

## Projet de Programmation en C (LC4) : rendu et soutenance

---

### Dates et échéances :

- Déposez la version quasi-définitive du projet avant le 20 mai, 18h00.
- La soutenance aura lieu du 22 au 25 mai par créneaux de 25 minutes par binôme (voir la page Doodle pour s'inscrire <http://doodle.com/4d5e8rbcvx2nuxmy>).

### Le rendu :

- Sur DidEL dans la catégorie Travaux du cours LC42012.
- Il s'agit d'un travail de groupe, vous aurez donc besoin de vous inscrire dans l'un des groupes au préalable (catégorie Groupes).
- Format du rendu : vos documents de groupe sur DidEL (DidEL nous propose de les télécharger sous forme archivée automatiquement, vous n'avez pas à créer un fichier .zip ou équivalent).
- Le rapport doit être au format PDF.

### La soutenance :

- La durée totale est d'environ 25 minutes.
- Il faut prévoir une démonstration/présentation de 10-15 minutes environ mettant en avant les atouts de votre projet.
- Le reste du temps sera consacré aux questions du jury (pendant et après la présentation).
- **Attention** : le jury recompilera votre code avant votre démonstration ! Soignez votre Makefile.
- Il est possible (mais pas indispensable) de préparer des "transparents" électronique.

### Le rapport :

- Il consiste en un court résumé d'1 à 4 pages reflétant l'état final du projet.
- Il doit expliquer le(s) problème(s) que vous cherchez à résoudre et ceux que vous avez effectivement résolus. En particulier, signalez, le cas échéant, en quoi et pourquoi vous avez dévié des consignes proposées (notamment concernant le découpage en fonctions et les structures de données).
- Il devra faire l'état des lieux des fonctionnalités que vous avez programmées : marchent-elles comme vous l'espériez ? Quels bugs sont-ils encore présents ? Qu'essayeriez vous de faire pour les corriger si vous en aviez le temps ?

### Le projet :

- Le projet a été réalisé par votre groupe, et réellement par tout le groupe.
- Le projet est un programme en C89. Votre code devrait pouvoir être compilé sans erreur ni avertissement avec la commande `gcc -Wall -std=c89`.
- Un code sans bugs est préféré évidemment, mais à défaut, des bugs connus doivent être délimités et signalés.
- Le code doit être clair et lisible. En particulier, il devra être correctement indenté. Un découpage équilibré en fichiers (.c et .h) et fonctions est aussi apprécié.
- Chaque fonction doit être brièvement documentée par un commentaire (spécifiant entrées, sorties et effets de bord).
- Ajoutez les commentaires qui vous semblent nécessaire pour comprendre votre code source.
- Le nombre de fonctionnalités influera sur la note... à condition que ce ne soit pas aux dépens de la qualité.