
Rédaction informatique et mathématique – L1 · S1

Projet – L^AT_EX & git

Votre projet, à rendre avant le **dimanche 16 janvier à 23h59**, consiste en la rédaction en L^AT_EX d'un exposé (*i.e.*, un rapport d'environ 4 à 10 pages¹) sur le thème suivant :

“La décomposition en produit de facteurs premiers”.

Il est à réaliser en équipe de deux ou trois² étudiants d'un **même groupe de TP**. Vous pouvez utiliser *teams*, configuré par groupe de TP sur [la page moodle du cours](#), afin de trouver un (ou deux) coéquipier(s) ou encore, par la suite, pour échanger (chat, visio) entre coéquipiers.

Votre collaboration doit obligatoirement se faire en utilisant **git** et [le gitlab de l'ISIMA](#). Votre chargé de TP ainsi que le responsable du cours devront en effet avoir accès à votre projet **gitlab**.

Ce présent document donne une liste d'éléments qui devront être trouvés dans le rendu ([section 1](#)), des pistes pour se lancer ([section 2](#)) incluant une aide à la configuration de **gitlab** ([section 2.2](#)) et la spécification du rendu ([section 3](#)).

1 Éléments attendus

Vous êtes assez libres en ce qui concerne le contenu et la forme de votre document. Votre document devra néanmoins inclure, sans s'y limiter, certains éléments.

1.1 Éléments de contenu

- l'introduction d'une notation adéquate pour l'ensemble des entiers naturels ;
- la définition des nombres premiers ;
- le théorème fondamental de l'arithmétique, et sa preuve découpée en deux lemmes prouvés (un pour l'existence de la décomposition, l'autre pour l'unicité de la décomposition) ;
- l'écriture réduite d'une factorisation en facteurs premiers (c'est à dire avec des exposants pour éviter de répéter un même facteur premier) ;
- la définition du PGCD et du PPCM de deux nombres entiers et l'explication de leur lien avec la décomposition en facteurs premiers ;
- un algorithme permettant de calculer une décomposition en facteurs premiers d'un entier donné en entrée ou, à défaut, d'en tester la primalité (indiquez clairement ce qu'est sensé faire votre algorithme) ;
- des paragraphes explicatifs ;
- des exemples ;

1. Le nombre de pages est donné à titre indicatif, et ne consiste pas en des bornes strictes.

2. donc ni un, ni quatre, ni plus.

- des références aux documents et ouvrages consultés, réunis dans une dernière section³ et comportant des liens hypertextes le cas échéant.

1.2 Éléments \LaTeX

Votre document devra regrouper un certain nombre d'éléments \LaTeX . La qualité et la propreté⁴ de votre code source sera également évaluée. Voici des éléments attendus :

- des formules mathématiques, dont certaines avec quantificateurs ;
- des sections (et possiblement sous-sections), apportant une réelle structure au contenu ;
- une table des matières ;
- des blocs sémantiques pour le(s) théorème(s), les lemmes, le(s) définition(s), et les preuves ;
- des listes ;
- des références internes selon le mécanisme vu en cours ;
- un algorithme, à écrire en pseudo-code ou bien dans le langage de programmation de votre choix, dans un bloc adéquat ;
- des méta-données (titre, auteurs, date) ;
- un entête de titre ;
- un préambule ne contenant que du code (import de paquet, définition de commande, *etc.*...) utile ;
- une définition de commande ;
- l'utilisation de la commande définie ;
- une inclusion de fichier.

1.3 Éléments git

Votre collaboration doit impérativement passer par `git` et le `gitlab` de l'ISIMA. Une aide à la configuration de `gitlab`, en vue de l'interaction entre coéquipiers, est donnée en [section 2.2](#). Vous pouvez également demander de l'aide à vos chargés de TP en cas de blocage, même si aucun délai de réponse n'est garantie.⁵

Votre dépôt `git`, qui doit être unique pour l'équipe entière, doit contenir les éléments suivants :

- des versions avec des messages significatifs ;
- au moins une version réalisée par chacun des membres de l'équipe (ceci implique une configuration de l'utilisateur `git` comme vu en TP) ;
- au moins une version obtenue par fusion de deux versions en conflit ;
- un fichier `.gitignore` tracé, qui permet d'avoir un statut propre du dépôt, quand bien même des fichiers de compilation non-tracés seraient présents ;
- un fichier `README.md` tracé, qui contiendra une brève description de votre projet (affichée sur la page principale de votre projet `gitlab`) ;

3. Vous pouvez par exemple placer la liste des références aux ouvrages (dont sites web) consultés dans un fichier séparé que vous incluez dans votre document principal.

4. *e.g.*, aération, indentation, longueur des lignes, *etc.*...

5. Évitez les demandes tardives !

- un ensemble de fichiers tracés bien choisi :
 - tous les fichiers **sources** nécessaires à l’obtention du rendu pdf du rapport ;⁶
 - les fichiers `.gitignore` et `README.md` sus-mentionnés ;
 - aucun fichier de compilation (`.aux`, `.pdf`, ...).

2 Pour bien commencer

2.1 Se documenter

Commencez par vous documenter sur les nombres premiers et la fameuse décomposition en produit de facteurs premiers. Un bon point de départ est [la page wikipédia traitant de ce sujet](#) (une compréhension exhaustive de son contenu n’est pas attendue). Tout au long de vos recherches, trie les informations qui vous semblent pertinentes, et notez les ouvrages (y compris les sites web) consultés, car vous devrez y faire référence en fin de rapport.³

2.2 Configurer gitlab

Choisissez un membre de l’équipe – que l’on nommera le *propriétaire* dans la suite – et faites-lui créer un nouveau projet **privé** sur [gitlab.isima.fr](#) comme ceci :

1. cliquez sur le bouton bleu Nouveau projet ;
2. choisissez la boîte Create blank project ;
3. choisissez un nom de projet, puis valider via le bouton bleu Create project.

Une fois créé de la sorte, le dépôt peut d’ores et déjà être cloné par le propriétaire en une copie de travail sur sa machine.

Dans un second temps, le propriétaire doit ajouter les autres membres de l’équipe au dépôt **gitlab**. Ceci peut être fait comme suit, depuis la page principale du projet :

1. allez dans le menu *Project information* (panneau latéral de gauche), puis sélectionnez le sous-menu *Membres* ;
2. ajoutez les autres membres de l’équipe, en précisant le rôle *Maintainer* (pour donner des droits en écriture et en lecture) mais sans préciser de date d’expiration de l’accès.

Désormais, les autres membres peuvent également cloner le dépôt sur leur machine.

Enfin, par la même méthode, ajoutez votre chargé de TP et le responsable du cours (Bruno GUILLON) au projet, en leur octroyant le rôle **Reporter**.

En cas de problème, contactez au plus vite votre chargé de TP.

6. Pour vous assurer de cela, il est conseillé de cloner votre dépôt avant le rendu, et de vérifier que le document compile correctement dans cette nouvelle copie locale de travail.

Configuration avancée. Il est possible d’aller plus loin dans la configuration de `gitlab`, en utilisant les clés `ssh` (et l’accès `ssh`). Cette configuration n’est pas obligatoire mais recommandée. Elle ne concerne pas directement votre dépôt, mais votre compte personnel sur le `gitlab` de l’ISIMA. Cela se passe [ici](#), où vous trouverez des liens de documentation pour créer votre clé `ssh`. Une fois votre compte `gitlab` configuré correctement, vos identifiants ne seront plus demandés à chaque échanges (`pull`, `push`) avec le dépôt `gitlab`.

2.3 Conseils

Il est vivement conseillé de se partager le travail. Une fois que vous aurez une idée de la structure de votre rapport, vous pouvez attribuer à chaque coéquipier un ensemble de parties à rédiger. Pour éviter au maximum les conflits (même si un conflit devra avoir lieu, puisqu’il est demandé, *c.f.* [section 1.3](#)), favorisez, dans la mesure du possible, le travail sur des fichiers distincts ou sur des zones distinctes d’un même fichier, et préférez les lignes courtes aux lignes longues.

2.4 Plagiat

Votre rapport est personnel (à l’équipe), et cela sera testé. Évitez donc de copier-coller des paragraphes depuis vos sources (qui seront probablement consultés par d’autres équipes) et privilégiez la reformulation.

3 Spécification du rendu

Le projet est à rendre avant le **dimanche 16 janvier à 23h59**. Des points seront abondamment retirés pour les rendus tardifs. Le rendu consistera en deux fichiers :

- le fichier texte `team.txt`, [à télécharger ici](#) puis à compléter ;
- le fichier `pdf` de votre rapport (environ 4 à 10 pages¹).

Tout deux sont à déposer sur [la page moodle du cours](#). Bien que non déposé sur moodle, le dépôt, auquel votre chargé de TP et le responsable du cours ont accès, fait partie du rendu. La version considérée sera celle du dernier `commit`⁷ (attention, les `commit` sont datés). Il est donc important de ne pas publier de versions inutiles après la limite de rendu, sous peine d’être sanctionné pour retard.⁸

Il est vivement conseillé de ne pas attendre la dernière minute pour effectuer un rendu. Il est possible d’écraser un rendu précédent sur moodle, le dernier sera pris en compte. Vérifiez bien que votre chargé de TP et le responsable du cours (Bruno GUILLON) ont tout deux accès à votre dépôt (attention, le rôle *guest* ne suffit pas, il faut donner au moins le rôle **Reporter**).

7. Ou bien celle du `commit` tagué *rendu* s’il y en a un.

8. Ou alors, il faut (re)définir le tag *rendu* sur la version qui doit être considérée, *c.f.* [note 7](#).