Transition Scratch/Python

Séminaire « Mathématiques et numérique »

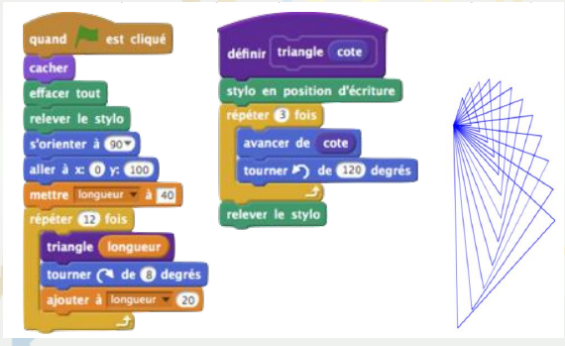
29 janvier 2020

Jérôme Michaud-Bonnet

Transition Scratch/Python

Ressources pour le lycée, juin 2017

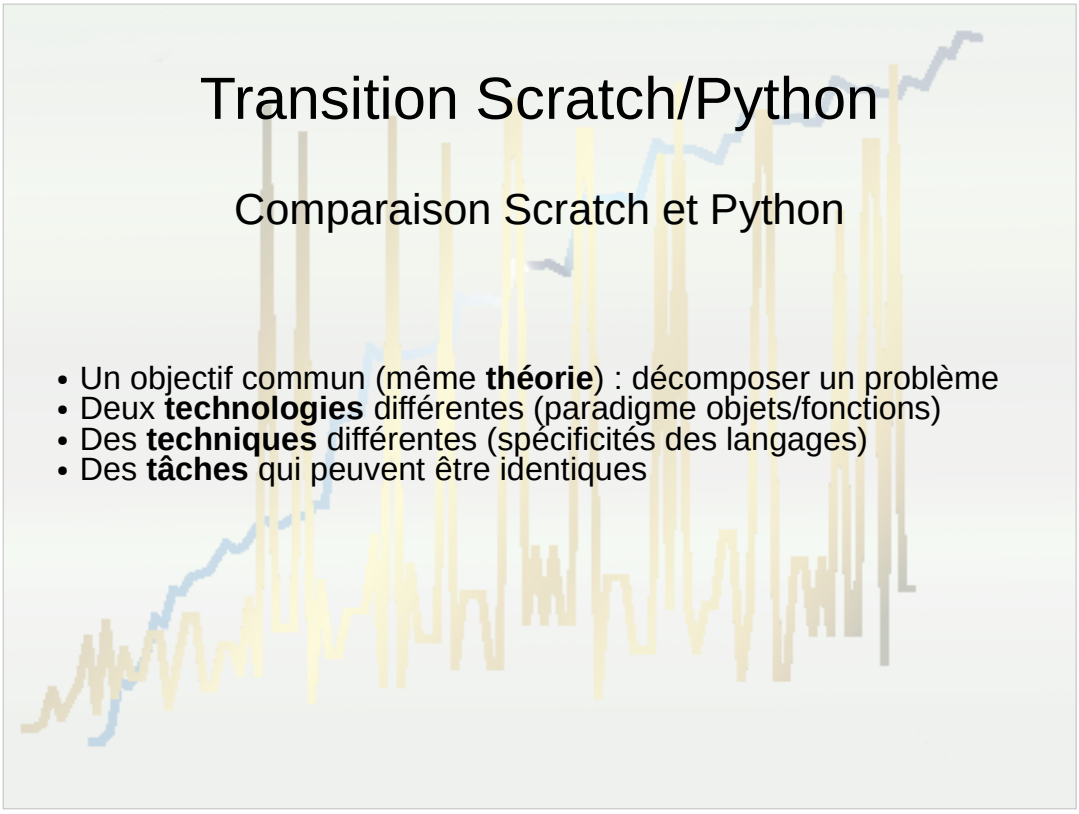
« Les deux langages comportent, au-delà des différences évidentes de forme, des similitudes qui facilitent la transition. »




```
import turtle
def figure():
    turtle.hideturtle()
    turtle.clear()
    turtle.up()
    turtle.setheading(0)
    turtle.goto(0,100)
    longueur = 40
    for i in range(12):
        triangle(longueur)
        turtle.right(8)
        longueur = longueur + 20

def triangle(cote):
    turtle.down()
    for i in range(3):
        turtle.forward(cote)
        turtle.left(120)
    turtle.up()
```

On constate des similitudes mais on verra que peut-être que l'essentiel n'est pas là.



Praxéologie. Ces définitions vont servir par la suite à définir les deux langages comme des technologies différentes.

Transition Scratch/Python

Comparaison Scratch et Python

Deux **technologies** différentes :

Décomposer un problème en plusieurs objets. Décomposer les actions de chaque objet en événements parallèles

Décomposer un problème avec des fonctions

On peut parler de paradigme de programmation

Transition Scratch/Python

Comparaison Scratch et Python

Deux **technologies** différentes :

Décomposer un problème en plusieurs objets. Décomposer les actions de chaque objet en événements parallèles

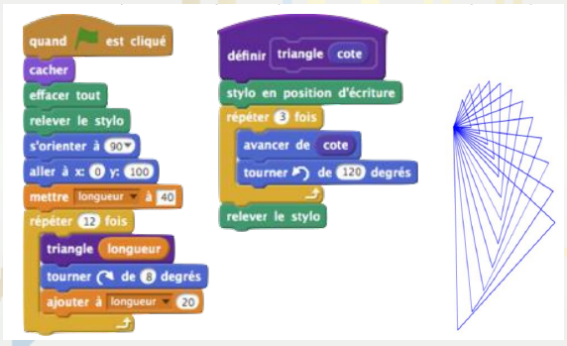
Blocs Scratch (pas de valeur renvoyée) : action, le « faire »

Décomposer un problème avec des fonctions

Fonction Python : valeur renvoyée

Transition Scratch/Python

Changement de technologie (paradigme, fond) ou de technique (forme) ?




```
import turtle
def figure():
    turtle.hideturtle()
    turtle.clear()
    turtle.up()
    turtle.setheading(0)
    turtle.goto(0,100)
    longueur = 40
    for i in range(12):
        triangle(longueur)
        turtle.right(8)
        longueur = longueur + 20

def triangle(cote):
    turtle.down()
    for i in range(3):
        turtle.forward(cote)
        turtle.left(120)
    turtle.up()
```

On voit ici que finalement, il n'y a pas de changement de paradigme. Pour la transition, il va falloir creuser un peu plus.

Transition Scratch/Python

La conjecture de Syracuse

Suite de nombres entiers.

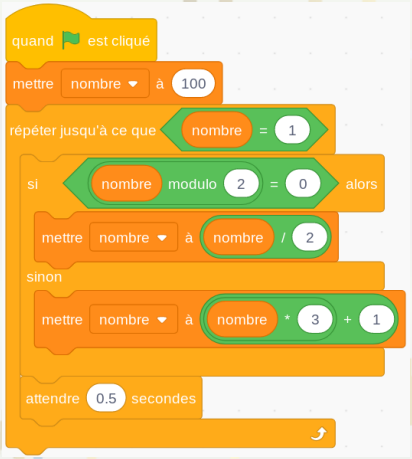
Le successeur d'un nombre N est :
- N/2 si N est pair
- 3 *N+1 si N est impair

On finit toujours par arriver à 1 (?)

Exemple : 17, 52, 26, 13, 40, 20, 10, 5, 16, 8, 4, 2, 1
Temps de vol de l'exemple : 13

Transition Scratch/Python

Changement de Technologie

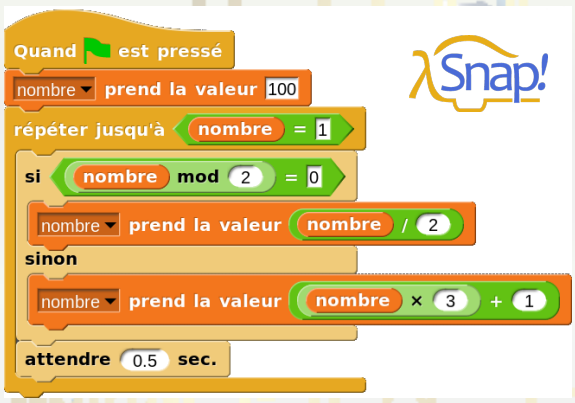



```
nombre=100
while nombre!=1:
    if (nombre%2==0) :
        nombre=nombre/2
    else :
        nombre=nombre*3+1
    print nombre
```

On présente la situation en scratch et Python

Transition Scratch/Python

Changement de Technologie

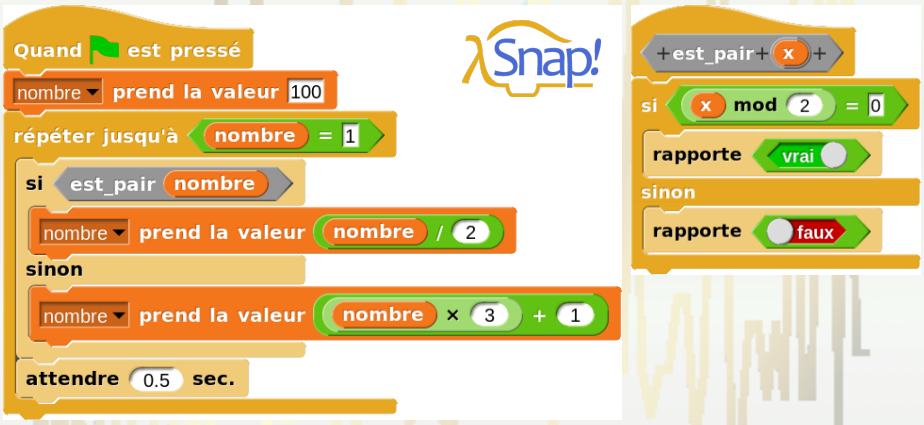



```
nombre=100
while nombre!=1:
    if (nombre%2==0) :
        nombre=nombre/2
    else :
        nombre=nombre*3+1
    print nombre
```

Puis en Snap ! Pour l’instant rien de spécial

Transition Scratch/Python

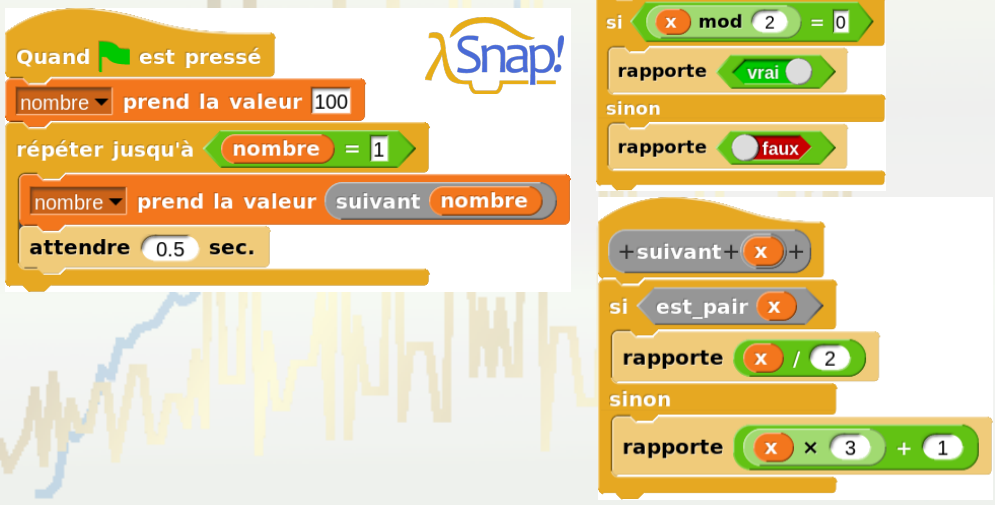
Changement de Technologie



Vous avez remarqué quelque chose ?
→ le « est pair » est une fonction qui renvoie (rapporte) une valeur !

Transition Scratch/Python

Changement de Technologie

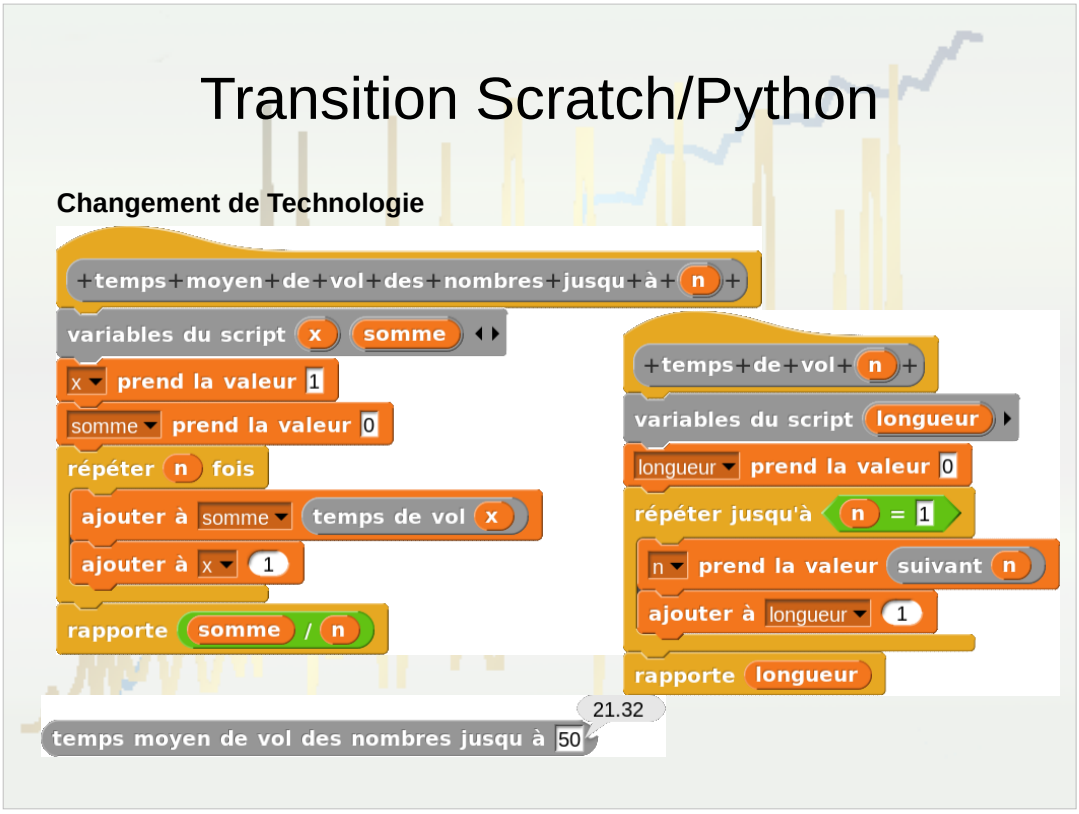


Une autre fonction qui peut être utilisée dans le script principal qui devient plus lisible.

Transition Scratch/Python

Changement de Technologie

On clique sur « temps de vol 500 » et le 110 apparaît en bulle pour donner le résultat. Va vous fait aussi penser à quoi en Python ? → La console !



Et on peut rajouter des surcouches facilement.

Transition Scratch/Python

Changement de Technologie

A travers cet exemple, on peut voir :

- Comment on décompose le problème en fonctions
- L'utilisation des blocs construits peut s'utiliser un peu comme en mode « console »
- Comment l'impossibilité d'écrire des fonctions (qui renvoient une valeur) amène à un affichage qui devient inopportun dès qu'on utilise des fonctions qui renvoient une valeur.
- La création de fonctions permet de masquer la complexité de l'algorithme et permet de développer un algorithme plus complexe. Attention à la mise à disposition systématique de fonctions que l'élève n'écrit pas (principe des bibliothèques dans les langages) : l'élève peut ne pas percevoir l'intérêt de la création de fonctions pour soulager la charge cognitive. En lui permettant de fabriquer lui-même ce qui lui permet d'alléger la complexité, il a à la fois une vue d'ensemble et à la fois une compréhension des détails

Du coup, avec Snap !, on a un changement de technologie par rapport à Scratch alors qu'en apparence, on peut penser que c'est pareil (si on ne voit que la programmation par bloc visuels à assembler).

Transition Scratch/Python

Expérimentation

- Deux classes de troisième (3D classe expérimentale et 3E classe témoin)

Depuis le mois de mars :

3D : utilisent Snap! Créent des blocs qui « rapportent » des valeurs et des blocs « assertion »
3E : utilisent Scratch. Ils ont les mêmes situations que les 3D

A partir de juin :
Les deux classes utilisent Python

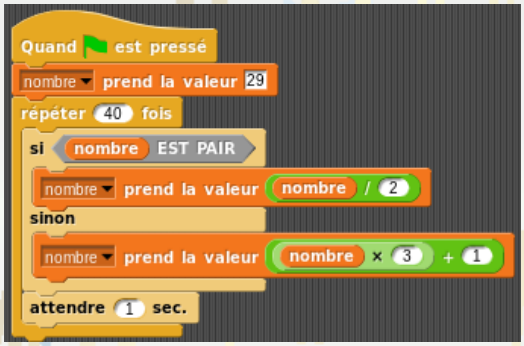
Observation : compréhension et utilisation des fonctions en Python

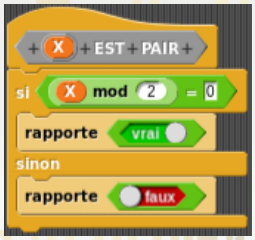
Transition Scratch/Python

Expérimentation

3D (Snap!)

Eve



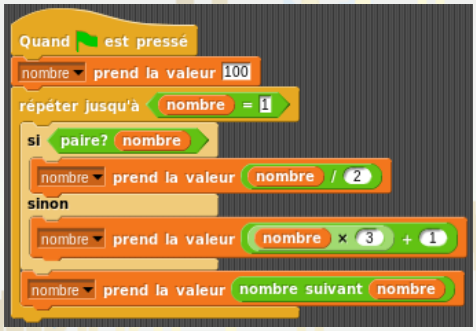


Transition Scratch/Python

Expérimentation

3D (Snap!)

Lara



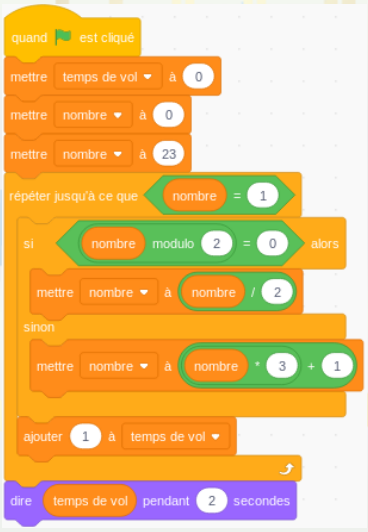
L’élève a préféré appeler sa fonction « paire ? <n> ».

Transition Scratch/Python

Expérimentation

3E (Scratch)

Andréa



C’est faisable en scratch

Transition Scratch/Python

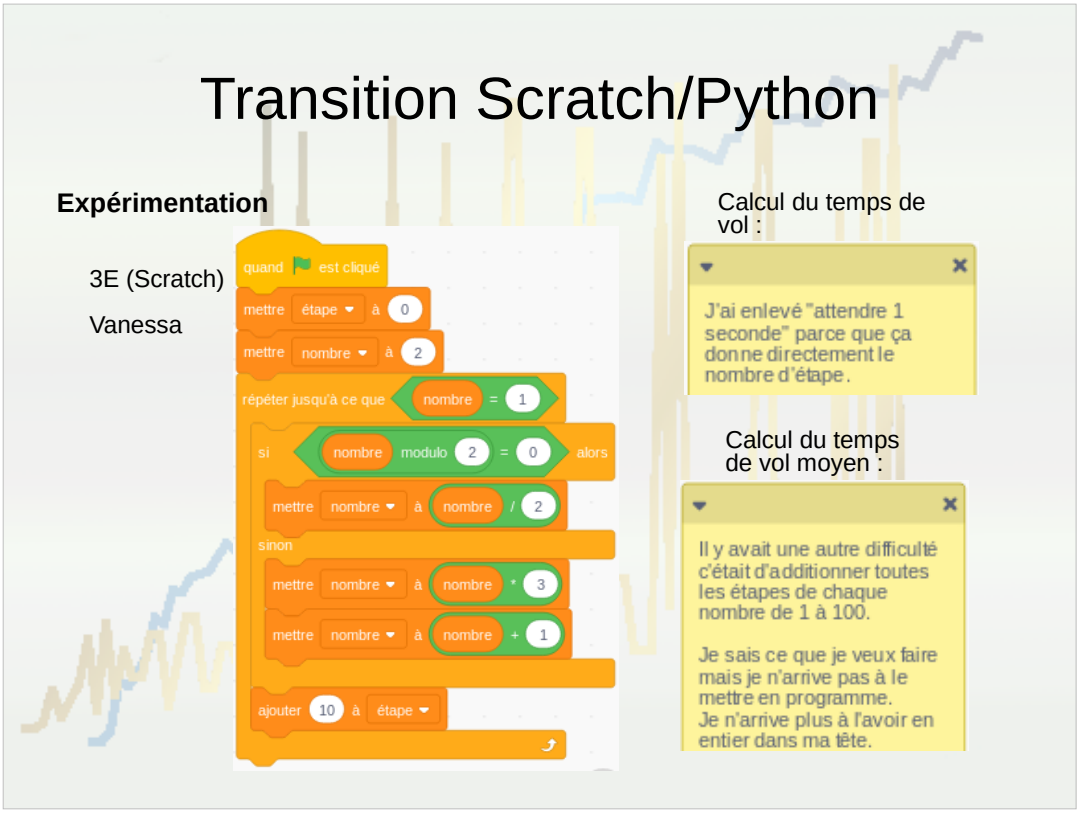
Expérimentation

3E (Scratch)

William

(Pour le temps de vol moyen des nombres jusqu'à 100) :
« Est-ce que je peux prendre des valeurs au hasard entre 1 et 100 ? »

Et là aussi mais il ne reste que les très bons élèves qui tiennent debout devant la complexité.



En Scratch, Vanessa touche du bout des doigts l'utilisation de fonctions : elle a comme encapsulé le calcul des termes successifs et ne s'intéresse qu'au résultat final (« j'ai enlevé attendre 1 seconde parce que ça me donne directement le nombre d'étape ») puis trop complexe pour le calcul du temps de vol moyen, l'élève voit le principe mais se sent débordée. Avec des fonctions…