

RES - Challenge de la Saint Patrick

OLIVIER LIECHTI

20 mars 2017

Repo du challenge, où vous trouverez le code et ce pdf :

<https://github.com/SoftEng-HEIGVD/Teaching-HEIGVD-RES-2017-StPatrickChallenge>.

Formulaire pour le rendu :

<https://goo.gl/forms/LSf39RkatIM5AmkP2>.

Comme annoncé lors de la première leçon, la note de contrôle contenu pour le module RES dépend en partie d'exercices et de tâches proposés tout au long du semestre. Le challenge de la Saint Patrick est la première opportunité de récolter un point. Pour rappel, après 6 opportunités de récolter 1 point, la note est calculée selon la formule $\min(6.0, \text{nombre de points acquis} + 1)$.

1 Règles

Règle 1 : il est interdit de communiquer, que ce soit par oral, par écrit ou en utilisant un outil en ligne.

Règle 2 : en revanche, vous pouvez utiliser vos notes, votre code et faire des recherches sur le web.

Règle 3 : il est obligatoire de forker le repo contenant la donnée. Il est interdit de pusher des commits vers votre fork avant 12h40 et après 12h45. Après avoir pushé vos commits, il est également obligatoire d'ouvrir une pull request. Finalement, il est obligatoire de copier le hash de votre dernier commit et de le saisir dans le formulaire de rendu.

Règle 4 : pour récupérer le contenu du labo (tests, code), il est obligatoire d'utiliser les commandes git en ligne de commande. Il est interdit de faire des copier-coller de depuis l'interface web de GitHub pour récupérer des fichiers. Il est aussi interdit d'utiliser la commande "Download zip" offerte par git. Ce que vous pouvez faire via l'interface web, c'est la création de votre fork et la création de pull requests.

Règle 5 : attention : vous ne pouvez soumettre le formulaire Google Forms qu'une seule fois et vous n'avez pas la possibilité de modifier vos réponses par la suite.

Règle 6 : quand vous avez terminé, gardez vos commits au chaud (vous pouvez déjà remplir le formulaire Google Forms !). Vous pouvez commencer le labo de la semaine. A 12h40, poussez vos commits vers votre fork et faites la pull request.

2 Objectifs

Le premier objectif du challenge est de confirmer que vous maîtrisez le workflow de base que nous avons vu avec git et GitHub. Le deuxième objectif est de confirmer que vous êtes au point avec l'utilisation des classes d'entrées-sorties de base (comme mis en pratique dans le labo 01). Finalement, le troisième objectif est de confirmer que vous avez compris le principe de communication client-serveur au dessus de TCP. Le point est acquis si et seulement si tous les objectifs sont atteints.

3 Déroulement du challenge

3.1 Lecture des règles et du déroulement

Faites bien attention à l'heure pour pusher vos commits ! N'oubliez pas non plus de relever le hash de votre dernier commit.

3.2 Fork et clone du repo du challenge

Rien de spécial, vous connaissez la routine.

3.3 Lecture du code et des tests unitaires

Prenez bien le temps de lire les commentaires.

3.4 Implémentation du code pour faire passer les tests unitaires

Rien de spécial.

3.5 Traitement de l'annonce qui surviendra 20 minutes après le début du challenge

C'est à cette étape que vous devrez utiliser les commandes git, avant de reprendre le code.

3.6 Tâche relative à la communication client-serveur au dessus de TCP

Un leprechaun se cache dans l'ordinateur du professeur. En sachant que l'adresse IP de cet ordinateur est affichée au tableau et que le leprechaun a choisi le port TCP 1703 pour se mettre à votre écoute, vous devriez facilement obtenir la réponse qui vous aidera à gagner le point en jeu.



FIGURE 1 – Le leprechaun qui utilise TCP

3.7 Soumission des résultats via le formulaire Google Forms

1. votre adresse mail (heig-vd.ch)
2. votre identifiant GitHub
3. le hash de votre dernier commit
4. la réponse que vous aurez obtenue du leprechaun

3.8 Entre 12h40 et 12h45 : push des commits et ouverture de la pull request