

notifications [See all notifications](#)



[0](#)

[aazra](#)

- [Manage slots](#)
- [View my profile](#)
- [Settings](#)
- [Logout](#)

[Have a problem?](#)

- [Profile](#)
- [Projets](#)
- [E-learning](#)
- [Forum](#)
- [Companies](#)
- [Meta](#)
- [Shop](#)

Menu

[My projects](#) [Holy Graph](#) [List projects](#) [Available Coursus](#)

Your projects

[C++ Module 03 Exam Rank 04](#)

Scale for project [webserv](#)

You should evaluate 3 students in this team

Git repository

`git@vogsphere.42paris.fr:vogs`

Introduction

Merci de respecter les règles suivantes:

- Restez polis, courtois, respectueux et constructifs pendant le processus d'évaluation. Le bien-être de la communauté repose là-dessus.
- Identifiez avec la personne évaluée ou le groupe évalué les éventuels dysfonctionnements de son travail. Prenez le temps d'en discuter et débattre des problèmes identifiés.
- Vous devez prendre en compte qu'il peut y avoir de légères différences d'interprétation entre les instructions du projet, son scope et ses fonctionnalités. Gardez un esprit ouvert et notez de la manière la plus honnête possible. La pédagogie n'est valide que si la peer-évaluation est faite sérieusement.

Guidelines

- Ne notez que ce qui est contenu dans le dépôt Git cloné de l'étudiant(e) ou du groupe.
- Vérifiez que le dépôt Git appartient bien à l'étudiant(e) ou au groupe, que le projet est bien celui attendu, et que "git clone" est utilisé dans un dossier vide.
- Vérifiez scrupuleusement qu'aucun alias n'a été utilisé pour vous tromper et assurez-vous que vous évaluez bien le rendu officiel.
- Afin d'éviter toute surprise, vérifiez avec l'étudiant(e) ou le groupe les potentiels scripts utilisés pour faciliter l'évaluation (par exemple, des scripts de tests ou d'automatisation).
- N'ayez pas une confiance aveugle dans les programmes de test ! Ces derniers pourraient être faux.
- Vérifiez le Makefile : pas de wildcard, ne re-link pas. Auquel cas, la note est de 0.
- Utilisez make, puis make --debug pour vous en assurer.
- Si vous n'avez pas fait le projet que vous allez évaluer, vous devez lire le sujet en entier avant de commencer l'évaluation.
- Utilisez les flags disponibles pour signaler un rendu vide, un programme ne fonctionnant pas, une erreur de Norme, de la triche... Dans ces situations, l'évaluation est terminée et la note est 0, ou -42 en cas de triche. Cependant, à l'exception des cas de triche, vous êtes encouragé(e)s à continuer la discussion sur le travail rendu, même si ce dernier est incomplet. Ceci afin d'identifier les erreurs à ne pas reproduire dans le futur.
- Pendant toute la durée de l'évaluation, aucun segfault ou autre arrêt inattendu, prémature ou incontrôlé ne sera toléré. Auquel cas, la note finale sera de 0. Utilisez le flag approprié.
Vous ne devriez jamais avoir à éditer un fichier hormis un fichier de configuration si existant. Dans le cas où vous souhaitez modifier un fichier, vous devez expliciter clairement les raisons de l'édition et être en accord avec l'étudiant(e) évalué(e) avant de faire quoi que ce soit.
- Vous devez aussi vérifier l'absence de fuites mémoire. Toute mémoire allouée sur le tas doit être libérée proprement avant la fin de l'exécution du programme.
Vous avez le droit d'utiliser tout outil disponible sur la machine tel que leaks, valgrind ou e_fence. En cas de fuites mémoire, cochez le flag approprié.

Attachments

[subject.pdf](#)

[tester](#)[ubuntu cgi tester](#)[cgi tester](#)[ubuntu tester](#)

Partie obligatoire

Vérifiez le code et posez des questions

- Lancez l'installation de siege avec homebrew.
- Demandez au groupe d'expliquer les bases d'un serveur HTTP.
- Demandez quelle fonction le groupe a utilisé pour le multiplexage d'I/O.
- Demandez une explication du fonctionnement de select() (ou équivalent).
- Demandez si un seul select() (ou équivalent) est utilisé et comment le groupe a géré l'acceptation du serveur et la lecture/écriture du client.
- Le select() (ou équivalent) doit être dans la boucle principale et doit vérifier les descripteurs de fichier pour lire et écrire EN MÊME TEMPS. Sinon, veuillez mettre un 0 et arrêter l'évaluation.
- Il ne doit y avoir qu'une seule lecture ou une seule écriture par client par select() (ou équivalent). Demandez au groupe de vous montrer le code qui va du select() (ou équivalent) à la lecture et à l'écriture d'un client.
- Recherchez tous les read/recv/write/send sur un socket et vérifiez que, si une erreur est retournée, le client est supprimé.
- Recherchez tous les read/recv/write/send et vérifiez si la valeur retournée est bien vérifiée (vérifier uniquement -1 ou 0 n'est pas bon, les deux doivent être checkés).
- Si une vérification de errno est effectuée après read/recv/write/send, veuillez mettre un 0 et arrêter l'évaluation.
- L'écriture ou la lecture de N'IMPORTE QUEL descripteur de fichier sans passer par le select() (ou équivalent) est strictement INTERDITE.
- Le projet doit compiler sans aucun problème de re-link. Sinon, utilisez le flag 'Invalid compilation'.
- Si un point n'est pas clair ou n'est pas correct, utilisez le flag 'Incomplete work'.

☐ Yes ☒ No

Configuration

Dans le fichier de configuration, vérifiez si vous pouvez effectuer les opérations suivantes et testez le résultat :

- Recherchez la liste des codes d'état de réponse HTTP sur internet. Si lors de cette évaluation des codes d'état sont erronés, n'accordez pas de point à la section correspondante.
- Configurez plusieurs serveurs avec un port différent.
- Configurez plusieurs serveurs avec un nom d'hôte différent (utilisez quelque chose comme : curl --resolve example.com:80:127.0.0.1 http://example.com/).

- Configurez la page d'erreur par défaut (essayez de changer l'erreur 404).
- Limitez le body client (utilisez curl -X POST -H "Content-Type: plain/text" --data "BODY IS HERE écrivez quelque chose de plus court ou plus long que la limite du body").
- Configurez des routes dans un serveur vers différents répertoires.
- Configurez un fichier par défaut à rechercher si vous demandez un répertoire.
- Configurez une liste de méthodes acceptées pour une certain route (ex : essayer de supprimer quelque chose avec et sans permission).

☐ Yes ☒ No

Vérifications basiques

L'utilisation de telnet, de curl, de fichiers préparés, démontre que les fonctionnalités suivantes fonctionnent correctement :

- GET request -> devrait fonctionner
- POST request -> devrait fonctionner
- DELETE request -> devrait fonctionner
- UNKNOWN request -> ne devrait pas produire de crash
- Pour chaque test le code d'état doit être bon
- Téléchargez un fichier sur le serveur et récupérez-le

☐ Yes ☒ No

Vérifications avec un navigateur

- Utilisez le navigateur de référence de l'équipe. Ouvrez la partie réseau de celui-ci et essayez de vous connecter au serveur.
- Regardez le request header et le response header.
- Il doit être compatible pour servir un site web entièrement statique.
- Essayez une mauvaise URL sur le serveur.
- Essayez de lister un répertoire.
- Essayez une URL redirigée.
- Essayez ce que vous voulez.

☐ Yes ☒ No

Problèmes de ports

- Dans le fichier de configuration, configurez plusieurs ports et utilisez différents sites web. Utilisez le navigateur pour vérifier que la configuration fonctionne comme prévu et affiche le bon site web.
- Dans la configuration, essayez de configurer le même port plusieurs fois. Cela ne devrait pas fonctionner.
- Lancez plusieurs serveurs en même temps avec des configurations différentes mais avec des ports communs. Est-ce que ça marche ? Si cela fonctionne, demandez pourquoi le serveur devrait fonctionner si l'une des configurations ne fonctionne pas. Continuez l'évaluation.

☐ Yes ☒ No

Siege & stress test

- Utilisez Siege pour exécuter des stress tests.
- La disponibilité doit être supérieure à 99,5% pour un simple accès sur une page vide avec un siege -b sur cette page.
- Vérifiez qu'il n'y a pas de fuite de mémoire (surveillez l'utilisation de la mémoire du processus, elle ne devrait pas augmenter indéfiniment).
- Vérifiez qu'il n'y a pas de connexion suspendue.
- Vous devriez pouvoir utiliser siege indéfiniment sans redémarrer le serveur (regardez siege -b).

☐ Yes ☒ No

Partie bonus

Les bonus ne seront examinés que si la partie obligatoire est excellente. Cela signifie que la partie obligatoire doit avoir été réalisée du début à la fin, avec une gestion d'erreur parfaite même en cas d'usage inattendu. Si tous les points obligatoires n'ont pas été attribués pendant cette soutenance, aucun point bonus ne sera comptabilisé.

Cookies et session

Il y a une session de travail et un système de cookies sur le serveur Web.

☐ Yes ☒ No

CGI

Il existe plusieurs systèmes CGI.

☐ Yes ☒ No

Ratings

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

Ok Outstanding project

Empty work Incomplete work Invalid compilation Cheat Crash Incomplete group Concerning situation Leaks Forbidden function

Conclusion

Flag

☒ Ok ☐ Empty work ☐ Incomplete work ☐ Invalid compilation ☐ Cheat ☐ Crash ☐ Incomplete group ☐ Outstanding project ☐ Concerning situation ☐ Leaks ☐ Forbidden Function

Leave a comment on this evaluation

Finish evaluation

[Terms of use for video surveillance](#)[Declaration on the use of cookies](#)[General term of use of the site](#)[Legal notices](#)[Règlement Intérieur](#)[Privacy policy](#)

Flash modal content (raw)