

1 Analyse critique des résultats fournis par un moteur de recherches

Avant de répondre aux questions suivantes, lire la page suivante jusqu'au paragraphe *Sélectionner*, puis ranger inclus :

<https://interstices.info/un-moteur-de-recherche-pour-le-meilleur-et-pour-le-pire/>

☞ Quel est le rôle d'un navigateur? Citer trois navigateurs connus.

.....
.....

☞ Quels usages différents permettent de distinguer moteur de recherche et navigateur Web?

.....
.....



☞ Quel est l'origine du nom Google?

.....
.....

☞ Visionner l'animation d'URL <http://lienmini.fr/3389-309>, quelles sont les tâches principales réalisées par un moteur de recherche?

.....
.....
.....
.....

☞ Quel est le rôle de l'algorithme PageRank de Google search?

.....

-
- ☞ Quelle requête faut-il écrire dans un moteur de recherche pour obtenir des pages traitant du tennis mais pas du tennis de table?

.....

.....

- ☞ Parmi les cinq articles figurant sur cette page <https://epreuves.pix.fr/sourceinfo/fredericf2.html> déterminer les articles présentant des informations fiables avec le moteur de recherche **Decodex** du journal Le Monde.

.....

.....

.....

- ☞ Soumettre la requête « trotinette électrique »¹ aux moteurs de recherche **Google** et **Qwant** en désactivant les éventuels bloqueurs de publicité. Repérer les résultats qui sont des annonces sponsorisées. Combien de résultats sont-ils communs aux deux moteurs sur les dix premiers résultats non sponsorisés?

.....

.....

.....

.....

.....

- ☞ Quelle est la signification de l'acronyme SEO?

.....

.....

.....

.....

2 Simulation de l'algorithme PageRank

Une bonne présentation de l'algorithme PageRank est donnée dans l'activité 2 du **manuel Numworks** qui a inspiré cet exercice.

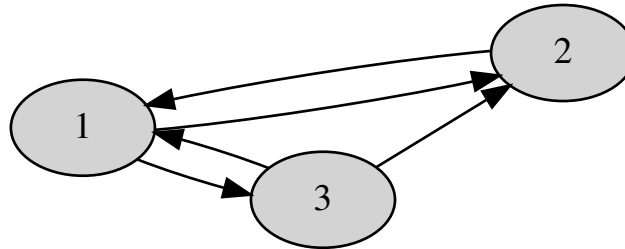
Le PageRank est l'algorithme d'analyse des liens hypertextes utilisé pour le classement des pages Web par le moteur de recherche Google. Le PageRank n'est qu'un indicateur parmi d'autres dans l'algorithme qui permet de classer les pages du Web dans les résultats de recherche. Ce système a été inventé par Larry Page, cofondateur de Google. Ce mot est une marque déposée.

Le principe de base, dit du surfeur aléatoire, est d'attribuer à chaque page une valeur proportionnelle au nombre de fois que passerait par cette page un utilisateur parcourant le Web en cliquant aléatoirement, sur un des liens apparaissant sur

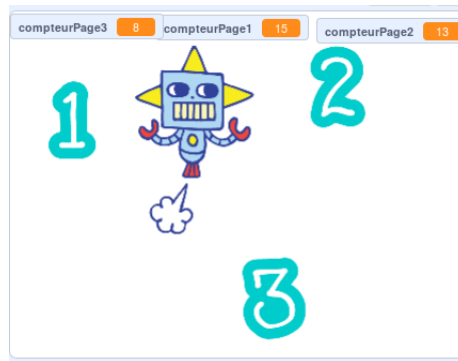
1. La faute d'orthographe sur trotinette est volontaire, comparez la gestion de la correction automatique par les deux moteurs

chaque page. A chaque instant, des robots, parcourent la toile, de lien en lien, et mettent ainsi sans cesse à jour le PageRank des pages du web.

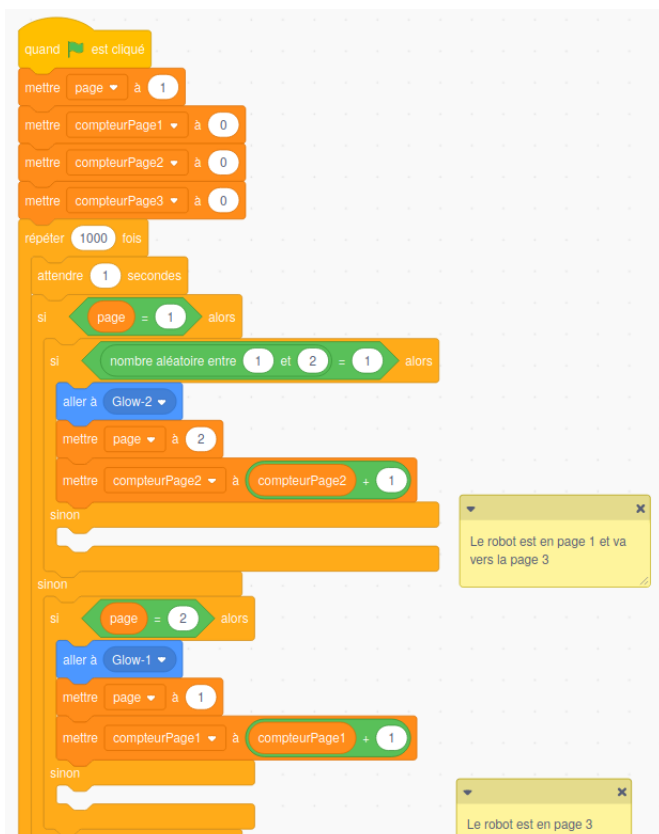
Si on considère trois pages Web numérotées 1, 2 et 3 reliées par les liens hypertextes $1 \rightarrow 2$, $1 \rightarrow 3$, $2 \rightarrow 1$, $3 \rightarrow 1$ et $3 \rightarrow 2$ on peut représenter schématiquement cette situation comme ci-dessous. L'objet mathématique ainsi obtenu s'appelle un graphe.



On va classer ces pages avec l'algorithme de PageRank en simulant 100 déplacements aléatoires de lien en lien d'un robot d'indexation.



On vous fournit une version incomplète de cette simulation en Scratch dans le fichier PageRankEleve.sb3 et sa traduction en Python dans le fichier PageRankEleve.py.



```
#Ceci est un commentaire
#randint(a,b) retourne un entier aléatoire
entre a et b
from random import randint

page = 1
compteurPage1 = 0
compteurPage2 = 0
compteurPage3 = 0
for k in range(1000):
    if page == 1:
        if randint(1, 2) == 1:
            page = 2
            compteurPage2 = compteurPage2 + 1
        else:
            "Le robot est en 1 et va vers 3"
    else:
        if page == 2:
            page = 1
            compteurPage1 = compteurPage1 + 1
        else:
            "le robot est en page 3"
print(compteurPage1, compteurPage2)
```

1. Sur quelle page se trouve le robot au début du programme?

.....
.....
.....

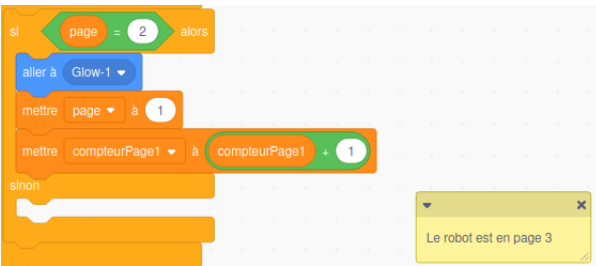
2. Ouvrir le fichier PageRankEleve.sb3 avec Scratch et le fichier PagRankEleve.py avec Pyzo, environnement de programmation Python.

3. Comment est traduite en Python l'instruction Scratch  ?

Comment s'appelle cette structure algorithmique?

.....
.....
.....

4. On donne ci-dessous un extrait de code Scratch et sa traduction en Python :



```
if page == 1:  
    if randint(1, 2) == 1:  
        page = 2  
        compteurPage2 = compteurPage2 + 1  
    else:  
        "Le robot est en 1 et va vers 2"
```

a. Expliquer le code Scratch. Pourquoi parle-t-on de déplacement aléatoire du robot?

.....
.....
.....
.....

b. Quelle est la correspondance en Scratch de l'instruction `compteurPage2 = compteurPage2 + 1`? Que représente `compteurPage2`?

.....
.....

c. Comment les blocs d'instructions sont-ils délimités en Python?

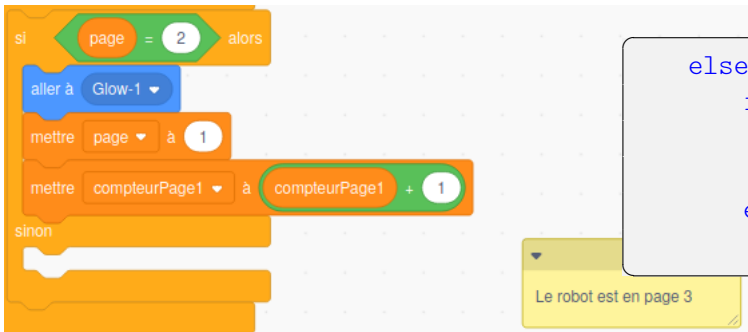
.....
.....
.....

d. Compléter le bloc Scratch du `Sinon` correspondant au cas où le robot est en page 1 et emprunte le lien qui l'amène en page 2. Traduire en parallèle dans le programme Python.

.....

.....
.....

5. Compléter le bloc Scratch du Sinon correspondant au cas où le robot est en page 3. Traduire en parallèle dans le programme Python.



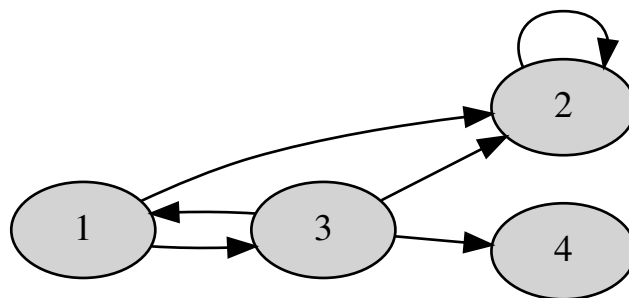
```
else:  
    if page == 2:  
        page = 1  
        compteurPage1 = compteurPage1 + 1  
    else:  
        "le robot est en page 3"
```

.....
.....
.....

6. Exécuter les programmes Scratch et Python lorsqu'ils sont complets. Le nombre de passages du robot sur chaque page représente son PageRank. Quel classement de notoriété des trois pages obtenez-vous après simulation de mille déplacements aléatoires du robot d'indexation? Comparez avec votre voisin. Ce résultat était-il prévisible?

.....
.....
.....

7. Quel problème risque de se poser si on veut simuler le PageRank avec le graphe de documents suivant qui comporte 4 pages reliées par des liens hypertextes? Quelle solution pourrait-on imaginer pour contourner cet obstacle aux déplacements du robot de page en page?



.....
.....