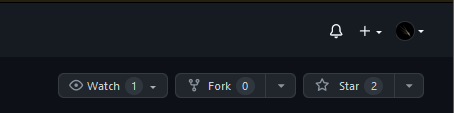
Application Pull Request

Etudiant C

1. Si ce n’est pas déjà fait, créez un compte GitHub (<https://github.com/signup>)
2. Attendez que l’Etudiant A vous fournisse son nom d’utilisateur GitHub ou l’url du projet qu’il vient de créer.
3. Trouvez le projet « syracuse\_project » récemment publié par votre collègue sur son profil
4. Faites un fork du dépôt vers votre profil :

En haut à droite de la page du projet de l’Etudiant A, cliquez sur le bouton « Fork »



1. Maintenant que vous avez créé un fork du projet de l’étudiant A, vous pouvez cloner votre Fork en local sur votre poste (l’url qui contient votre nom d’utilisateur)  
   Inspectez le code une fois que vous aurez cloné le projet.
2. Vous allez aussi pouvoir intégrer un lien vers le dépôt originel :

git remote add upstream [url repo etudiant A]

upstream est communément utilisé pour désigner la remote source d’un projet duquel vous dépendez. Ce lien vous permet de suivre les modifications qui pourraient être intégrées sur la branche principale du projet.

En utilisant la commande

git remote -v

Vous pouvez vois les remote configurées sur votre projet :

Par défaut, un projet cloné depuis GitHub possède la remote origin qui désigne la source initiale de votre travail.

1. L’algorithme de Syracuse que l’étudiant A vient de publier présente une multitude de trous à compléter :

Vous allez compléter **uniquement le cas IMPAIR.**

(Créez une nouvelle branche « add\_even » pour ajouter cette fonctionnalité)

**MAIS : Intégrez une faute grossière lors de votre ajout :)   
ou alors, ajoutez un fichier aléatoire dans l’arborescence du projet.**

1. Publiez votre nouvelle branche modifiée vers votre fork GitHub

git push origin add\_even

Puisque votre branche n’existe pas encore sur le serveur cible, git devrait se plaindre et vous fournir une commande adaptée afin de créer la nouvelle branche cible. La nouvelle commande devrait ressembler à :

git push -–set-upstream origin add\_even

1. Lorsque vous revenez sur votre dépôt personnel, un bandeau doit apparaitre, mentionnant qu’une nouvelle branche possède des modifications récentes.

Réalisez une Pull Request vers le dépôt de l’Etudiant A à partir de cette nouvelle branche

1. Attendez que l’Etudiant A, intervienne sur votre Pull Request.

Espérons qu’il y voit l’erreur que vous y avez introduit…

1. Corrigez l’erreur depuis la même branche que vous avez utilisez jusqu’à maintenant. Et publiez cette correction sur votre dépôt de la même manière que précédemment. A vous de voir, si vous souhaitez faire un nouveau commit, ou un amend.

(L’utilisation d’amend n’est pas recommandé puisque vous allez réécrire l’histoire. Dans ce cas, le serveur Github va vous interdire l’écriture, même sur votre propre dépôt. Pour forcer l’écriture, utilisez l’argument « -f » soit : « git push -f »)

Puisque votre branche « add\_even » fait actuellement l’objet d’une Pull Request, tous les commits que vous publierez sur cette branche seront automatiquement ajoutez à la PR.

1. Une fois votre Pull Request acceptée, revenez dans votre environnement de travail et tirer les modifications depuis le remote upstream :

git pull upstream main

Vous pouvez également tester que le script fonctionne (accessoirement)

1. Les modifications sur le projet cible ont maintenant été prises en compte.

Inspectez le fichier que vous avez maintenant en local.

Notez que vos modifications n’ont pas été les seules à avoir été ajoutée au projet. L’étudiant A (et C) ont également pu travailler sur le script.

Vous pouvez valider que le script fonctionne correctement depuis le terminal Matlab :

syracuse(chiffre)

1. Vous pouvez maintenant supprimer votre branche local et distante :

git branch -f add\_even

git push origin **:**add\_even

1. Essayez maintenant de collaborer simultanément pour rédiger une documentation pour l’algorithme dans le fichier README.md

Vous pouvez vous aider de <https://www.markdownguide.org/basic-syntax/>