

Apprentissage de la programmation, contexte numérique des enseignants d'informatique genevois

Eva Dechaux¹[0000-0001-8579-1398] and Sébastien Jolivet²[0000-0003-3915-8465]

¹ IUFE, Université de Genève, Suisse

² IUFE & TECFA, Université de Genève, Suisse
eva.dechaux@unige.ch ; sebastien.jolivet@unige.ch

Abstract. Cet article s'inscrit dans le contexte d'une étude compréhensive des situations d'enseignement de l'informatique. Plus précisément nous identifions les différents contextes numériques (langage étudié ; IDE ; environnement numérique d'apprentissage) existants dans l'enseignement secondaire genevois, quelques usages associés et certains des facteurs ayant mené à leur mise en place. Cette étude est basée sur l'analyse d'une enquête réalisée auprès des enseignants d'informatique du secondaire du canton de Genève. Nous mettons en évidence la diversité des contextes numériques existants et analysons les différents facteurs menant à leur construction.

Keywords: apprentissage de la programmation, contexte numérique, enquête, secondaire 2 à Genève.

1 Introduction et problématique

Pour favoriser les premiers apprentissages en lien avec la science informatique au primaire et au début de l'enseignement secondaire (élèves de 3 à 14 ans), en particulier autour de la programmation, diverses modalités ont été développées et promues ces dernières années allant de l'informatique dite débranchée à l'utilisation de robots et cela a notamment donné lieu au développement d'environnements de programmation par blocs de type Scratch. Dans le canton de Genève, lors du passage au secondaire II (élèves de 15 à 18 ans), il est plutôt de mise d'utiliser des langages qui sont effectivement exploités dans le monde professionnel. Si les premiers apprentissages sont généralement réalisés dans le paradigme de la programmation impérative, cela ne détermine pas de manière univoque les choix de langage de programmation et d'environnements associés. Dans le canton de Genève, et plus largement en Suisse Romande, il n'y a pas de préconisations officielles relatives au choix du langage, ni même d'orientation indiquée (voir par exemple dans les curriculums produits par la Direction Générale de l'Enseignement Secondaire II (DGES II, 2021)). De plus, l'absence d'une épreuve de fin d'études commune à l'ensemble des établissements, favorise une grande liberté dans les options choisies. Il peut donc y avoir des différences significatives d'un établissement à l'autre, et parfois même entre enseignants au sein d'un établissement. Or, comme cela est étudié par exemple dans (Branthôme,

2022; Khazaei & Jackson, 2002; White & Sivitanides, 2005) le choix du langage de programmation et/ou du paradigme n'est pas sans impact sur les apprentissages, il est donc pertinent de dresser un état des lieux de l'existant.

Si dans cet article nous ne proposons pas une étude permettant de comparer la situation entre différents cantons ou pays, il est intéressant de noter qu'en France le choix concernant les langages est beaucoup moins libre. Ainsi, les choix de Scratch (langage utilisé notamment aux épreuves du brevet des collèges français en mathématiques) pour le secondaire 1, et de Python pour le secondaire 2¹, sont quasi imposés.

En plus de celui du ou des langages travaillés, d'autres choix se posent aux enseignants ; en particulier celui du ou des environnements de développement intégré (IDE) et d'exploiter un environnement numérique d'apprentissage (ENA) de la programmation. Dans la suite de l'article nous appellerons *contexte numérique d'apprentissage de la programmation*, noté *contexte numérique* par simplicité, l'ensemble constitué du ou des langages, du ou des IDE, du ou des ENA, exploités par un enseignant. Le contexte numérique global d'un enseignant comprend potentiellement aussi d'autres éléments (ENT, système de pilotage de salle...) que nous ne prenons pas en compte dans ce travail car non spécifiquement liés à l'apprentissage de la programmation.

La description des choix réalisés par les enseignants dans le canton de Genève est donc le premier objectif de ce travail et donne lieu à la question :

- (QR1) Dans le contexte genevois, qui est propice à l'existence d'une diversité, quels sont les choix des enseignants en matière de langage, IDE et ENA ? Quels sont les usages associés ?

Par ailleurs, dans (Vandeput & Henry, 2018) les auteurs pointent le manque de réflexion didactique des enseignants dans le choix des outils. Qu'en est-il dans le canton de Genève ? Les déterminants des choix sont-ils les mêmes pour le langage, l'IDE et les ENA ? Ces déterminants sont-ils plutôt pragmatiques, fonctionnels ou didactiques ? Par pragmatique nous entendons les facteurs du type choix imposé au niveau de l'établissement, seul environnement installé, etc. ; par fonctionnel nous entendons les facteurs liés par exemple aux modalités d'accès (en ligne ou nécessitant une installation), la langue de l'environnement, etc ; par didactique nous entendons les facteurs liés à un effet, au moins supposé, sur l'apprentissage.

Ceci nous amène à notre deuxième question :

- (QR2) Quels facteurs influent sur la construction par les enseignants de leur contexte numérique d'apprentissage ?

Pour dresser cet état des lieux, nous avons commencé par construire un inventaire des ressources existantes puis nous avons diffusé un questionnaire à destination des enseignants d'informatique du canton de Genève pour identifier choix et facteurs

¹ Les programmes de SNT et de NSI (lycée français) stipulent qu'il faut utiliser "un langage simple d'usage, interprété, concis, libre et gratuit, multiplateforme, largement répandu, riche de bibliothèques adaptées et bénéficiant d'une vaste communauté d'auteurs dans le monde éducatif. (...) le langage retenu est Python (...)" (MEN, 2021, p. 3)

influent sur ces choix. Dans cet article nous présentons le questionnaire et l'analyse des réponses. Pour conclure, nous discutons les premiers résultats obtenus et proposons quelques prolongements à cette étude.

2 Méthode

Pour obtenir des réponses à nos deux questions, nous avons réalisé une enquête en direction des enseignants d'informatique du secondaire 2, du canton de Genève. Cette enquête a pris la forme d'un questionnaire LimeSurvey diffusé en ligne. Pour construire ce questionnaire, afin d'investiguer la QR1, nous avons tout d'abord construit des listes de langages, IDE et ENA. Ces listes ont été élaborées à l'aide de nos observations de terrain, notre expertise du domaine et la littérature professionnelle (de nombreux sites proposent des listes, selon différents critères, d'IDE pour tel ou tel langage). La présence systématique d'une réponse « autre », avec champ à compléter, permet de pallier d'éventuels manques. Pour disposer de quelques éléments permettant une compréhension plus fine qu'une simple liste d'éléments nous avons aussi intégré des questions permettant d'obtenir des informations sur les usages associés (fréquence, modalités, etc.).

Pour obtenir des éléments de réponse à la QR2, nous avons associé, à chaque question relative au choix d'un élément du contexte numérique, une question sur les facteurs qui ont orienté ce choix. Pour faciliter le traitement des réponses nous avons proposé des questions à choix multiples dans une liste plutôt que des réponses ouvertes. Pour contraindre les répondants à se positionner nous avons limité le nombre de choix possibles à trois. Les listes de réponses proposées permettent d'exprimer des raisons pragmatiques (choix de l'établissement), fonctionnelles (multiplateforme ou accès en ligne par exemple) ou didactiques (présence d'une fonctionnalité particulière). A nouveau la réponse « autre » permet d'étendre les choix proposés *a priori*. Le questionnaire est trop long pour être présenté intégralement ici mais est disponible en ligne en format PDF ([lien d'accès au questionnaire](#)).

Pour des raisons de pratiques institutionnelles propres à l'organisation de l'enseignement scolaire secondaire genevois, le questionnaire n'a pas été diffusé directement à l'ensemble des enseignants mais aux responsables de la discipline informatique dans les 11 établissements du secondaire 2 à Genève, avec la demande de relayer aux enseignants d'informatique de leur établissement. Nous n'avons pas accès à l'effectif exact mais en estimant à une moyenne de 5 enseignants par établissement cela représente environ 55 répondants potentiels.

Nous avons obtenu 24 réponses dont 13 complètes. L'anonymat et la méthode de diffusion du questionnaire ne nous permettant pas de savoir si les réponses partielles étaient redondantes ou non avec des réponses complètes nous n'avons traité que ces dernières. Nous présentons le résultat de ce traitement dans la section suivante.

3 Résultats

Nous présentons dans un premier temps les éléments permettant de répondre à notre QR1 en décrivant les divers contextes numériques identifiés dans les réponses et quelques éléments sur les usages associés. La section suivante permet de revenir sur la QR2 au travers de l'analyse des critères de choix menant à la construction de ces contextes numériques.

3.1 Contextes numériques et usages

Concernant les langages, tous les enseignants qui ont répondu déclarent utiliser Python. Pour sept d'entre eux c'est le seul travaillé, alors que pour les six autres, au moins un autre langage est utilisé durant l'année. Dans ce cas-là il s'agit systématiquement d'utilisations successives, aucun ne déclare travailler deux langages en parallèle.

Aucune utilisation de C, C++, Java ou TygerJython n'est signalée. Nos observations sur le terrain nous ont cependant permis de constater des utilisations de TygerJython. Ce sont donc au moins sept langages différents qui sont utilisés (**Figure 1**).

Pour les IDE, Thonny² et Replit³ sont les deux IDE très majoritairement utilisés avec les élèves (resp. 11 et 10 utilisateurs dont huit qui utilisent les deux). Le seul autre IDE, cité dans une réponse, est Access, en complément avec Replit et Thonny.

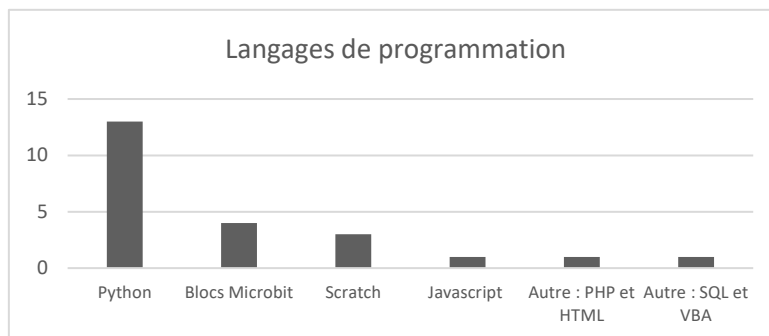


Fig. 1. Langages de programmation utilisés en classe (effectifs d'utilisateurs en ordonnée)

En complément du ou des couples (langage de programmation ; IDE), neuf enseignants déclarent utiliser un environnement numérique d'apprentissage avec leurs élèves en classe (question C2). Dans cette question, les enseignants devaient associer à chaque ENA une fréquence d'utilisation évaluée en nombre de périodes sur l'année. La **Figure 2** présente les résultats de cette question. Sur les onze ENA que nous avons proposés comme réponses possibles, sept sont utilisés par au moins deux enseignants et quatre par aucun (question C2). Seul AlgoPython est déclaré comme faisant l'objet

² <https://thonny.org/>

³ <https://replit.com/site/ide>

d'un usage fréquent (deux enseignants), quatre ne le sont que très occasionnellement (de 1 à 5 fois dans l'année). Sur cette évaluation de la fréquence, on peut noter un certain décalage entre les réponses à la question C2 et celles obtenues à la question C8 qui demandait aux enseignants de préciser si l'utilisation de l'ENA, lors du travail sur la programmation est occasionnelle, régulière ou systématique puisque pour cette question les effectifs pour chaque alternative sont respectivement 5 ; 3 et 1.

Quand il y a un ENA utilisé, en complément de l'évaluation quantitative de sa place, nous avons questionné les intentions que les enseignants associent à leur usage. L'ENA est essentiellement un élément auxiliaire dans la pratique de l'enseignant, mais il est intégré au travail du savoir. Plus précisément, on distingue deux intentions principales attribuées aux ENA : *faire découvrir certaines notions* (six sur neuf) et *permettre aux élèves de s'entraîner sur des notions déjà travaillées* (sept sur neuf). Un enseignant a répondu que toutes les notions sont travaillées au travers de l'ENA, tandis que deux d'entre eux précisent qu'il ne s'agit que d'une activité visant à occuper les élèves les plus rapides, qui n'est pas exploitée ultérieurement.

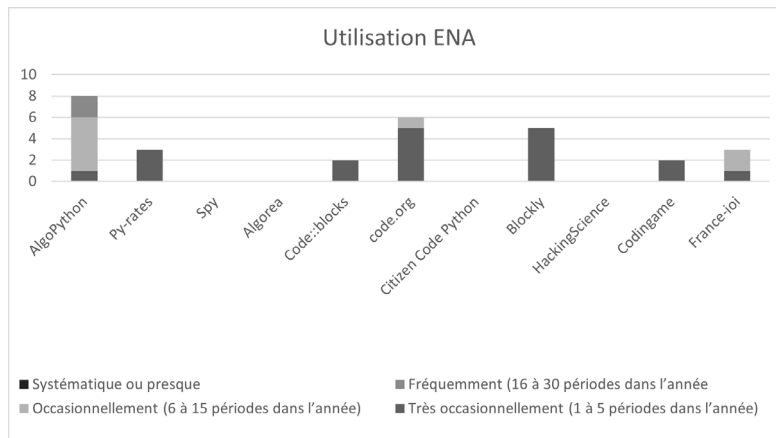


Fig. 2. ENA utilisés et fréquence d'utilisation

On relève également que personne n'utilise l'ENA dans un but évaluatif. La question formulée ne distinguait pas l'évaluation purement formative et l'évaluation à visée sommative, il est donc possible que ce soit cette deuxième forme qui ait été considérée pour formuler les réponses, il y a sinon une certaine contradiction avec la volonté annoncée de suivre le travail des élèves.

Cependant, les pratiques sont plus partagées concernant la place de l'ENA par rapport à l'IDE. Ainsi trois enseignants utilisent en alternance IDE et ENA ; cinq d'abord l'ENA puis l'IDE, ce qui confère une fonction plutôt introductive à l'ENA, et un enseignant qui les présente comme indépendants.

3.2 Facteurs impactant la construction des contextes numériques

Le premier élément définissant le contexte numérique est le langage de programmation. Celui-ci influence largement ceux des IDE et des ENA. Pour le langage il s'agit, pour la totalité des enseignants, de se conformer à des contraintes externes (choix d'équipe, épreuves communes ou contraintes matérielles), pour moins de la moitié d'entre eux, cinq sur treize, ces contraintes rejoignent leur choix personnel.

On constate aussi que, bien que le choix ne soit pas personnel, la grande majorité se dit satisfaite ou très satisfaite du choix puisque seuls trois enseignants se disent peu ou très peu satisfaits du choix de Python. Même s'il est majoritairement positif, le regard porté sur l'utilisation de Python n'est pas exempt de critique. Deux limites sont pointées, la première est la difficulté de rendre les programmes attrayants pour les élèves contrairement à un langage de programmation plus visuel (cette limite est parfois contournée par l'utilisation de la bibliothèque Pygame qui permet de disposer d'un environnement graphique riche). La deuxième limite signalée est relative aux éléments de programmation masqués par un langage de haut niveau comme Python, notamment sur les types des variables. *A contrario*, pour d'autres, cet aspect est cité comme un avantage, car cela rend Python plus accessible pour les élèves. Enfin, l'importante communauté d'utilisateurs et les nombreuses ressources à disposition sont des points positifs de Python selon plusieurs enseignants.

Table 1 : facteurs guidant le choix de l'IDE

Facteur	Effectif
Facilité d'accès dans l'établissement	8
Facilité d'utilisation (sauvegarde)	8
Facilité accès hors de la classe	7
Utilisation hors ligne	4
Exécution pas-à-pas	2
Choix imposé par l'établissement	1
Possibilité de paramétrer la langue de l'IDE	1
Paramétrer l'IDE	1
Autre : possibilité d'installer nouveaux packages	1
Autre : transition directe entre la classe et la maison	1

Si le premier facteur de choix de l'IDE est évidemment sa compatibilité avec le langage travaillé, leur diversité⁴ nécessite de réaliser un ou des choix. Dans la table 1, nous présentons les différents facteurs de choix relatifs aux IDE, par ordre décroissant d'importance relativement au nombre de fois où ils ont été cités. On peut noter qu'il s'agit essentiellement de facteurs liés à des aspects fonctionnels (accès, sauvegarde...)

⁴ Par exemple sur cette page <https://www.commentcoder.com/ide-python/> il y a 11 IDE identifiés pour développer en Python

et que les dimensions qui pourraient donner lieu à une exploitation didactique (exécution pas-à-pas, paramétrages de l'IDE) ne sont citées que de manière marginale.

Concernant les raisons exprimées pour le choix d'un ENA il y a, à nouveau, une diversité importante comme le montre la Table 2 (trois réponses possibles au maximum, non ordonnées). Celle qui est identifiée comme importante par le plus grand nombre d'enseignants (sept sur neuf) est la *possibilité de suivre le travail des élèves*. Il serait intéressant de pouvoir mettre en relation cette déclaration avec l'exploitation des outils de suivi de l'activité proposés par les ENA ; est-elle effective ou non, selon quelles modalités ?

Table 2 : facteurs guidant le choix de l'ENA

Facteur	Effectif
Possibilité de suivi pour l'enseignant (tableau de bord...)	7
La scénarisation de l'apprentissage proposée par l'ENA (ordre des notions...)	4
Langue(s) de l'environnement	3
Présences de rétroactions	3
ENA accessible hors la classe	3
Type d'environnement : jeu sérieux (SG), exerciceur...	2
Présences d'aides (rappels de cours, exemples...)	2
Accès en ligne	2
Possibilité de paramétrer (parcours, accès conditionnel à des ressources)	1
Possibilité d'installation locale	0
Protection des données personnelles	0

Même si la scénarisation de l'apprentissage proposée par l'ENA arrive en deuxième position avec 4 réponses, il est notable qu'aucun des facteurs que l'on peut considérer comme relatifs à la place des savoirs dans l'ENA (scénarisation de l'apprentissage portée par l'ENA ; présences de rétroactions et/ou d'aides) n'est majoritaire. A la différence du choix de l'IDE, ce sont plutôt des facteurs de nature didactique qui semblent guider le choix des ENA.

Enfin, on pourra s'interroger sur le fait que la "protection des données personnelles" n'est citée comme élément important pour le choix d'un ENA par aucun enseignant. Ceci questionne à la fois le rapport que les enseignants entretiennent à cette question, et le lien qu'ils opèrent ou non entre leurs pratiques et l'enseignement qu'ils délivrent puisque la question des données personnelles fait partie des éléments auxquels ils doivent sensibiliser leurs élèves.

3.3 Quelques éléments de synthèse sur les réponses à QR1 et QR2

Le constat qui synthétise l'ensemble de ces résultats est celui de *diversité*.

- Diversité importante des contextes (langage, IDE, ENA) qui définissent les conditions d'apprentissage de la programmation pour les élèves, même si Python est un incontournable⁵, articulé ou non avec d'autres éléments.
- Diversité des facteurs qui influencent la construction de ces contextes. On peut cependant noter la place relativement faible accordée à des éléments qui relèvent du didactique dans les réponses recueillies, comme si cette dimension était une préoccupation indépendante du contexte numérique d'exercice.
- Diversité des usages déclarés, notamment sur la place et les objectifs associés aux ENA.

4 Conclusion et perspectives

Dans cette contribution, nous avons dressé un panorama des contextes numériques créés par les enseignants dans le canton de Genève pour l'apprentissage de la programmation. Nous nous inscrivons dans une approche compréhensive des conditions d'enseignement de l'informatique au niveau scolaire, notamment dans le prolongement du travail présenté dans (Jolivet et al., à paraître) qui se focalise sur la formation initiale des enseignants d'informatique en Suisse Romande.

Cette première enquête met en évidence une grande diversité des contextes numériques d'enseignement, tant pour les moyens utilisés que pour les facteurs qui influent sur les choix des enseignants. Même si le nombre de réponses récoltées est relativement modeste, nous faisons l'hypothèse qu'il est assez représentatif de la réalité genevoise, au regard du nombre d'établissements et d'enseignants. Si un nombre plus important de réponses avait sans doute permis de faire émerger encore quelques facteurs supplémentaires de diversité le constat ne serait pas remis en cause. Cependant, il serait pertinent de pouvoir disposer d'un nombre plus important de réponses pour affiner l'importance relative des différents critères menant à la construction de ces contextes numériques.

Il est aussi possible d'enrichir l'enquête en intégrant les éléments liés à l'enseignement de la programmation qui ne sont pas pris en compte dans le contexte numérique, par exemple les temps réalisés en « mode débranché » et les moyens d'enseignement non numériques exploités (brochures, cahiers d'exercices, etc.).

Par ailleurs, dans le prolongement d'une telle enquête, il serait intéressant de pouvoir confronter les déclarations à la réalité des pratiques, par exemple au moyen d'observations *in situ*. L'accès au terrain que nous donne notre fonction de formateurs ne nous a pas permis d'identifier d'écarts évidents *a priori*, mais cela mériterait d'être confirmé.

D'autre part, cette étude a été menée dans un contexte institutionnel qui favorise l'existence de cette diversité et l'on constate que les enseignants s'en emparent. Il serait intéressant de l'élargir à d'autres institutions d'enseignement de la programmation, par exemple d'autres cantons suisses ou d'autres pays. Par exemple dans le con-

⁵ Ce résultat de l'enquête est sans doute à modérer un peu puisque nous avons, depuis la passation du questionnaire, identifié un établissement dans lequel l'apprentissage de la programmation est mené uniquement au travers du triplet JavaScript, HTML et CSS

texte du lycée français, le choix du langage de programmation est contraint mais il serait intéressant d'étudier si les autres éléments du contexte numérique sont variés. La description et la compréhension de ces différents contextes étant sans doute une base nécessaire, sans être exclusive, à la compréhension d'éventuelles différences observées sur l'apprentissage de l'informatique entre institutions. Il s'agirait ainsi d'aborder la question fondamentale de l'étude des effets sur les apprentissages de ces différents contextes.

Références

1. Branthôme, M. (2022). Pyrates : A Serious Game Designed to Support the Transition from Block-Based to Text-Based Programming. In I. Hilliger, P. J. Muñoz-Merino, T. De Laet, A. Ortega-Arranz, & T. Farrell (Éds.), *Educating for a New Future : Making Sense of Technology-Enhanced Learning Adoption* (Vol. 13450, p. 31-44). Springer International Publishing. https://doi.org/10.1007/978-3-031-16290-9_3
2. DGES II. (2021). *Programmes des disciplines enseignées dans la filière gymnasiale au Collège de Genève valable pour les élèves scolarisés dès 2021* (p. 160-165) [Programmes de la filière Gymnasiale]. <https://www.ge.ch/document/programmes-disciplines-enseignees-dans-filiere-gymnasiale-au-college-geneve-valable-eleves-scolarises-2021>
3. Jolivet, S., Wang, P., & Dechaux, E. (2024). Un retour d'expérience pour identifier constats et problématiques relatifs à la formation des enseignants d'informatique. *Revue RADIX, 1*.
4. Khazaei, B., & Jackson, M. (2002). Is there any difference in novice comprehension of a small program written in the event-driven and object-oriented styles? *Proceedings IEEE 2002 Symposia on Human Centric Computing Languages and Environments*, 19-26. <https://ieeexplore.ieee.org/abstract/document/1046336/>
5. MEN. (2021). *Programme de NSI de 1ère générale*. Bulletin Officiel de l'Éducation Nationale. <https://eduscol.education.fr/document/30007/download>
6. Vandeput, É., & Henry, J. (2018). Apprendre à programmer Comment les enseignants justifient-ils le choix d'un outil didactique ? In G. Parriaux, J.-P. Pellet, G.-L. Baron, & V. Komis (Éds.), *De 0 à 1 ou l'heure de l'informatique à l'école* (p. 325-342). Peter Lang. <https://library.oapen.org/bitstream/handle/20.500.12657/47395/9783034333245.pdf#page=327>
7. White, G., & Sivitánides, M. (2005). Cognitive Differences Between Procedural Programming and Object Oriented Programming. *Information Technology and Management*, 6(4), 333-350. <https://doi.org/10.1007/s10799-005-3899-2>