**FICHE PROF :**

|  |  |
| --- | --- |
| Auteur(s) : | Tautu TAMATAI / Chistian LAU / Raihei BOOSIE |
| Cycle(s) / Niveaux : | Lycée / Première Spécialité NSI. |
| Mots clés : | Recherche occurrence, parcours séquentiel. |
| Type de tâche assignée à l’élève : | Programmer en Python, calculer une complexité, faire une preuve par terminaison. |
| Objectif(s) : | Concevoir un algorithme de recherche d’occurrence dans un cadre précis. |

|  |  |
| --- | --- |
| **Programme officiel** | |
| Thème(s) : | Algorithmique. |
| Attendu(s) de fin de cycle : | Écrire un algorithme de recherche d’une occurrence sur des valeurs de type quelconque. |
| Prérequis : | -Algorithmique : variables, opérations élémentaires (affectation, arithmétique), instructions conditionnelles, boucles bornées et non bornées.  -Programmation : algorithmique en langage Python, définir une fonction, fonctions prédéfinies en python (len(), print(), input()).  -Notion de complexité asymptotique.  -Notion de terminaison, correction partielle et totale d’un algorithme. |

|  |  |
| --- | --- |
| **Proposition de mise en œuvre** | |
| Durée : | 55 min. |
| Matériel(s) / Logiciel(s) nécessaires | Logiciels de programmation Python 3 sur ordinateur : EduPython, Python Pyzo, Syder ou Jupyter notebook. |
| Support(s) de travail pour l’élève : | Feuille d’activité / ordinateur. |
| Support(s) de travail pour le professeur : | Feuille d’activité / tableau / ordinateur. |
| Organisation spatiale de la salle : | Salle informatique, idéalement avec des ordinateurs disposés en forme de U. Au centre de la salle, des tables et des chaises en face d’un tableau velleda. La salle est équipé d’un vidéo projecteur. |
| Mode de fonctionnement de la classe : | Les élèves travaillent sur ordinateur et le professeur possède un ordinateur (fixe ou portable) avec lequel il pourra faire de la vidéo projection de l’activité et des corrections. |

|  |  |
| --- | --- |
| **Anticipation et différenciation** | |
| Réponse(s) attendue(s) (finales, partielles ou autres) : | -Problème pour le choix des valeurs de début et de fin de boucle « for » ou pour la condition de boucle « while ».  -Décision pour la valeur de l’indice à renvoyer : i ou i+1 si gagnant trouvé.  -Erreurs de syntaxe Python (indentation, : , etc).  -Fonction récupérant la taille d’une liste. |
| Coups de pouce : | -Fonction « len() » pour la taille d’une liste.  -Choix du début et de la fin de boucle « for » ou de la condition de boucle « while ».  -Correction syntaxe Python.  -Valeur de l’indice à renvoyer si gagnant trouvé.  -Aide comptage des opérations pour trouver la complexité asymptotique.  -Aide preuve de terminaison de la boucle « while ». |
| Différenciation d’organisation : | 1 ou 2 élèves par ordinateur. |
| Différenciation des consignes : | Reformulation des consignes ou coups de pouce. |
| Différenciation des moyens de réponses : | Explications orales, par écrit sur tableau ou par logiciel de programmation Python 3 en vidéo projection. |
| Complexification de la tâche : | -Ecrire des algorithmes utilisant les deux types de boucles « for » et « while ».  -Preuve de la terminaison de la boucle « while ».  -Calcul de la complexité asymptotique des algorithmes utilisés. |

|  |  |  |  |
| --- | --- | --- | --- |
| **Postures envisagées** | | | |
|  | **Durée** | **Elèves** | **Enseignant** |
| **Préparation** |  |  |  |
| **Entrée en classe** | 5 min | Les élèves s’installent au centre de la salle pour recevoir les consignes du professeur. | Le professeur fait l’appel, puis distribue l’activité et le lit avec les élèves.  Il répond ensuite aux éventuelles questions. |
| **Mise en activité** | 10-15 min | -Les élèves vont sur les ordinateurs.  -Ils réfléchissent à la situation 1 de l’activité.  -Ils mettent par écrit leur algorithme en langage Python sur le logiciel qu’ils auront choisi (Edupython, Pyzo, Syder, Jupyter notebook) et le testent.  -Ils appellent le professeur pour vérification ou pour avoir des explications supplémentaires.  -Les élèves écoutent et regardent les aides apportées par le professeur au tableau. | -Le professeur demande aux élèves de se répartir sur les postes.  -Il demande aux élèves de faire la situation 1 de l’activité.  -Il passe auprès des élèves pour vérifier qu’ils sont bien en activité.  -Il répond aux éventuelles questions.  -Si un blocage général (ou que certains élèves ont du mal à démarrer) se fait sentir (au bout de 5 min), le professeur attire l’attention des élèves au tableau pour leur donner un coup de pouce. |
| 5-10 min | -Les élèves écoutent et regardent la correction de la situation 1 au tableau et corrigent leur algorithme en fonction. | -Le professeur attire l’attention des élèves au tableau et fait la correction de la situation 1. |
| 5-10 min | -De la même manière, les élèves travaillent sur la situation 2 en se basant sur la correction de la situation 1. | -Le professeur demande aux élèves de faire la situation 2 de l’activité.  -Il repasse auprès des élèves et répond aux éventuelles questions. |
| 5 min | -Les élèves écoutent et regardent la correction de la situation 2 au tableau et corrigent leur algorithme en fonction. | -Le professeur attire l’attention des élèves au tableau et fait la correction de la situation 2. |
| 5 min | -Les élèves réfléchissent aux questions du professeur sur la terminaison de la boucle « while ». | -Le professeur prouve de manière interactive avec les élèves la terminaison de la boucle « while ». |
| 5 min | -Les élèves réfléchissent avec le professeur au calcul de complexité asymptotique. | -Le professeur calcul de manière interactive avec les élèves la complexité asymptotique des algorithmes des situations 1 et 2. |
|  |  |  |