

Retour sur les ateliers numeriques

April 29, 2017

Auteur: Nathan KRUPA, *krupanatha@gmail.com*

1 Introduction

Afin de présenter brièvement le contexte je vais rapidement parler du projet et de moi. Les ateliers numériques étaient des cours donnés à des lycéens au lycée Jacques-Monod à Lescar par des élèves volontaires de l'EISTI (école d'Ingénieur informatique) et particulièrement de son association "geek" et réseau: Air-EISTI. Nous avons également le soutien de l'Association Française de Python (AFPy).

Je vais tenter, à travers ce document, de donner des conseils et des voies d'améliorations potentielles. J'ai participé à l'ensemble des séances et étais un élève d'ING2 à l'époque. Il s'agit là de mes idées personnelles pour la plupart, ce n'est donc pas la solution parfaite mais simplement des possibilités d'approfondissements pour ceux qui souhaitent reprendre le projet.

2 Les retours

2.1 La préparation

Dans un premier temps il ne faut pas négliger la phase de préparation. Au-delà de contacter des personnes intéressées (lycée, organisme, ...), des accompagnants motivés, un lieu, des créneaux et tout ce qui est absolument nécessaire. Il faut décider d'un sujet, le préparer en le décomposant en différents modules qui orienteront chaque séance. Si vous n'avez pas d'idées du niveau vous pouvez préparer quelque chose d'un peu plus complexe comme une interface graphique par exemple. Mais je pense qu'ils auront assez à faire!

2.2 Le sujet

Pour le choix du sujet il faut réussir à trouver quelque chose de ludique comme un jeu dont les règles sont connues. Ça évitera que la personne qui apprend l'algorithme et la programmation ait d'autres difficultés. Ensuite il faut réussir à le décomposer en différentes parties significatives: création du plateau ou de la grille, mise en place des règles, gestion des tours, interfaces, ... Cela dépendra de ce que vous choisirez.

Dans notre cas nous avons choisi de faire un démineur car nous jugions ça assez facile, connu, modulable, regroupant pas mal de choses; bref quelque chose de nous trouvions pas intéressant. Malheureusement compte tenu du temps que nous avons eu le sujet était trop compliqué. Nous avons vu peut-être un peu trop haut (ou le temps nous a manqué) mais il ne faut pas oublier

qu'au lycée même une simple boucle ou un petit bout d'algo est difficile, du moins plus que pour nous. Donc réussir à expliquer de la manière de résonner sur papier puis leur montrer comment faire en Python, le tout en peu de temps, c'est très compliqué.

2.3 Les séances

Ensuite par rapport aux séances il faut réussir à prévoir un objectif clair pour chacune. Pour notre part nous essayions au début de chaque séance de présenter le résultat attendu en réfléchissant ensemble à la manière de l'atteindre. Il faut arriver à trouver des résultats atteignables dans le temps imparti. Le but est que ce soit eux qui le fassent par eux-mêmes et non pas que ce soit nous qui le faisons à leur place... même si nous leur expliquons. C'est quelque chose que nous avons mal géré pour notre part et c'est facilement améliorable. Nous sommes là pour les aiguiller et leur apprendre des "astuces" de réflexion ou du langage mais pas le faire devant eux. Je trouve ça dommage, internet est rempli de tutoriels alors autant avoir une plus-value!

Je pense que la première séance devrait être consacrée à leur présenter le projet mais aussi les outils avec lesquels ils vont travailler et expliquer comment ils fonctionnent. Ce sont des choses basiques pour nous mais il s'agit d'une vraie découverte pour eux. Présenter un interpréteur, un langage de programmation, Python en général... Bref leur faire jouer sur des petites notions comme les variables, les types ou des choses plus utiles comme les *docstrings*, les fonctions **dir** ou **help** voire introduire PEP8 et des normes de programmation. Ils seraient comme ça plus autonomes pour la suite!

```
In [1]: a = 2 + 1
        "a = " + str(a)
```

```
Out[1]: 'a = 3'
```

```
In [2]: dir(a)
```

```
Out[2]: ['__abs__',
         '__add__',
         '__and__',
         '__class__',
         '__cmp__',
         '__coerce__',
         '__delattr__',
         '__div__',
         '__divmod__',
         '__doc__',
         '__float__',
         '__floordiv__',
         '__format__',
         '__getattribute__',
         '__getnewargs__',
         '__hash__',
         '__hex__',
         '__index__',
         '__init__',
         '__int__']
```

```

'__invert__',
'__long__',
'__lshift__',
'__mod__',
'__mul__',
'__neg__',
'__new__',
'__nonzero__',
'__oct__',
'__or__',
'__pos__',
'__pow__',
'__radd__',
'__rand__',
'__rdiv__',
'__rdivmod__',
'__reduce__',
'__reduce_ex__',
'__repr__',
'__rfloordiv__',
'__rlshift__',
'__rmod__',
'__rmul__',
'__ror__',
'__rpow__',
'__rrshift__',
'__rshift__',
'__rsub__',
'__rtruediv__',
'__rxor__',
'__setattr__',
'__sizeof__',
'__str__',
'__sub__',
'__subclasshook__',
'__truediv__',
'__trunc__',
'__xor__',
'bit_length',
'conjugate',
'denominator',
'imag',
'numerator',
'real']

```

In [52]: help(a.conjugate)

Help on built-in function conjugate:

```
conjugate(...)
    Returns self, the complex conjugate of any int.
```

```
In [3]: if __name__ == "__main__":
        print("Ceci est le main de mon programme !")
```

```
Ceci est le main de mon programme !
```

Par la suite le mieux est d'essayer de suivre le programme prévu en prenant compte du rythme. Je pense qu'une des plus grosses difficultés sera de gérer le temps. Pour ce qui est du nombre d'élèves par accompagnant nous étions presque de un pour un. Évidemment c'est l'idéal pour les deux parties! Pour autant je pense que 1 pour 2 ou 3 est tout à fait correct. Mais au-delà je pense que ça devient plus compliqué de faire un réel suivi particulier. Bien entendu tout dépend de ce que vous voulez transmettre, de vos possibilités: je parle ici du cas idéal!

Il ne faut pas hésiter non plus à prendre des séances afin de faire des mises au point, de détailler des parties pour savoir si elles ont été comprises voire maîtrisées. Il faut cependant avoir le temps pour ça ce qui n'est pas toujours possible. C'est à vous ensuite d'organiser vos séances comme vous le sentez! Il faut que vous sachiez ce que vous voulez faire et transmettre et quels sont vos objectifs? Vous pouvez orienter vos interventions plus vers l'algorithmie même simple (*ex: la récursivité*),

```
In [4]: def facto_rec(n):
        """ Calcul de la factorielle en récursif """
        if n == 0:
            result = 1
        else:
            result = n * facto(n - 1)
        return result
```

plutôt vers de la programmation en apportant des astuces de langages (*ex: les listes en compréhension*)

```
In [5]: [i ** i for i in range(10)]
```

```
Out[5]: [1, 1, 4, 27, 256, 3125, 46656, 823543, 16777216, 387420489]
```

ou bien faire un peu des deux. Toutefois vous devez être conscients qu'il est difficile d'être sur les deux plans lorsque les personnes en face ne connaissent ni l'un ni l'autre.

2.4 D'autres petites choses

Je pense que ces ateliers numériques sont un moment d'échange et de partage de connaissance. Pour ma part de penser avoir appris autant si ce n'est plus que les élèves. C'est pour ça qu'il faut vraiment parler avec eux pour connaître leur niveau, leurs attentes, leur motivation, ... Je pense pour ça qu'un petit cours sur l'algorithmie n'est pas de trop. Bien entendu les cours magistraux sont à éviter mais ils sont parfois importants.

Ensuite, nous avons eu l'occasion de le faire cette année et je pense que c'était une bonne idée, vous pouvez faire une séance "discussion". C'est-à-dire que nous nous étions assis avec eux et nous avons discuté de tous les sujets qui les intéressent: nos études et notre école, les études en général, l'informatique ou simplement répondre à leurs questions. Je pense que pour eux ça a été une séance très constructive.

Enfin la dernière séance peut être consacrée à une présentation à tous d'un code que vous avez réalisé "parfaitement" et qu'ils pourraient récupérer par la suite!

3 Conclusion

En bref, les ateliers doivent être un plaisir d'apporter aux autres un peu de vos connaissances. Ne négligez pas la phase de préparation avant de commencer, elle vous permettra que la suite se déroule facilement. N'hésitez pas à discuter avec vos élèves et de modifier le programme selon les événements. Pour moi l'essentiel est que chacun prenne du plaisir et apprenne quelque chose donc il faut les laisser faire. Essayez que ce soit quelque chose de ludique et amusez-vous!

Quoique notre démineur était trop ambitieux dans le temps imparti, j'ai pour ma part énormément aimé le faire. C'est une expérience unique avec laquelle j'ai beaucoup appris et j'espère que ce genre d'initiative pourra être repris et amélioré par la suite. Merci à vous et n'hésitez à nous contacter si vous avez des questions!