



La classe inversée. Exemple 2: Retour d'expérience sur une classe inversée en bio-informatique

Pierre Poulain

► To cite this version:

Pierre Poulain. La classe inversée. Exemple 2: Retour d'expérience sur une classe inversée en bio-informatique. Jean-François Parmentier, Quentin Vicens. Enseigner et former - 2e édition Psychologie appliquée et pédagogies actives, secondaire, supérieur, formation professionnelle, Dunod, 2022. hal-03737367

HAL Id: hal-03737367

<https://hal.science/hal-03737367>

Submitted on 24 Jul 2022

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

La classe inversée

Exemple 2 : Retour d'expérience sur une classe inversée en bio-informatique

Motivation

En septembre 2019, j'ai souhaité faire évoluer mon enseignement de la programmation Python en master de Bio-informatique (~30 étudiants), pour recentrer les sessions de cours sur les étudiants et rendre ces derniers plus actifs. J'ai profité d'un support de cours mature¹ pour mettre en place une classe inversée. J'ai bénéficié pour cela de l'accompagnement d'Antoinette Bouziane et Marine Lanteri, conseillères pédagogiques au service de pédagogie SAPIENS (Paris).

Mise en place et implémentation

Ce cours d'informatique est constitué de 30 h réparties en 7 séances de cours de 2 h et 8 séances de travaux pratiques 2 h sur un semestre. Ma classe inversée est basée sur un modèle de classe inversée de type I de Lebrun². Dans ce modèle, l'enseignant fournit l'essentiel du contenu pédagogique, les étudiants travaillent leur cours en dehors de la classe, puis vérifient et consolident leurs apprentissages en classe avec l'enseignant.

En amont du cours

Avant le cours, les étudiants lisent environ 3 chapitres du cours, disponibles sous la forme d'un polycopié papier ou du cours en ligne sur la plateforme Moodle de l'université. Cela représente au total une quinzaine de pages comprenant de nombreux exemples. Pour chaque chapitre, les étudiants réalisent une fiche mémo (manuscrite, au format A4, recto uniquement) qui résume les notions abordées. Ils déposent les photos de ces fiches sur la plateforme avant le cours. Ce travail préparatoire représente environ 1 h de travail personnel et permet aux étudiants de découvrir et se familiariser avec le nouveau contenu de la semaine. Il compte pour 15 % de la note finale.

En séance « débranchée »

Lors des séances de cours, les étudiants n'ont pas accès à un ordinateur, mais ils développent leurs compétences en programmation selon des modalités variées dont voici quelques exemples :

Retour sur le travail préparatoire

En début de séance, en binômes, ils échangent entre eux sur leurs fiches, les corrigent et les complètent (15 min). Pendant ce temps, je circule entre les binômes et réponds à leurs questions.

Quiz

J'utilise la plateforme en ligne WooClap pour proposer aux étudiants des QCM et des questions ouvertes auxquelles les étudiants répondent avec leurs téléphones. Les réponses aux questions ouvertes sont visualisées sous forme de nuage de mots. Pour chaque question, les réponses sont débattues avec toute la classe. À titre d'illustration, voici un exemple de question dans laquelle je demande aux étudiants ce que produit le petit script ci-dessous :

```
import sys
print(sys.argv[1:])
```

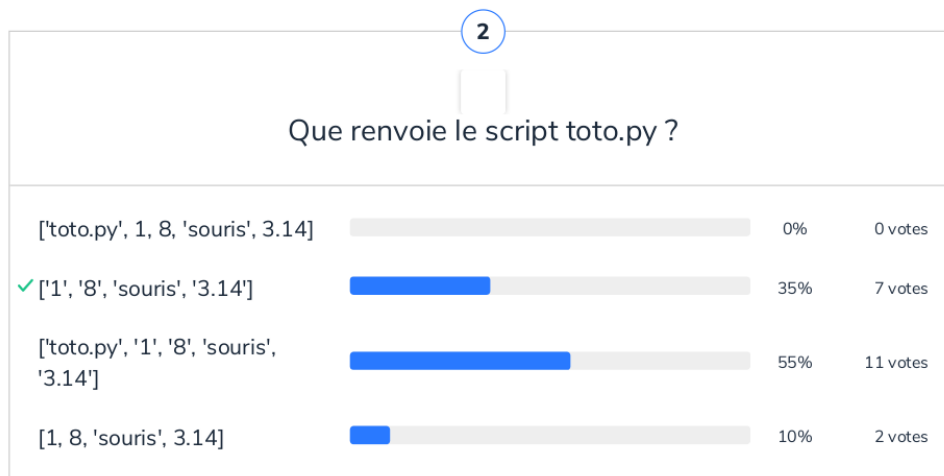
lancé avec la commande :

```
python toto.py 1 8 souris 3.14
```

¹ <https://python.sdv.univ-paris-diderot.fr/>

² M Lebrun, C Goffinet & C Gilson, Vers une typologie des classes inversées. Contribution à une typologie des classes inversées : éléments descriptifs de différents types, configurations pédagogiques et effets. Education & Formation, Vol. e-306, no.2, p. 125-146, 2016.

Les réponses proposées par les étudiants sont les suivantes :



Dans cet exemple, on remarque que la réponse correcte (la deuxième) n'est pas majoritairement trouvée par les étudiants. Une occasion parfaite pour discuter de la pertinence de chaque réponse et de corriger un apprentissage erroné sur ce point.

Scripts à trous

Partant d'un problème donné, je propose aux étudiants un programme Python que j'ai imprimé, mais pour lequel il manque certaines instructions. Les étudiants complètent alors les trous. Cet exercice demande aux étudiants de s'adapter à un contexte existant (le script à trous) pour répondre à un problème.

Question suivante

Les étudiants en groupe conçoivent une série de questions et réponses associées, portant sur des points du cours, avant de se défier entre eux. Pour motiver les étudiants, j'annonce au début de l'activité que les meilleures questions se trouveront dans le prochain examen de QCM et représenteront environ la moitié des questions.

Bilan

- Les fiches mémos déposées en amont du cours, la participation à des activités en séance de cours, la résolution de problèmes en cours (sans ordinateur) et en TP (avec ordinateur) me renseignent sur les progrès des étudiants dans leur apprentissage.
- La mise en place de cette classe inversée a nécessité d'être très explicite dans mes intentions, tant avec les étudiants (diffusion d'un syllabus au début du semestre, énoncé des objectifs pédagogiques à chaque séance) qu'avec moi-même (construction d'un scénario pédagogique).
- L'évaluation de l'enseignement par les étudiants montre que la majorité considère que la classe inversée les aide dans l'appropriation des concepts en programmation Python. L'interactivité en cours est aussi très appréciée, comme en témoigne un.e étudiant.e : « *Selon moi, le gros point fort est l'interactivité des cours, possible grâce à la pédagogie inversée, il faut absolument garder cette interactivité, car c'est ce qui rend ce cours passionnant. Les activités en groupe sont aussi à garder car c'est une façon de nous faire travailler de manière ludique sans que l'on ait l'impression de travailler, et surtout cela donne l'envie d'aller plus loin dans la programmation.* »

Exemple écrit par Pierre Poulain, enseignant-chercheur à l'Université de Paris.

Pour en savoir plus sur cet exemple : <https://cupnet.net/python-ci-2019/>