

Perfectionnement à la programmation en C

Fiche de TP 0

L2 Informatique 2020-2021

Instructions générales

Ce document décrit des éléments à respecter scrupuleusement au cours des TP du module. S'y référer en cas de doute ou avant de poser une question.

Le manquement au respect de ces consignes iront du simple avertissement à l'exclusion, en passant par des sanctions sur la notation.

1 Sur le déroulement des séances

La présence aux séances de TP est primordiale et un émargement aura lieu au début de chacune. Aucun retard non justifié n'est accepté (5 minutes maximum sont tolérées exceptionnellement).

Lors de la première séance de TP, des binômes seront constitués. Chaque TP est à faire ainsi à deux, les groupes restant les mêmes pour toute la durée du module. Il ne peut y avoir qu'un unique groupe formé d'un seul étudiant si l'effectif total est impair.

Il est vivement conseillé d'échanger pendant les séances des informations avec son coéquipier. En revanche, il est seulement toléré de communiquer avec des étudiants d'un autre binôme. Dans tous les cas, tout échange doit se faire le plus discrètement possible.

Les téléphones portables (et tout objet apparenté) doivent être éteints (ou placés en mode silencieux et sans vibration) et rangés. Pour des raisons exceptionnelles, si en cas de force majeure un étudiant doit consulter son téléphone, il est tenu d'en informer l'enseignant et de lui demander l'autorisation au tout début de séance.

Il est strictement interdit de consulter des sites internet sans rapport immédiat avec l'enseignement dispensé dans le module.

2 À propos du travail à rendre

Tous les TP sont notés. Une note de TP est attribuée à chaque étudiant à l'issue du semestre. Tous les TP ne valent pas pour le même coefficient : les derniers TP ont des coefficients en général plus élevés que les premiers.

Chaque TP est à réaliser en une ou plusieurs séances et certains autoriseront un délai de rendu supplémentaire. Les modalités propres à chaque TP seront précisées dans leur énoncé.

Dans tous les cas, chaque rendu du TP est à déposer avant échéance du temps imparti sur la plate-forme prévue à cet effet. Le rendu est composé des sources (fichiers `.c` et `.h`), de fichiers textes éventuels et de rapports au format `.pdf` exclusivement. Il est autorisé également de joindre des fichiers image et son si cela est nécessaire au rendu. Il ne doit figurer dans le rendu aucun fichier régénérable (pas de `.o` ni d'exécutable). Le seul format d'archivage autorisé est le format `.zip` (les rendus utilisant un autre format d'archive ne seront pas considérés).

Tout travail rendu en retard ne sera pas considéré.

3 À propos du rapport

Un rapport doit accompagner chaque rendu de TP. Il est constitué, au minimum, des parties suivantes :

1. une introduction, présentant les objectifs du projet et ses enjeux ;
2. le corps du rapport, présentant les moyens mis en œuvre, les réponses aux questions posées explicitement dans les énoncés des sujets et toute autre information jugée utile à mentionner. Cette partie devra être structurée de manière adéquate en fonction de la nature du TP ;
3. une conclusion, évoquant les difficultés rencontrées, ce que le sujet a apporté ;
4. une première annexe, contenant les instructions pour compiler le projet ;
5. une deuxième annexe, contenant la documentation pour utiliser le projet.

4 Conseils à suivre

Voici quelques recommandations de programmation et de gestion des TP :

- lire entièrement chaque énoncé de TP avant de commencer à y répondre ;
- pour chaque fonction programmée, s'interroger sur les arguments qui peuvent poser des problèmes. Capturer ces cas à l'aide de pré-assertions ;
- utiliser au maximum les fonctions déjà programmées dans les nouvelles à écrire. Il faut éviter au maximum la duplication de code ;

- il ne suffit pas de programmer uniquement les fonctions qui sont demandées. Pour mener le sujet à bien, des fonctions supplémentaires et non mentionnées explicitement dans le sujet seront utiles. Soigner leurs noms ;
- dès qu'une fonction est écrite, la tester de manière à recouvrir tous les cas significatifs possibles et conserver les tests (le but étant de les relancer pour vérifier l'intégrité du programme lors de modifications futures) ;
- commenter le code en évitant absolument les commentaires inutiles. Il faut commenter chaque fonction écrite, juste avant son prototype. Il faut y mentionner les trois points suivants : (1) une description du rôle de la fonction, (2) une description de ses paramètres et (3) une description de ce qu'elle renvoie ;
- préférer la concision au maximum. Un code concis et lisible possède une grande valeur ;
- appliquer également toutes les autres recommandation de bonnes habitudes de programmation progressivement introduits dans le cours.