

ANALYSE D'UNE SITUATION D'APPRENTISSAGE DE LA PROGRAMMATION "RENTRE AU PORT",
DEDIEE A LA DECOUVERTE DE LA VARIABLE INFORMATIQUE AU CYCLE 4

Nadia ZEBICHE, Christophe DECLERCQ

Résumé : L'apprentissage de la notion de variable informatique au cycle 4 se heurte à la fois aux difficultés conceptuelles de la notion déjà identifiées par Samurçay et Rogalski dans les années 1980, et aux obstacles engendrés par les caractéristiques du milieu avec l'utilisation de l'environnement Scratch. Nous avons présenté une situation originale "Rentrer au port", dont la résolution nécessite d'abord l'utilisation d'une variable accumulateur puis d'une véritable variable d'état. Nous avons explicité l'usage que l'on a fait de cette situation en formation initiale d'enseignants. L'atelier a proposé aux participants d'analyser un corpus de données constitué d'une série de travaux d'élèves récoltés en classe de 3eme.

L'objectif de l'atelier visait à relater et à faire vivre un dispositif de formation proposé en formation initiale à l'ESPE d'Angers. Ce dispositif de formation visait à faire réfléchir des étudiants de master 2 à la difficulté que pose la notion de variable informatique.

Modalités de ce dispositif :

Temps 1	Premières réflexions sur les potentielles difficultés de la variable informatique en cycle 4. Des échanges et une synthèse devaient clore ce temps.
Temps 2	Confrontation à une tâche proposée à des élèves de cycle 4.
Temps 3	Etudes des productions des élèves. Catégorisation
Temps 4	Eclairage sur le concept de variable informatique

Nous allons à présent revenir sur chacun de ces temps.

Quelles difficultés peut poser la variable informatique ?

Cette question constitue une entrée en matière afin d'interroger les conceptions initiales des stagiaires. Conformément à nos attentes, les réponses sont peu fournies. D'une part c'est une réflexion qu'ils ont rarement engagée, d'autre part ils ne voient pas "où est le problème". Les stagiaires sont ensuite invités à réfléchir sur une tâche proposée à des élèves de cycle 4.

Présentation de la tâche ‘Retour au port’¹ :

Depuis 2016, le thème Algorithmique et Programmation a fait son apparition dans les programmes de cycle 4 et *“entre dans le cadre d’un enseignement de l’informatique dispensé conjointement en mathématiques et en technologie”* (Eduscol, 2018). Cette nouveauté a amené les enseignants à réfléchir à des stratégies d’enseignement et à poser des questions pragmatiques essentielles. Quelles sont les connaissances à acquérir en cycle 4 ? Quelles tâches pour répondre à quels objectifs en termes de savoirs et ou de compétences ? Les ressources disponibles sont très nombreuses mais rarement accompagnées d’un éclairage didactique.

Par ailleurs, les injonctions institutionnelles, plutôt évasives laissent toute liberté aux enseignants quant aux modalités de mise en œuvre. Enfin même si un logiciel n’est pas spécifiquement nommé, les sujets de brevet proposent l’étude de scripts sous Scratch. Les enseignants sont ainsi de fait, incités à utiliser ce logiciel afin de familiariser leurs élèves avec cet environnement initialement dédié à la programmation créative.

Le concept de variable informatique : quels types de tâches pour favoriser son appropriation ?

Parmi les concepts à aborder au cours du cycle 4 figure celui de variable informatique, sans distinction quant à son type. Son apprentissage se heurte à la fois aux difficultés conceptuelles déjà identifiées par Samurçay et Rogalski dans les années 1980, et aux obstacles engendrés par les caractéristiques du milieu avec l’utilisation de l’environnement Scratch. Ainsi le choix des tâches à proposer aux élèves s’avère primordial afin de permettre l’appropriation de cette caractéristique fondamentale : une variable informatique mémorise un état passé dont dépend l’exécution future d’un programme. A ce titre les tâches type “programmes de calcul” ne nous paraissent pas être des contextes pertinents : ils nécessitent souvent plusieurs variables et leurs résolutions apparaissent bien plus aisées sur tableur. Nous faisons l’hypothèse que restreindre le champ d’utilisation des variables à ce type de tâche montre un intérêt appauvri du concept de variable. La situation “Retour au port” créée par le groupe Algorithmique et programmation au cycle 4 de l’IREM de Nantes présente selon nous des caractéristiques permettant de s’appropriier le concept de variable informatique.

1 Lien de téléchargement : <https://scratch.mit.edu/projects/315299642>

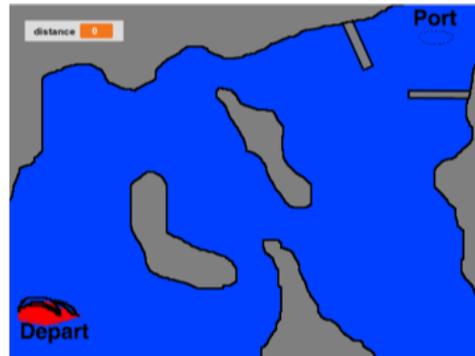
Analyse de la tâche proposée

Consigne élève :

Défi 1 : Trouver le plus court chemin pour permettre au bateau de rejoindre son emplacement dans le port.

Indice : modifier le programme pour qu'il calcule distance parcourue par le bateau.

Défi 2 : Que se passe-t-il si le bateau heurte un rocher ? Modifier le programme pour empêcher un bateau coulé de continuer à avancer.



Eléments concernant la consigne :

- Il s'agit donc de résoudre un problème d'optimisation : amener le lutin en un minimum de déplacement jusqu'à son emplacement dans le port, matérialisé par des pointillés.
- Dans cette tâche certains éléments du script sont déjà donnés à dessein. Tout d'abord cela permet une entrée directe dans le problème. L'élève devra concentrer son attention sur la manière dont il devra rendre compte de la distance parcourue. Par ailleurs s'appropriier un programme déjà écrit est aussi une compétence qu'il s'agit de développer. Cela permet aussi à tous les élèves de partir de la même base. Enfin la présence d'une boucle "répéter indéfiniment" est conceptuellement difficile à mobiliser chez des élèves. Ainsi donner des éléments du programme permet de recentrer l'élève sur la tâche à réaliser.
- Une variable distance a aussi été créée mais décochée pour donner l'idée aux élèves d'utiliser la variable.
- La consigne est donnée sous la forme de défis de manière à favoriser le processus de dévolution. La question de la preuve n'est pas attendue par les élèves. Elle sera débattue en plénière.
- L'élève alternera les postures de programmeur et d'utilisateur.

Quels sont les savoirs visés ?

La notion essentielle est celle de variable. En effet les élèves doivent rendre compte de la distance parcourue afin de pouvoir comparer avec leurs camarades et assurer ainsi qu'ils ont la plus petite distance. Il faudra donc mémoriser à tout moment la distance parcourue. Pour cela seule une variable de type numérique pourra répondre à cette question. Il s'agit dans ce cas d'une variable accumulateur.

Cette tâche donne aussi l'opportunité d'introduire un autre type de variable : les variables d'état afin de permettre de rendre compte de l'état du bateau et répondre ainsi à la 2^{ème} question.

A la mobilisation d'une variable s'ajoutent bien sûr son initialisation, son insertion dans le script et la prise en compte de ses variations.

Public visé :

Cette tâche est destinée à des élèves de cycle 4 (milieu ou fin de cycle). En effet les élèves doivent avoir été familiarisés avec l'environnement de Scratch. De plus des prérequis sont indispensables comme la notion de déplacement, la notion d'instruction conditionnelle, et la notion de variable.

Difficultés prévisibles :

Certaines difficultés sont prévisibles :

- D'ordre conceptuel : la résistance à la variable, la capacité à penser autrement le problème, de manière analogue à un problème mathématique où l'on mobilise le calcul algébrique.
- D'ordre syntaxique : les difficultés liées à l'insertion des instructions et à l'initialisation de la variable.
- D'autre environnemental : les difficultés liées à l'environnement scratch avec tous ses implicites et ses variables cachées.

Il est à noter que la résolution de la tâche a posé des difficultés à certains stagiaires peu à l'aise avec les notions de programmation et/ ou l'environnement de la programmation par blocs.

Etudes des productions des élèves - Catégorisation

Nous proposons ensuite d'étudier des productions d'élèves confrontés à la même tâche. En effet cette situation a été proposée dans une classe de troisième. Lors des deux heures qui lui ont été dédiées, des captures d'écran ont été réalisées en cours de séance afin de rendre compte du cheminement des élèves dans la résolution des deux défis.

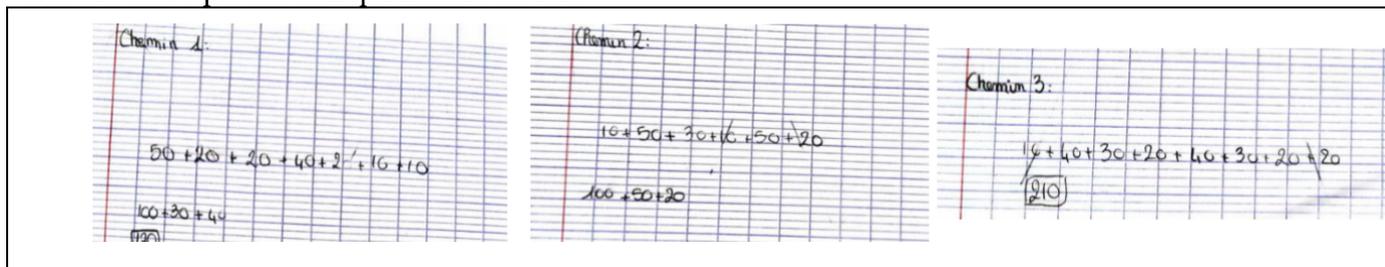
La diversité des réponses apportées a permis aux stagiaires de découvrir des raisonnements autres qu'ils n'avaient pas envisagés mais surtout les a confrontés aux difficultés multiples des élèves dans l'utilisation de variables informatiques.

A l'issue de ce temps une classification des productions a été proposée.

- Résistance à la variable (production Y)

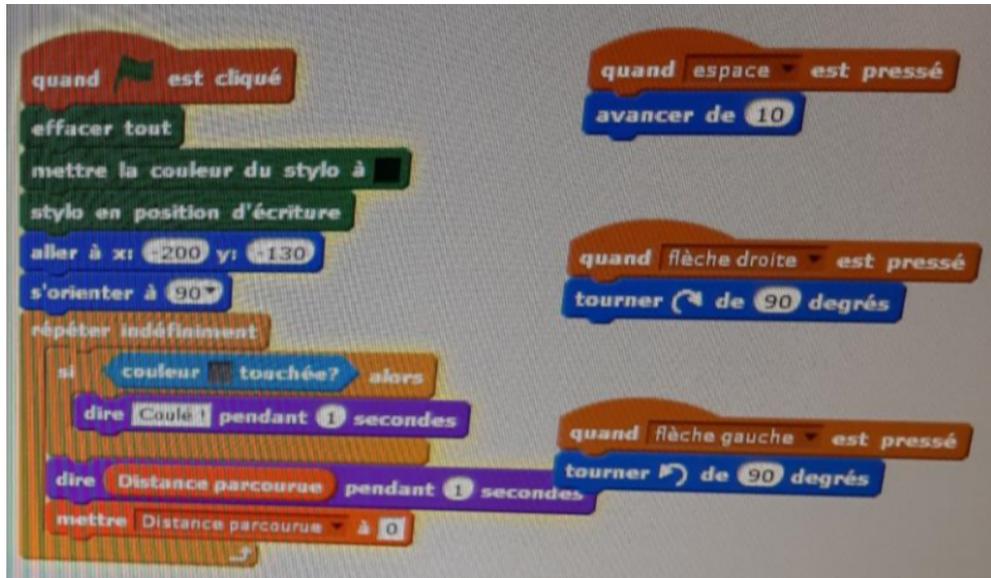
Les élèves ne mobilisent pas de variable. Ils essaient de construire le trajet le plus court par un raisonnement essais-erreurs. Cela illustre la difficulté à penser autrement le problème.

Voici une autre production qui illustre ce fait.



- Insertion dans le script des instructions relatives à la variable permettant de rendre compte de son initialisation ou de sa modification (Productions N et A)

Les difficultés à insérer correctement les instructions relatives à l'incrémentation de la variable, ou de son initialisation sont récurrentes.



- Le problème du nom donné aux variables : nom mal choisi car source de confusion ou le nom induit la fonction (Productions K, H, L et E)

Le fait de nommer une variable constitue une difficulté majeure. Le nom attribué peut prêter à confusion. Quelquefois pour certains élèves le nom induit la fonction. Ainsi pour calculer la distance parcourue, un élève a créé un bloc "calculer le nombre de pixels" sans mettre d'instruction ensuite.

- La gestion des variables booléennes (Productions D et I)

Par exemple la création d'une variable booléenne nommée vrai plonge assurément l'élève dans la confusion la plus totale.

- Les difficultés liées à l'environnement Scratch (Productions M et O)

La difficulté à comprendre l'impact de certaines instructions sur un script. Par exemple l'instruction "cacher" rend invisible le lutin mais celui-ci peut continuer d'avancer.

L'étude de ces productions a ainsi permis de provoquer un questionnement chez les professeurs stagiaires et ainsi d'amorcer une réflexion sur l'enseignement de concepts informatiques.

Eclairages sur le concept de variable (Sajaniemi, 2005 ; Lagrange & Rogalski, 2017)

Outre la variable de type compteur/accumulateur mobilisée dans cette situation on peut citer d'autres types de variables.

- La variable fixée (fixed value)

C'est le cas le plus simple de variable, dont la valeur est fixée une fois pour toutes dès le début de l'exécution d'un programme ou dès que la question est posée à l'utilisateur.

Par exemple, un programme qui demande à son utilisateur la taille ou le nombre de côtés du polygone à dessiner mobilise une variable fixée.

Dans ce cas, le concept de variable correspond à celui de constante ou de paramètre en mathématiques

- La variable à affectation unique :

C'est le cas de situations - de calculs par exemple - dont l'énoncé fournit les formules permettant de calculer différentes grandeurs. La programmation de la solution implique alors de rédiger une affectation par variable permettant de calculer les valeurs des différentes grandeurs, puis de prévoir de séquencer ces affectations dans un ordre permettant leur calcul effectif. C'est le cas par exemple dans un programme calculant en fonction de a , b , c les coefficients d'une équation du 2nd degré, les valeurs de Δ , le discriminant et de x_1 , x_2 les éventuelles solutions... Ce cas de variable informatique est le plus proche du concept de variable en mathématiques.

Dans le sens le plus général, la variable est nécessaire pour mémoriser l'état d'exécution d'un programme. Le concept de variable est alors nécessité par le fait de devoir "mémoriser" des informations sur le comportement passé d'un programme, dont dépend l'exécution future de ce programme.

Conclusion

- Concernant la tâche proposée

Son analyse révèle des potentialités certaines. Sa résolution contraint la mobilisation de la variable informatique et oblige les élèves ainsi à se risquer à son utilisation.

Comme dans un problème ouvert en mathématiques où la construction du problème est aussi importante, voire plus, que sa solution, la confrontation des élèves à cette tâche a permis de mettre à jour des conceptions erronées, que certains ont pu dépasser.

Enfin cette situation conforte l'idée que la notion de variable informatique et plus généralement la construction des concepts informatiques (séquence, itération, variable, condition ...) posent problème et nécessitent d'être pensés

- Concernant le dispositif de formation.

La réflexion engagée a amené à questionner le choix des situations de références pour cibler des savoirs clés (séquence, boucle, instructions conditionnelles...). Il nous semble utile et nécessaire de poursuivre des travaux dans ce sens.

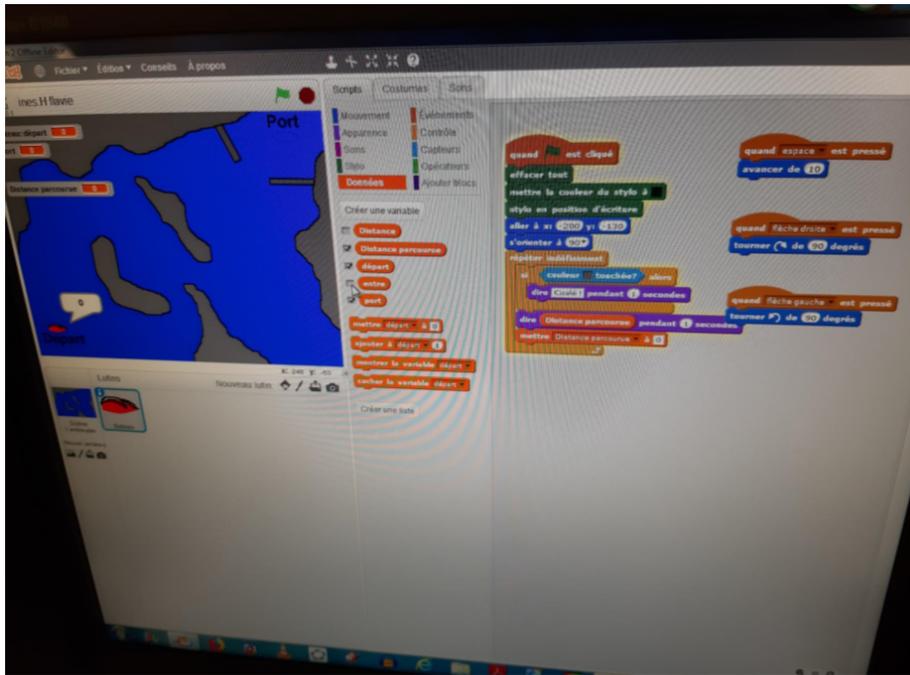
La question de l'évaluation dans le domaine de la programmation nous paraît aussi critique. Les compétences identifiées par Selby et Woollard (2013) dans la construction de la pensée algorithmique chez les élèves Evaluer, Anticiper, Décomposer, Généraliser, Abstraire, constituent un cadre d'analyse qui permettrait d'élaborer des échelles descriptives pour catégoriser les productions d'élèves. Ce travail sur les compétences pourrait ainsi compléter le dispositif de formation.

Bibliographie :

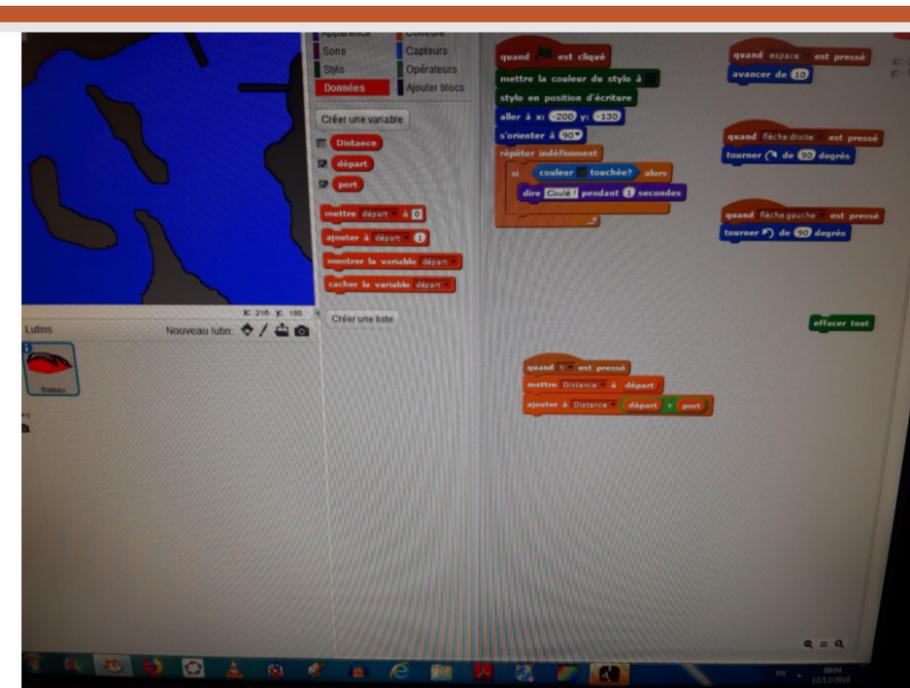
Briant, N. 2013. « Étude didactique de la reprise de l'algèbre par l'introduction de l'algorithmique au niveau de la classe de seconde du lycée français ». Thèse, Université Montpellier II - Sciences et Techniques du Languedoc. <https://tel.archives-ouvertes.fr/tel00920506>.

- Lagrange, J-B, Rogalski, J (2017). Savoirs, concepts et situations dans les premiers apprentissages en programmation et en algorithmique. *Annales de didactique et de sciences cognitives*, 22, pp : 119-158
- Sajaniemi, J. 2005. « Roles of Variables and Learning to Program ». In Proceedings of the 3rd Panhellenic Conference « Didactics of Informatics ». Korinthos, Greece.
- Samurçay, R. 1985. « Signification et fonctionnement du concept de variable informatique chez des élèves débutants (The Concept of Variable in Programming: Its Meaning and Use in Problem-Solving) ». *Educational Studies in Mathematics* 16 (2): 143-161.
- Selby, Cynthia and Woollard, John (2013) *Computational thinking: the developing definition* University of Southampton (E-prints) 6pp.
<http://eprints.soton.ac.uk/id/eprint/356481>

Production A :



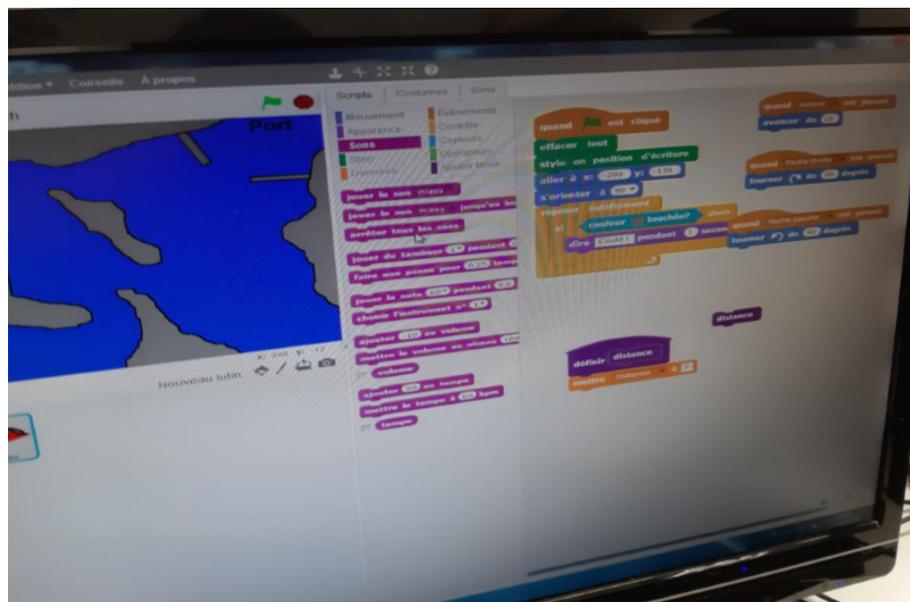
Production K :



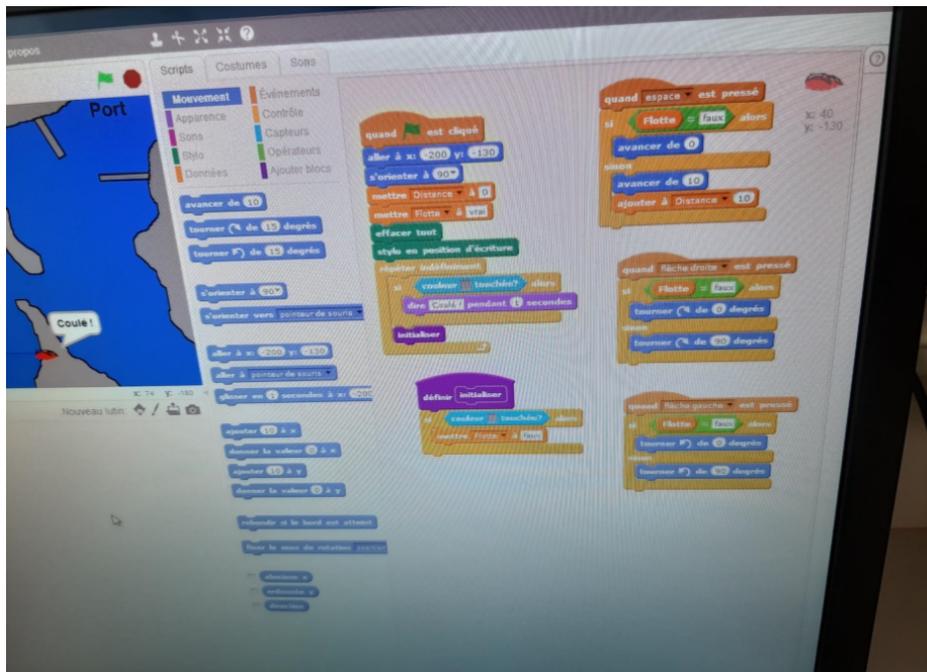
Production H



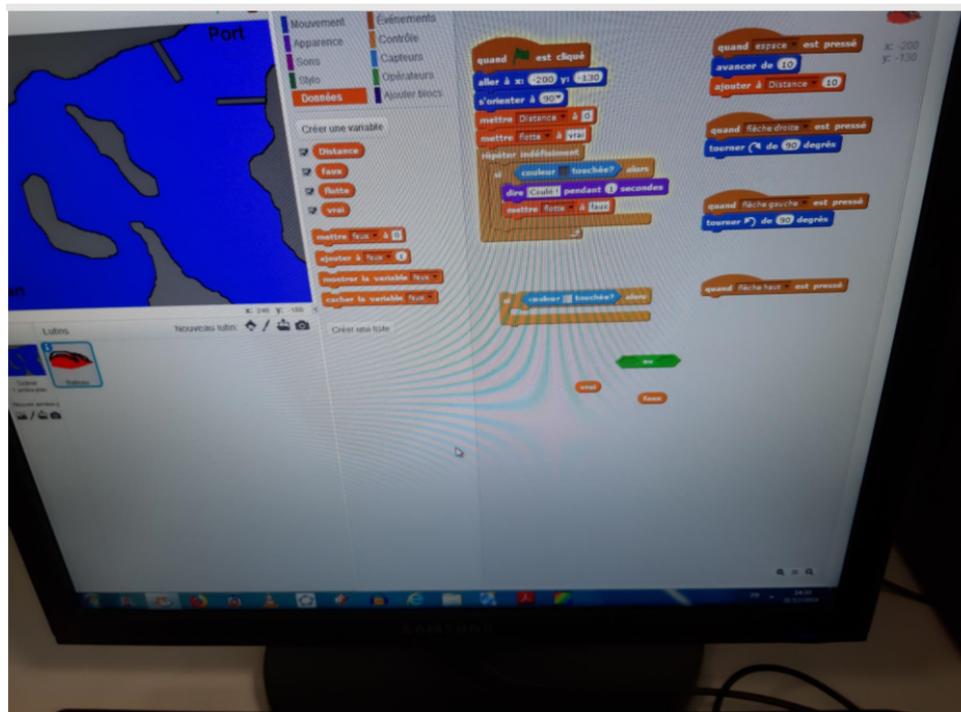
Production L :



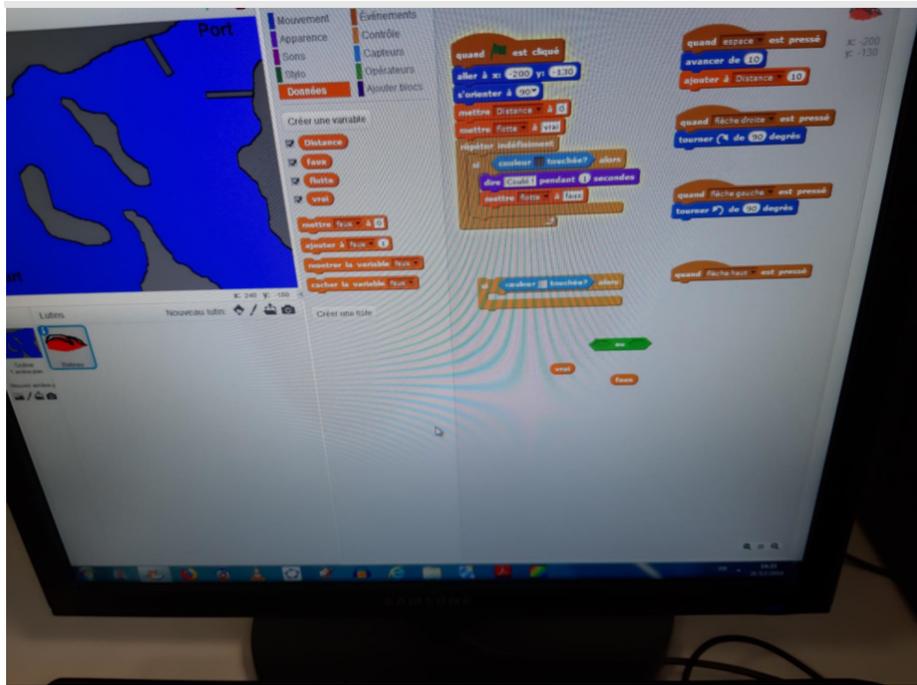
Production E :



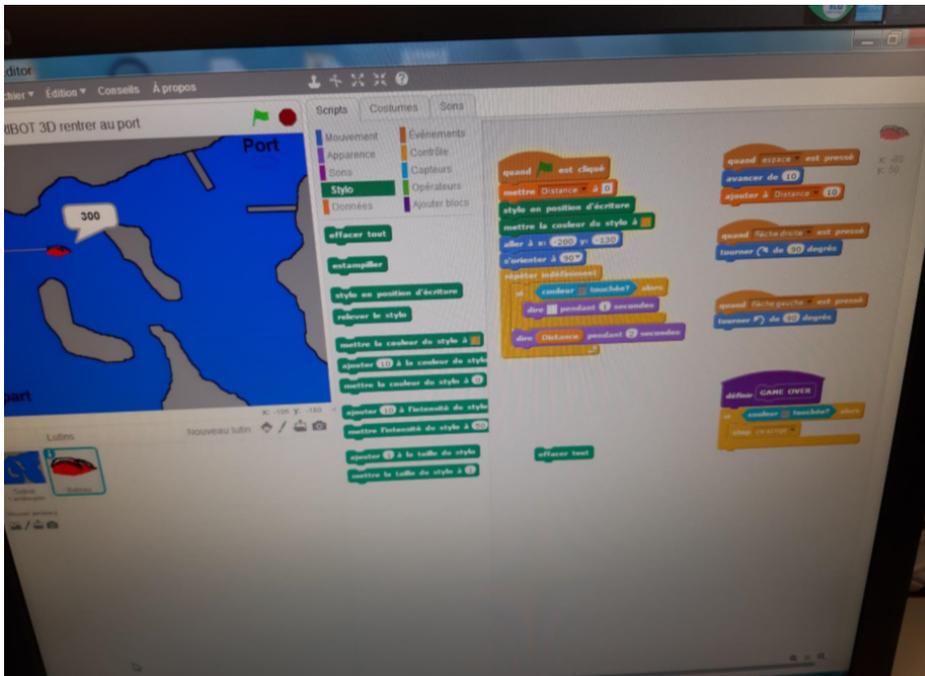
Production D :



Production I :



Production M



Production O :

