

## Les règles du jeu

- **Règle I:**  
Vos enseignants ont raison.
- **Règle II:**  
Vos enseignants ont toujours raison.
- **Règle III:**  
Même s'il arrive (parfois ou souvent) qu'un apprenti ait raison, c'est la **Règle I** qui s'applique.
- **Règle IV:**  
Les étudiants choisissent leurs projets. Le fil rouge standard (adapté aux nouveaux programmeurs C) permet une validation standard des matières Prog C1 / Prog C2 (c'était le programme oppéré avant).
- **Règle V: (Abolie par l'arrêté du 29/09/2019)**  
Pour valider les matières de C, il faut obtenir un nombre de point total minimal ainsi que certains nombres de points minimaux par notions. Ces minimas correspondent à la validation minimale des matières (à laquelle on pourrait associer une note de 10 sur 20).  
**Arrêté du 29/09/2019 : Les objectifs chiffrés en termes de points vous donne seulement une estimation de la quantité de rendus attendue. Clairement la qualité est plus importante que la quantité. Attention, certains ne se rendent pas compte non plus de ce que l'on peut produire en 20 heures sérieuse de TP.**
- **Règle VI:**  
Possibilité de travailler en binôme selon les projets : les enseignants sont seuls juges (toujours cette vilaine **Règle I**). Il faut donc toujours demander en amont pour faire équipe sur un projet. On aime pas beaucoup le elo boost et il se peut qu'on vous interdise la duo queue.
- **Règle VII:**  
On ne peut pas faire équipe plusieurs fois avec la même personne, sous peine de ne plus valider les points pour les travaux concernés. Toujours ce elo boost, ça peut nuire à bon apprentissage.
- **Règle VIII: (Abolie par l'arrêté du 29/09/2019)**  
Les enseignants valident les rendus au cours de l'année. Ils sont seuls maîtres dans l'attribution des points :
  - Un enseignant peut ne pas vous valider certains points  
(ex1 : pas de fct récursive dans votre rendu, pas de point récursivité)  
(ex2 : vous ne démontrez pas une compréhension/utilisation judicieuse de la notion)
  - Un enseignant peut vous rajouter des points

(ex1 : vous proposez une attaque récursive intéressante non demandée)

(ex2 : votre Sudoku graphique garde en mémoire les coups précédents joués via une liste chaînée, vous validerez possiblement des points supplémentaires à propos des structures)

**Arrêté du 29/09/2019 : Il apparait que valider au fur et à mesure est impossible en terme de temps. Nous allons passer en mode auto-évaluation et peer-review en utilisant des grilles d'évaluation critériée. Il s'agira donc principalement d'évaluation formative mais non certificative.**

**Arrêté du 29/09/2024 : La review est dans certains cas quasi autant bénéfique que la production (en particulier quand vous reviewer un projet que vous avez vous même traité). Il faut faire un max de reviews.**

- **Règle IX:**

Vous devez trouver vous-mêmes les informations pour réussir vos propres projets :

- Classiques supports de cours
- Tutorial Internet
- Demander à votre chargé de TP
- Demander à un autre apprenti
- Manuel du développeur Unix
- Stackoverflow
- ...

- **Règle X:**

Toute réalisation de projet demande de rédiger un rapport qui sera intégré à votre portefeuille pour la programmation C. Vous devez convaincre par l'écrit que vous avez travaillé/acquis des notions ou compétences. Vous devez ainsi proposer une démonstration de l'acquisition de savoirs. Vous aurez donc à rassembler des preuves objectives et des traces de production illustrant vos capacités personnelles à résoudre des problèmes.

- **Règle XI:**

Vous pouvez proposer des projets personnels UNIQUEMENT en amont. Seuls les enseignants valideront ou refuseront votre demande. En outre, seuls les enseignants peuvent vous donner une idée des points potentiels que vous pourriez obtenir.

- **Règle XII:**

Vos enseignants endossent les responsabilités suivantes :

- Acceptation/Refus de vos choix de projet.
- Assistance durant de développement : faire des revues de code, aide aux débogage, suggestion de modélisation...
- Vous aiguillez pour trouver l'information.
- Proposer des points de cours rapides sur les notions, répondre aux questions de programmation.
- Vous conseiller sur vos choix de projet à faire.

**(Mise à jour avec l'arrêté du 29/09/2019)**

**Valider les rendus.**

- **Pensez vraiment à demander des reviews de code durant les TP (votre chargé de TP va en gros vous faire une grosse liste de reproches pour vous montrer vos axes de progrès...)**
-

