

**CAHIER DES CHARGES** (Apprentissage par Projet / © Active Learning by Gaming Campus)

**NOM DU PROJET**

**Initiation à l'algorithmique avec Python**

**Numéro** : 3

**Date** : 31/10/2022 au 10/11/2022

**Classe concernée** : Bachelor Guardia Première Année

**Enseignements couverts par le projet**

∉ Algorithmique

∉ Compréhension du Pseudo-Code avec LARP et sur papier

∉ Langage Python hors frameworks

**Modalités d’apprentissage**

Travail en groupe de 3

**Intitulé, contexte et descriptif du projet**

Nous allons découper ce projet en 2 parties pour offrir un périmètre de compétences le plus complet possible. Durant ces 2 semaines il est nécessaire que vous ayez acquis les principes de pseudo-code et d’algorithmiques ainsi que les fondamentaux en Python. Pour ce faire, nous allons travailler sur 2 Mini-Projets qui vont constituer le projet final :

Première Semaine : Vous allez découvrir le pseudo-code, nous allons créer ensemble différents algorithmes sur LARP et sur papier pour que vous compreniez comment créer des structures conditionnelles, comment penser un logiciel et à terme, comment penser un jeu. Vous devrez à la fin de la première semaine être capable de créer un algorithme reprenant le jeu du Pierre Feuille Ciseau

Deuxième semaine : Maintenant que vous maîtrisez le pseudo-code et l’algorithme, nous allons voir ensemble comment fonctionne le Python, nous allons reprendre nos pseudo-code et nous allons le translater en Python, vous allez donc reprendre votre projet de Pierre feuille ciseau, Vous allez donc développer votre premier jeu en Python. Mais pas que ! Ce développement devrait vous prendre une petite poignée d’heures, vous allez ensuite créer le jeu du morpion, vous allez donc créer l’algorithme et le développer en python.

**Problématiques**

● Comprendre comment rédiger un algorithme.

● Comprendre comment interagir avec votre machine.

● Éviter les mauvaises pratiques.

● Avoir connaissance des bases pour être un bon programmeur.

● Créer sa première interface graphique

**Échéances et livrables**

**Étapes clefs : PRÉSENCE OBLIGATOIRE À l'ÉCOLE**

**31/10 / KICK OFF**

∉ 9h à 13h – Présentation du projet

**2/11 / COACHING**

∉ 9h – 13h – Prise en main de LARP, commencer ses premiers algorithmes sur papier et digitalisés

**4/11 / COACHING**

∉ 9h – 13h – Commencement de l’algorithme du Pierre Feuille Ciseau sur LARP

**7/11 / COACHING**

∉ 9h – 13h – Présentation du Python, et des différentes composantes constitutives – Utilisation de l’algorithme PFC pour la présentation ∉ Travail individuel demandé pour bien avancer

**8/11 / COACHING**

∉ 9h – 13h – Nouveau projet : Création d’un morpion avec algorithmie en python directement

**9/11 / COACHING**

∉ 9h – 13h – Amélioration et finalisation du code en python du Morpion

**10/11 / SOUTENANCE**

∉ 9h – 13h – Présentation des algorithmes en Pseudo-Code, et présentation du Morpion en python

| **Dates**  **échéances** | **Livrables attendus** | **Moyens / formats**  (comment on transmet  le livrable) | **Remarques** |
| --- | --- | --- | --- |
| Avant le  début du  projet | Constitution des environnements de travaux | Création d’un GitHub et d’un repository pour chaque sujets |  |
| 31/10 | Premiers  Algorithmes  Simples avec LARP | Fichier LARP sur Repo Git |  |
| 2/11 | Algo v0 PFC | Fichier LARP ou Capture papier, sur le repo GIT |  |
| 4/11 | Algo final pseudo code PFC | Fichier LARP Obligatoire sur le repo GIT | Merci de  mettre  “FINAL” dans votre  nomenclature pour bien  reconnaitre le fichier final. |

| 7/11 | Translation Pseudo Code -> Python  Algorithme PFC | Fichier PFC.py sur Github |  |
| --- | --- | --- | --- |
| 8/11 | V0 Algo Python du Morpion | Morpion.py sur Github |  |
| 9/11 | V1 de l’algo Python du Morpion | Morpion.py sur une autre  branche Git pour versionner |  |
| 10/11 | Soutenance l’après midi | Dossier de soutenance dans le dossier Ressources du projet sur Google Drive |  |

**Étapes clefs**

Préparation durant le KickOff :

● Sur son ordinateur :

⊄ Installer l’environnement de programmation en Python

o Installer un IDE : par exemple Sublime Text ou Visual Studio Code ⊄ Mettre en place un Git :

■ Qu'est-ce que c’est :

https://www.atlassian.com/fr/git/tutorials/what-is-git

■ Pour le mettre en place : https://git-scm.com/

● A consulter avant le cours

● Sur OpenClassrooms, faire le cours “Découvrez le fonctionnement des algorithmes”.

● Constituer les trinômes et remplir le tableau sur le Drive “Constitution des groupes”.

**Ressources**

● L’introduction au Python sur OpenClassrooms.

● Pour aller plus loin dans l’algorithmique, un autre cours sur OpenClassrooms. ● Revoir les bases du Python : https://courspython.com/bases-python.html

**Acquis d’apprentissage détaillés**

● Rédaction d’algorithmes simples puis complexes.

● Utilisation et apprivoisement de LARP

● Découverte de la ligne de commande.

● Découverte de l’interface graphique

● Exploitation du langage : Python.

● Bases du langage :

○ Variables,

○ Types,

○ Boucles,

○ Conditions

○ Graphes

○ Interfaces

○ Interactions

○ Structure de données : tableaux, matrices, tuples, dictionnaires ○ Manipulation de données

○ Traitement de données

○ Utilisation de ressources

● Utilisation d’un IDE.

● Exploitation d’une interface graphique

● Ecrire ses premiers programmes

● Ecrire des algorithmes complexes

**Compétences à acquérir**

**A l’issue du projet, l’étudiant sera capable de :**

● Comprendre les attentes de la pédagogie par projet.

● Connaître les bonnes pratiques de développement.

● Comprendre les principes de base de l’algorithmique.

● Rédiger des algorithmes de base.

● Dérouler des algorithmes sur papier.

**GRILLE D’ÉVALUATION DU PROJET**

**Numéro : 3**

**Date : 10/11**

**Noms des membres du jury :**

| **Connaissances / Compétences** | **Barème** |
| --- | --- |
| Fluidité de l’Algorithme | **/ 35** |
| Logique Algorithmique | **/ 35** |
| Qualité du code en Python | **/ 35** |
| Gestion des anomalies | **/ 35** |
| Bonus : Qualité de la prestation | **/35** |
| Bonus 2 : QCM | **/35** |
| Bonus 3 : Easter Egg | **/40** |
| TOTAL | **/250** |
|  |  |
|  |  |