



Apprendre à coder avec Python

Consignes pour la notation du projet

Sébastien Hoarau - Thierry Massart

6 novembre 2018

Ce document donne des recommandations et une grille d'évaluation pour noter le projet de notre MOOC (ou plus généralement un projet de programmation niveau débutant).

Principes de base

La réalisation d'un projet informatique comprend plusieurs aspects qui font l'objet de choix qui doivent généralement être justifiés, ou de mise en application, généralement sous forme de programme structuré.

Tout d'abord, il est important que l'évaluation soit le résultat d'une notation :

- **positive**, qui consiste à donner des points quand certains aspects sont présents dans la solution, et
- **négative**, qui consiste à enlever des points sur certains aspects qui sont absents ou incorrects.

Il convient aussi in fine d'avoir une évaluation globale où un projet qui "fonctionne" se voit "naturellement" attribué des points même si plusieurs problèmes existent dans la réalisation. Dans l'autre sens, ce n'est pas parce qu'un programme ne fonctionne pas parfaitement, qu'il ne mérite pas une partie des points pour le critère résultat (voir plus bas).

Le mieux pour obtenir une telle note qui est le fruit d'une évaluation "négative et positive", est de découper la note finale en sous notes qui chacune évalue un aspect (qui de même fait l'objet d'une évaluation "négative et positive" avec évaluation globale).

Grille de notation du projet

Ci dessous une découpe de la note à donner au projet, que nous vous demandons d'appliquer dans vos évaluations pour obtenir une note uniforme pour tous.

Le nombre de points pour chaque catégorie est proposée pour une note finale sur 25 points. Comme il s'agit d'un projet pour débutant, les structures de données et les méthodes pour résoudre les problèmes sont données et ne nécessitent pas de recherche pour l'apprenant ; aucune note n'est donc donnée pour ces deux aspects.



Grille de notation du projet

Découpe 4 points : La découpe du code global avec commentaires initiaux, et respect de l'ordre des parties : import des modules externes, suivit des définitions des constantes globales, suivit des définitions des fonctions et enfin, code global.

- 4 points si les quatre parties de la découpe du programme sont présentes et l'ordre est respecté ;
- 3 points si la découpe est bonne sauf une des quatre parties qui manque ou est dans le mauvais ordre ;
- 2 points si la découpe existe mais deux des quatre parties manquent ou sont dans le mauvais ordre ;
- 1 points si trois des quatre parties de la découpe manquent ou sont dans le mauvais ordre ;
- 0 point si la découpe est inexistante ou n'a aucun élément satisfaisant.

Structuration en fonctions 4 points : La structuration en fonctions de bonne taille (25 lignes maximum) et cohérente (pas de fonctions définies mais non appelées par exemple) est réalisée :

- 4 points la structuration est bien réalisée ;
- 3 points si des fonctions sont définies, mais certaines sont soit incohérentes soit sont trop longues ;
- 2 points si des fonctions sont définies, mais certaines sont incohérentes et d'autres sont trop longues ;
- 1 points si une certaine structuration en fonction existe mais ne respecte pas du tout les règles de bonnes pratiques ;
- 0 point si aucune fonction n'est définie.

Bonnes pratiques 4 points : Les bonnes pratiques sont respectées en matière d'utilisation des noms des constantes, variables, fonctions..., de l'indentation... :

- 4 points si toutes les bonnes pratiques sont globalement respectées (éventuellement à une ou deux exceptions près) ;
- 3 points si les noms des constantes, variables et fonctions respectent les bonnes pratiques, mais l'indentation (4 caractères par incrément) n'est pas respecté ;
- 2 points les règles sont globalement respectées à quelques exceptions prêt à la fois au niveau indentations et noms de constantes, variables ou fonctions ;
- 1 points si certaines règles sont respectées mais globalement ce n'est pas le cas à la fois au niveau indentations et noms de constantes, variables et fonctions ;
- 0 points si les règles de bonnes pratiques en matière d'encodage ne sont pas respectées.

Commentaires 4 points : Commentaires et docstrings sont bien présents : (le docstring initial possède les informations suivantes : identité de l'auteur, date, ce que fait le programme, les fonctions possèdent un docstring décrivant les paramètres, ce que fait la fonction aux paramètres (par défaut, ils ne sont pas modifiés) et le résultat de la fonction, des commentaires pertinents existent quand c'est utile à la compréhension)

- 4 points si commentaires et doctsrings sont présent et respectent les règles de bonnes pratiques ;



- 3 points si un des éléments docstring initial ou commentaires est manquant ou non satisfaisant ;
- 2 points si docstring initial et commentaires sont manquants ou non satisfaisants mais que les fonctions ont des docstrings corrects ;
- 1 point si certains éléments de commentaire ou docstrings existent mais globalement ces éléments sont absents du programme ;
- 0 point si les trois éléments docstring initial, commentaires et docstrings de fonctions sont manquants ou non satisfaisants.

Consignes 2 points : Les consignes données dans l'énoncé du projet sont respectées :

- 2 points si les consignes du projet sont respectées ;
- 1 point si la moitié des consignes est respectée ;
- 0 points si aucune consigne n'est respectée.

Résultat 7 points : Le programme fonctionne correctement

- 7 points si le programme fonctionne toujours correctement ;
- 6 points si le programme donne les bons résultats sur la plupart des jeux de données, mais ne fonctionne pas pour au moins un jeu ;
- 4 points si le programme donne le bon résultat sur au moins un jeu de données ;
- 2 points si le programme s'exécute sans échec sur au moins un jeu de données ;
- 0 points si le programme ne s'exécute pas sans échec.

Explication sur l'évaluation

En plus des points attribués, il est essentiel pour l'évaluateur, d'expliquer pourquoi telle ou telle note a été donnée, pour que l'auteur du projet puisse comprendre les notes données et puisse s'améliorer les fois suivantes.



Critère	Commentaires
Découpe	
Structuration en fonctions	
Bonnes pratiques	
Commentaires	
Consignes	
Résultat	