Introduction

The methodology of School 21 makes sense only if peer-to-peer reviews are done seriously. Please read all guidelines carefully before starting the review.
- Please, stay courteous, polite, respectful and constructive in all communications during this review.
- Highlight possible malfunctions of the work done by the person and take the time to discuss and debate it.
- Keep in mind that sometimes there can be differences in interpretation of the tasks and the scope of features. Please, stay open-minded to the vision of the other.
- If you have not finished the project yet, it is compulsory to read the entire instruction before starting the review.

Guidelines

- Evaluate only the files that are in src folder on the GIT repository of the student or group.
- Ensure to start reviewing a group project only when the team is present in full.
- Use special flags in the checklist to report, for example, an "empty work" if repository does not contain the work of the student (or group) in the src folder of the develop branch, or "cheat" in case of cheating or if the student (or group) are unable to explain their work at any time during review as well as if one of the points below is not met. However, except for cheating cases, you are encouraged to continue reviewing the project to identify the problems that caused the situation in order to avoid them at the next review.
- Doublecheck that the GIT repository is the one corresponding to the student or the group.
- Meticulously check that nothing malicious has been used to mislead you.
- In controversial cases, remember that the checklist determines only the general order of the check. The final decision on project evaluation remains with the reviewer.
- Check whether the feedback form is filled or not (https://forms.yandex.ru/cloud/641819b3c09c022518e7a4f3/)

Main part   ^

Setting up your gitlab-runner

- The peer repository has a registered gitlab-runner, which is proven by launched and completed pipelines in the repository
- The peer has a virtual machine dump with a registered gitlab-runner

No	✓ Yes

Build

- The commit from the last pipeline matches with the last commit in the repository
- The build job from the last pipeline is successfully completed (this job is marked with a tick)
- The job has building artifacts which can be viewed and downloaded
- The artifacts contain the s21_grep and s21_cat utilities from the SimpleBashUtils project

No	✓ Yes

Codestyle test

- The commit from the last pipeline matches with the last commit in the repository
- The job of testing the codestyle from the last pipeline is successfully completed (this job is marked with a tick)
- The job contains the output of the clang-format utility for each SimpleBashUtils project s

Integration tests

- The stage is run automatically after building and testing the codestyle
- The commit from the last pipeline matches with the last commit in the repository
- The integration test job from the last pipeline is successfully completed (this job is marked with a tick)
- The job contains a message about the success/failure of the integration tests

No	✓ Yes

Deployment stage

- Peer is able to describe how files are transferred from one machine to another using the bash script
- The bash script uses the ssh and scp utilities for deployment
- The commit from the last pipeline matches with the last commit in the repository
- The deployment job from the last pipeline is successfully completed (this job is marked with a tick)
- There are executable files of the SimpleBashUtils (s21_grep and s21_cat) project in the /usr/local/bin directory on the second virtual machine