**Comprendre la sélection naturelle grâce à la Programmation d’algorithmes sous Scratch**

L.Roux

L’enseignement de la sélection naturelle se heurte aux préconceptions déterministes : Pour nombre d’élèves, l’adaptation des êtres vivants est due au milieu qui produit les variations de façon orientée. Cette idée reste souvent ancrée jusque tard dans les cursus de nos élèves et étudiants et ce malgré la découverte des processus darwiniens qui sont donc souvent mal compris.

On propose ici d’étudier deux algorithmes produisant de « l’adaptation » : un algorithme déterministe et un algorithme darwinien (« variation / sélection »). L’objectif est ensuite de recenser les conditions logiques pour que ces deux programmes fonctionnent et de vérifier lesquelles sont réalisées dans la nature. Il s’agit donc de construire (partiellement) et faire fonctionner deux modèles puis de choisir lequel rend le mieux compte des propriétés des êtres vivants.

Si on peut demander aux élèves en début d’activité de concevoir une structure logique du programme, il n’est pas réaliste de faire réaliser tous les codes par les élèves. Après une phase de réflexion, les programmes sont donc fournis. (Si le professeur n’oriente pas la réflexion des élèves, il est de toute façon peu probable que la structure d’un programme de « variation / sélection » soit imaginée).

Selon le travail de programmation et/ou d’étude du code effectué par les élèves et leur maitrise de Scratch, on peut réaliser l’activité proposée à différents moments de la scolarité (Seconde, Terminale).

L’ensemble de la démarche devrait couvrir 3 ou 4 heures.

* *Démarche générale :*

Le prétexte de programmation est volontairement éloigné d’une situation naturelle (afin de ne pas confondre modèle et simulation). On propose donc aux élèves de concevoir la structure logique d’un programme au cours duquel la population d’un village s’adapte à l’arrivée de zombies. On peut en début de séance montrer un tel programme en fonctionnement. Les élèves constatent donc le résultat et on leur demande « comment ça marche ».

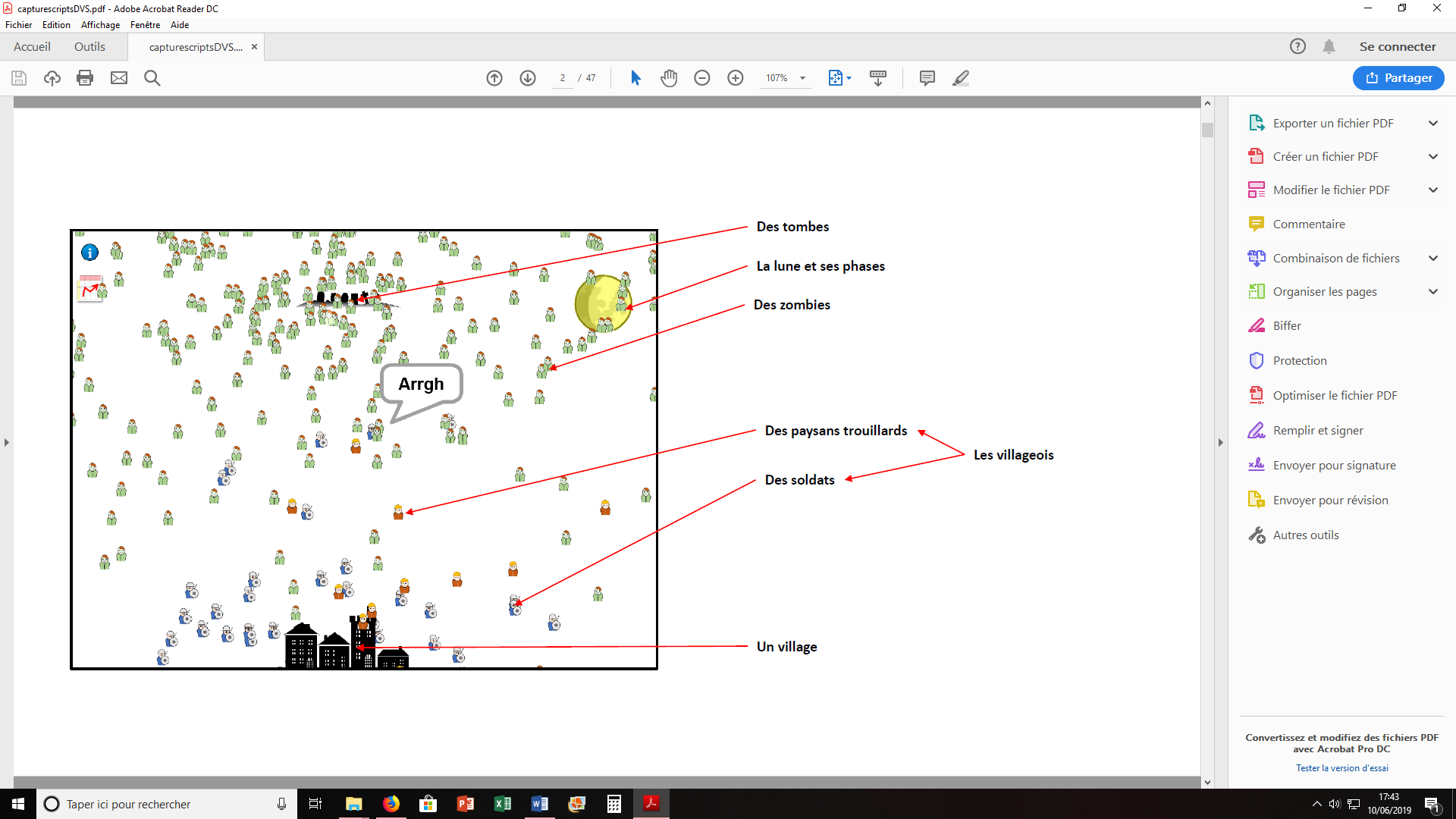
* *Phase de réflexion / conception \_ description d’un programme capable d’adaptation :*

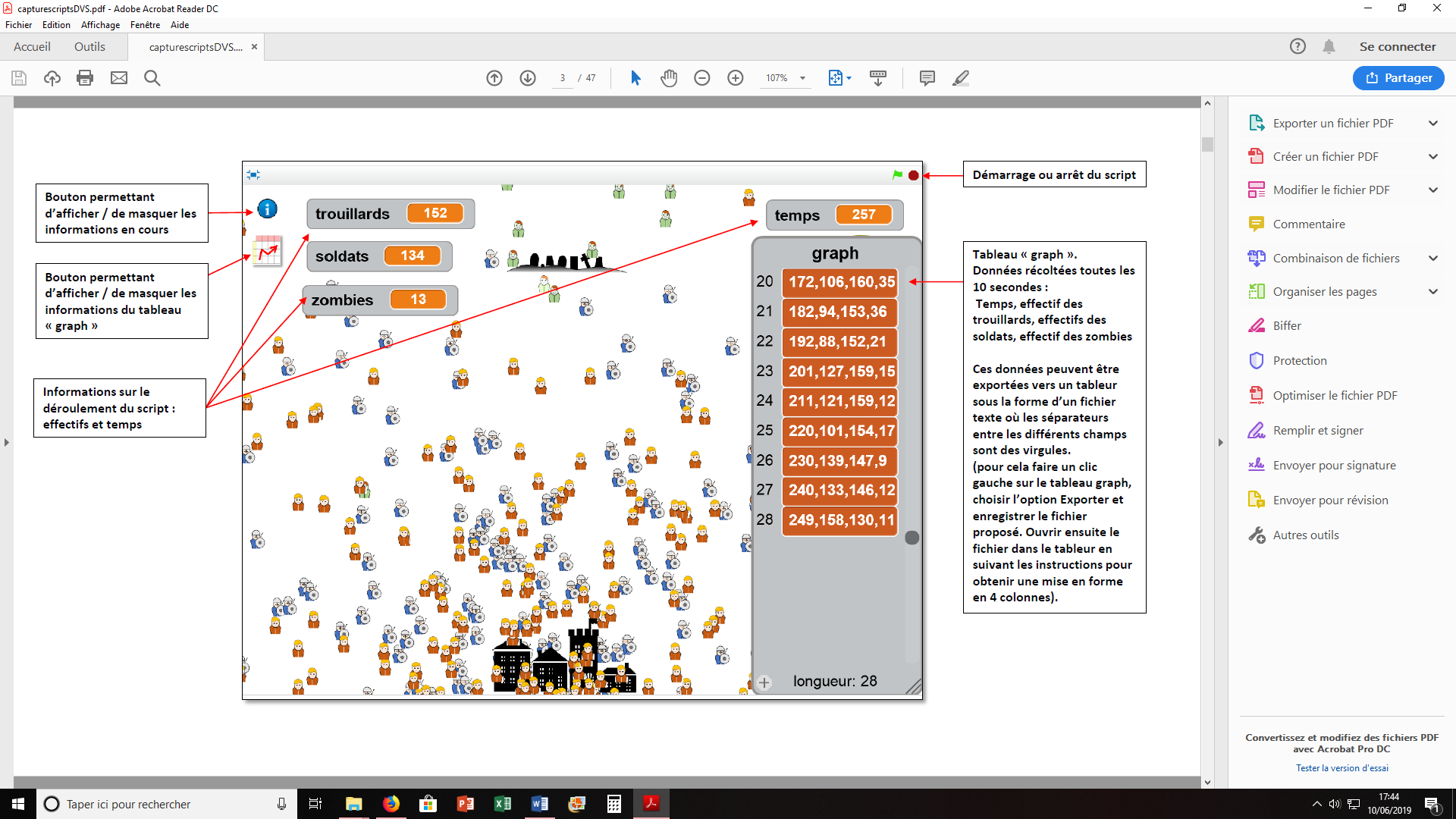
Sous scratch, les comportements d’objets-images (appelés « lutins ») sont modifiés par des fonctions. Chaque objet possède donc son propre code qui définit les interactions possibles avec d’autres objets. La possibilité de cloner des objets un nombre assez important de fois permet de construire des ensembles dans lesquelles ces propriétés d’interactions sont les mêmes pour toutes les entités. En revanche les paramètres (direction de déplacement, vitesse, aspect..) de chaque entité lui sont propres. Il est donc possible, par exemple, de concevoir une population de villageois qui bougeront chacun aléatoirement mais qui tous auront les mêmes interactions avec d’autres objets (les zombies ici).

Lors de la phase de réflexion, on impose les objets suivants :

* Village : lieu où sont créés les villageois à intervalle de temps régulier ou sous condition. Objet statique.
* Villageois « trouillards » : objet doué de mouvements aléatoires au départ du village. En cas de rencontre avec un zombie il meure. Un zombie apparait alors au niveau du cimetière
* Villageois « soldats » : objet au mouvement aléatoire qui s’il rencontre un zombie soit meure, soit supprime le zombie.
* Cimetière : objet d’où partent les zombies à intervalle de temps régulier ou parcequ’un villageois meure
* Zombie ; objet au mouvement aléatoire
* Tout villageois meure au bout d’un temps donné (40 s dans le script fourni)
* Au départ, la population est constituée de villageois uniquement. En présence de zombies, le programme doit donc augmenter la proportion de soldats pour que le village soit protégé.

Figure 1 : Exemple de capture d’écran d’un des deux programmes sous scratch.





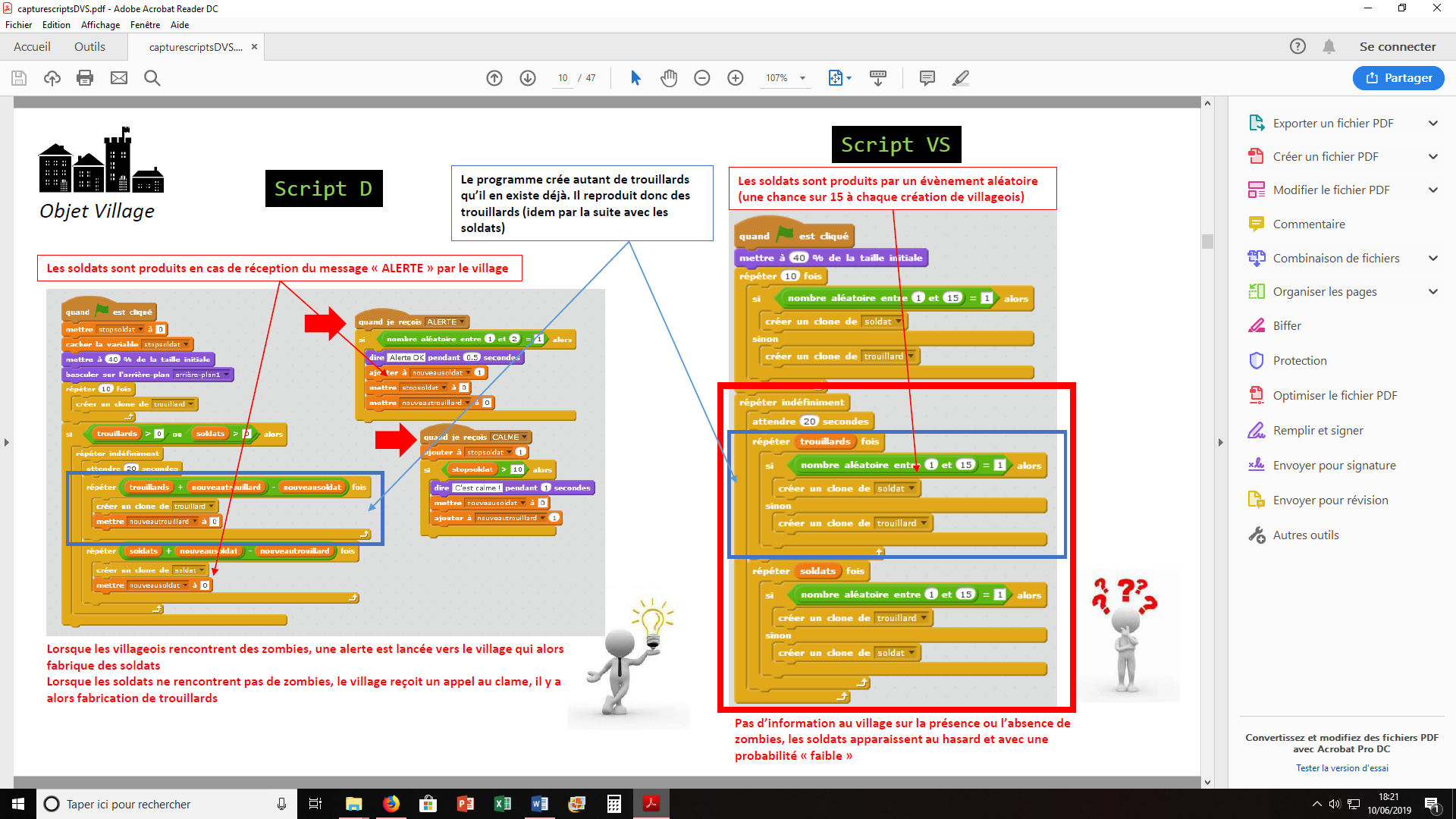
Le plus simple pour expliquer le fonctionnement de ce type de code est d’imaginer qu’il existe une fonction chez les villageois du type :

« *Si rencontre avec un zombie alors le village fabrique des soldats* » (Cette façon d’adapter la population du village à la présence de zombies sera nommée « déterministe » car c’est la présence de zombies qui déclenche –détermine- la fabrication des soldats)

On peut alors laisser les élèves chercher dans les deux scripts mis à leur disposition cette fonction « Si … Alors.. ». (selon les effectifs, la maitrise du code, le degré de guidage on peut soit faire analyser les deux codes à tous les élèves soit fournir un code à une moitié de classe l ‘autre à l’autre moitié).

* *Phase d’analyse du code*

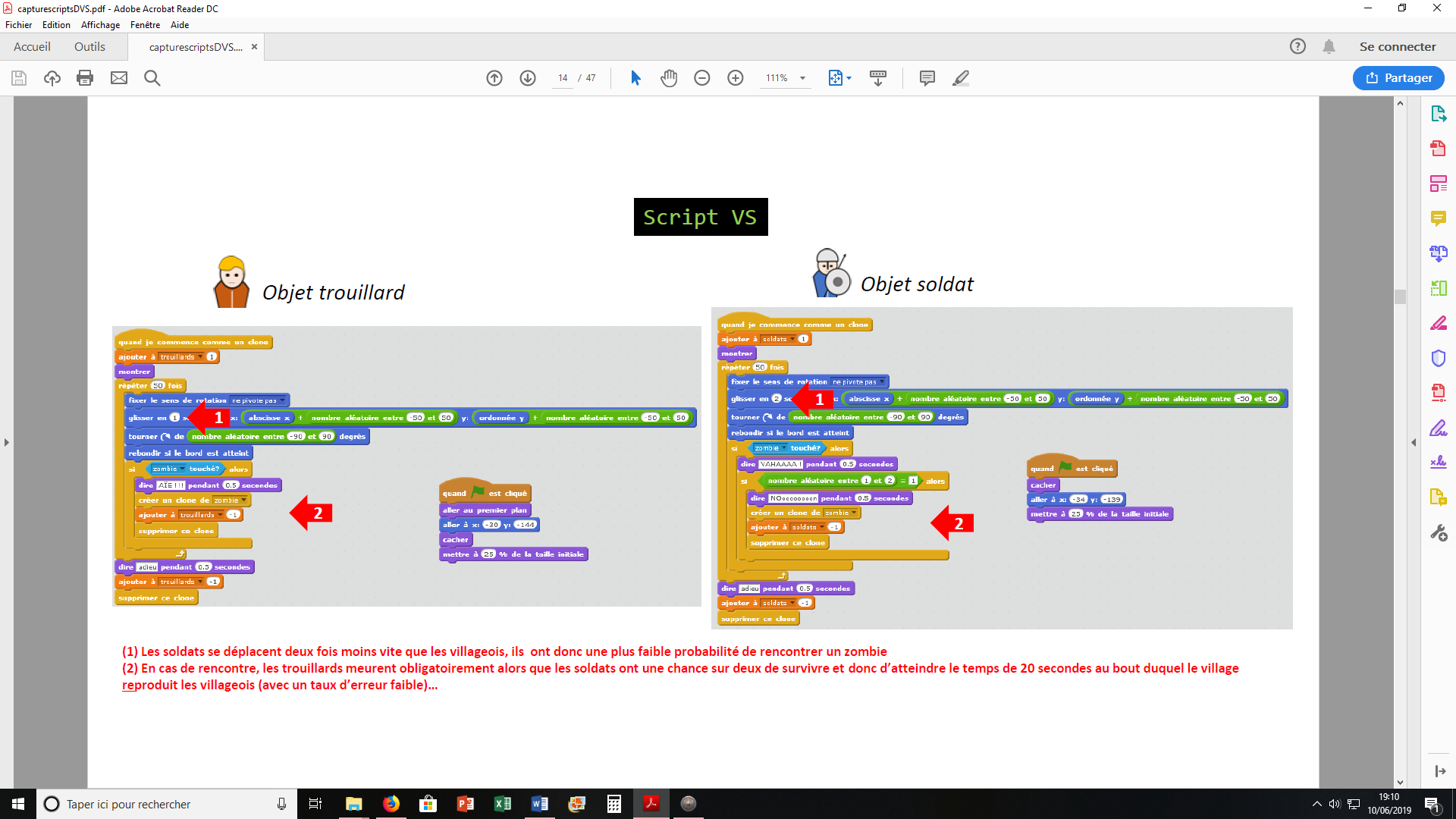
Figure 2 : Analyse du code pour l’objet Village, script déterministe « D » et Variation / sélection « VS »



L’élève doit remarquer que le script « VS » produit de l’adaptation sans pour autant que le système de fabrication des soldats (le Village) ne reçoive d’information sur la présence ou l’absence de zombies. Dans ce type de programme on ne retrouve pas la fonction « Si.. Alors » prévue.

Si le script VS produit quand même le résultat attendu c’est en raison de la différence de comportement entre les objets villageois « trouillards » et villageois « soldats » lors des rencontres avec les zombies.

Figure 3 : Analyse des interactions trouillard / zombie et soldat / zombie dans la programme VS

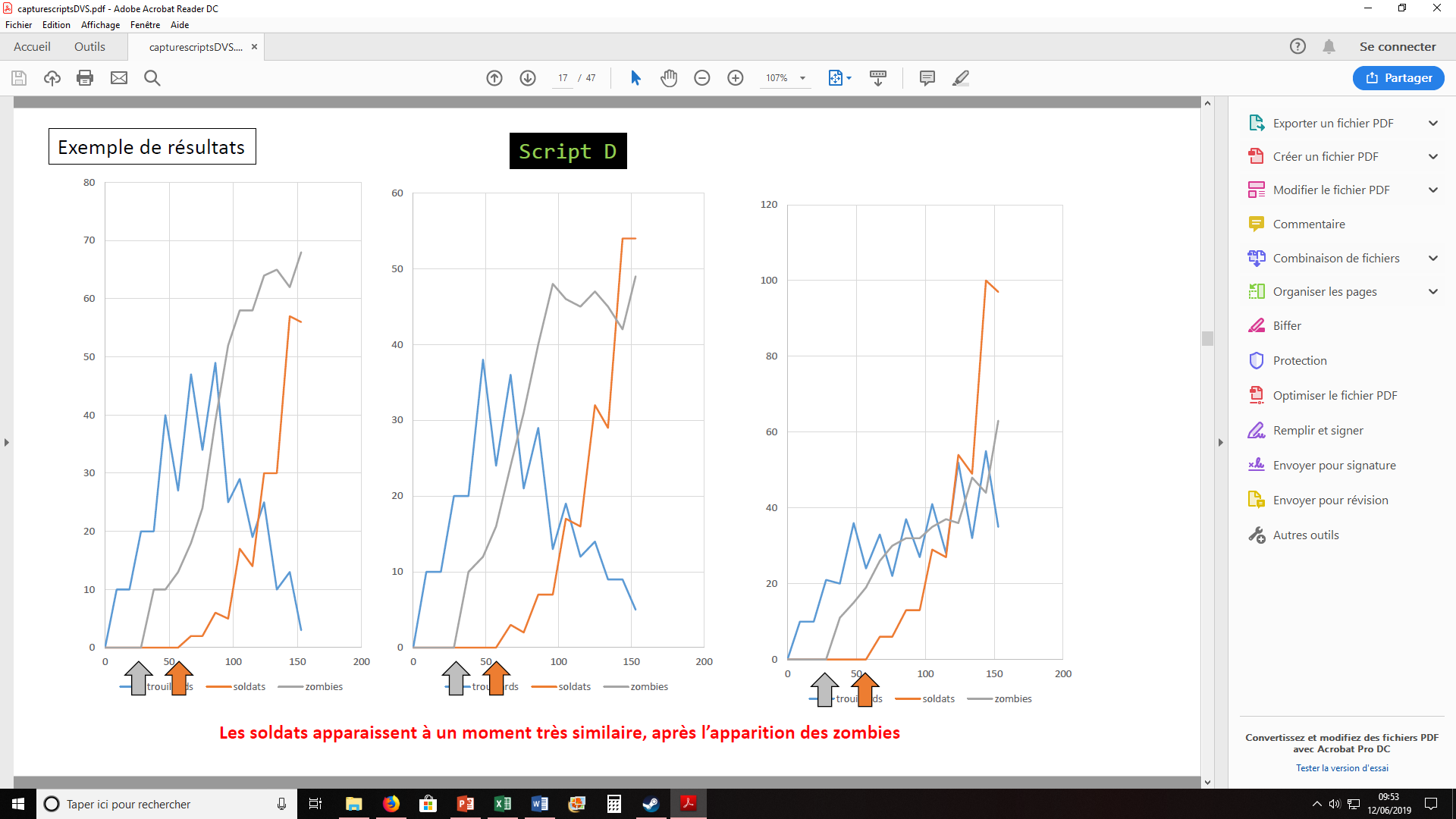


Le programme enregistrant les effectifs des villageois et des zombies toute les dix secondes, il est possible d’analyser plus finement le fonctionnement des deux algorithmes sous Excel.

Figure 4 : *Exemple de suivi des effectifs sur le temps long lors du fonctionnement du programme « Variation/sélection »*

On peut donc, par exemple comparer les temps d’apparition des soldats lors du fonctionnement des deux programmes sur plusieurs essais…

Figure 5 : Comparaison des moments d’apparition des soldats dans les deux programmes.





* *Lequel des deux processus D ou VS rend le mieux compte de l’adaptation des êtres vivants ?*

On peut après ces constatations rechercher si les populations d’âtres vivants obéissent davantage à une logique « VS » ou à une logique « D ».

Il s’agit donc à travers l’analyse de quelques documents confrontés aux résultats d’amener les élèves aux conclusions suivantes :

* Les cellules germinales constituent une lignée isolée

Au début du XXème siècle, A Weissmann défend l’idée que les cellules germinales ne sont pas influencées par le milieu. **Cet isolement interdit toute influence du milieu qui pourrait diriger le changement d’un caractère héréditaire**.

Depuis on sait que les mutations des cellules somatiques ne sont pas héréditaires et que les mutations sont des évènements aléatoires. Si elles peuvent être déclenchées par le milieu (agents mutagènes) les changements provoqués n’ont pas de rapport avec le facteur déclenchant : Par exemple, une mutation ne permet pas de résister au facteur qui a déclenché l’évènement de mutation.

**Dans le modèle D le «village» (système de reproduction des villageois) doit recevoir des informations de l’extérieur (ALERTE ou CALME) pour produire plus de soldats.**

Dans la nature, un tel mécanisme est difficile à imaginer si les cellules germinales ne peuvent être influencées par les facteurs de l’environnement.

Le modèle déterministe apparait donc peu pertinent.

* L’apparition des variants dans la nature est indépendante des changements de milieu

Dans les années 1940, Lüria et delBrück démontrent que l’apparition de bactéries résistantes à des phages précède l’introduction des virus dans les cultures de bactéries.

Chez Biston betularia, On a pu évaluer la date de plus grande probabilité d’apparition de l’allèle du gène cortex à l’origine de l’apparition du mélanisme industriel. La mutation serait intervenue vers 1830, c’est-à-dire avant la révolution industrielle. (<https://planet-vie.ens.fr/article/2558/mutation-origine-melanisme-industriel-phalene-bouleau>)

Autres exemples ?

Un mécanisme de variation aléatoire / sélection par le milieu rend mieux compte de ces données (voir figure 5)

* Des discussions demeurent sur l’importance des phénomènes « épigénétiques »

La découverte de mécanismes modifiant l’expression des gènes de façon héréditaire pose de nouveaux problèmes : si les mutations ne sont pas orientées par le milieu, l’expression des gènes, elle peut l’être. Si certains de ces modes d’expression sont héréditaires alors on peut imaginer une part de mécanismes déterministes.

(<https://www.pourlascience.fr/sd/biologie/lepigenetique-est-un-modulateur-cle-de-levolution-17064.php>)

* Conclusion : but recherché et faisabilité

La démarche proposée ici cherche à :

* Déconstruire l’idée qu’une adaptation est forcément liée à une influence directe du milieu sur l’apparition de nouvelles formes forcément adaptées
* Permettre à l’élève de formaliser un mécanisme de variation / sélection (ici la sélection naturelle mais on peut transposer ce type de mécanisme à beaucoup d’autres situation, la sélection clonale des lymphocytes par exemple …) en termes de « variation aléatoire + hérédité + filtre par le milieu »
* Donner à la sélection naturelle le statu d’un modèle (et non d’une loi au sens de la physique) de dynamique des populations
* Faire comprendre qu’en tant que modèle, la sélection naturelle est généralement le plus pertinent pour expliquer l’adaptation des êtres vivants parce qu’il tient compte des propriétés connues de la génétique.

In fine, l’élève doit comprendre pourquoi les scientifiques expliquent à priori l’adaptation par la sélection naturelle (hypothèse raisonnable et parcimonieuse).