

- Place a cylinder extending along the y axis, the camera facing the cylinder. Then put a 90° rotation (PI/2 radian) along the z axis and display the rendered image. The cylinder should be visible and displayed without glitch.

⊘ Yes   ✕ No

Multi-objects

In this section we'll evaluate that it's possible to put several object in one scene. Run the program and execute the following 2 tests. If at least one fails, no points will be awarded for this section. Move to the next one.

- Place several intersecting objects on the scene, such as for example a sphere and a cone, and display the rendered image. Both objects should be visible and displayed without glitch. (especially where both object intersect).

- Execute the same test, but ensure it's possible to place several times the same object, for example two cylinders and a plane.

⊘ Yes   ✕ No

Camera's position and direction

In this section we'll evaluate that rotation and translation transformations can be applied on scene's objects. Run the program and execute the following 4 tests. If at least one fails, no points will be awarded for this section. Move to the next one.

- Generate a random scene and place the camera extending

along the x axis pointed towards the following coordinates
{0, 0, 0} and display the rendered image. The scene
must be visible and displayed without glitch.

- Generate a random scene and place the camera extending
along the y axis pointed towards the following coordinates
{0, 0, 0} and display the rendered image. The scene
must be visible and displayed without glitch.

- Generate a random scene and place the camera extending
along the z axis pointed towards the following coordinates
{0, 0, 0} and display the rendered image. The scene
must be visible and displayed without glitch.

- Generate a random scene and place the camera at a random
location which isn't on any axis or a diagonal, pointed
towards the following coordinates {0, 0, 0} and display
the rendered image. The scene must be visible and
displayed without glitch.

⊘ Yes   ✕ No

Brightness 1/2

In this section we'll evaluate brightness on scene's objects.
Run the program and execute the following 2 tests. If at least
one fails, no points will be awarded for this section. Move to
the next one.

- Place a sphere at the following coordinates {0, 0, 0}, the camera
facing the sphere, and put a spot left or right of the camera
but positioned in such a way that the sphere will be lit sideways.
Display the rendered image. The sphere should be visible,
enlightened and displayed without glitch.

- Place a sphere at some coordinates resulting from a translation,
the camera facing the sphere, and place a spot left or right of
the camera but positioned in such a way that the sphere will be
lit sideways. Display the rendered image. The sphere should be
visible, properly enlightened and displayed without glitch.
Properly means that the halo of light should be computed
after translation not before.

⊘ Yes   ✕ No

Brightness 2/2

In this section we'll evaluate shadow management generated by scene's objects. Run the program and execute the following 2 tests. If at least one fails, no points will be awarded for this section. Move to the next one.

- Place a vertical spot, a sphere and a plane. The spot liting the sphere's position to create a sphere shadow on the plane. Put the camera aside so we can see the sphere, the plane and the sphere's shadow on the plane. The shadow must be properly displayed and without glitch.

- Put a complex scene together with several objects like on illustration V.6 page 10 of the subject. Shadows must be properly displayed and without glitch.

$\circledcirc$ Yes $\times$ No

Bonus

Reminder : Remember that for the duration of the defence, no segfault, nor other unexpected, premature, uncontrolled or unexpected termination of the program, else the final grade is 0. Use the appropriate flag. This rule is active thoughout the whole defence. We will look at your bonuses if and only if your mandatory part is EXCELLENT. This means that your must complete the mandatory part, beginning to end, and your error management must be flawless, even in cases of twisted or bad usage. So if the mandatory part didn't score all the point during this defence bonuses will be totally IGNORED.

Shine effect

In this section we'll evaluate shine effect on the scene's objects. Run the program and execute the following test. If it fails, no points will be awarded for this section. Move to the next one.

- The shine effect is characterized by the saturation of brightness towards the color of the spot, where the object's surface is perpendicular to the light. Put togethere a scere with a spot enlightening a sphere without shine effect and then with shine effect. Is the shine effect clearly visible?

$\circledcirc$ Yes $\times$ No

Multi-spots

In this section we'll evaluate that it's possible to have several spots in the same scene. Run the program and execute the following test. If it fails, no points will be awarded

for this section.

- Put together a scene with several objects including at least a plane on which shadows will be projected as well as 2 spots at the minimum. Check that brightness, shadows and shine effect (if implemented) work properly.

⊘ Yes   ✕ No

Ratings

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

✔ Ok   ★ Outstanding project

◼ Empty work   ◼ Incomplete work   💬 No author file   💀 Invalid compilation   📋 Norme

📑 Cheat   ♟ Crash   👤 Incomplete group   💧 Leaks   ⊘ Forbidden function

Conclusion

Leave a comment on this evaluation

[   ]

Finish evaluation