

FICHE DE TRAVAIL

SCIENCES NUMERIQUES ET TECHNOLOGIE

2^{de} – Le Web

Durée : 3 séances de 50 min

Nom Prénom :

ACTIVITE N°1

≈ 15 min

1/ C'est quoi une URL ? Donner ses constituants.

2/ Décomposer l'URL suivante : **<http://www.monlycee.fr/2nde/index.html>**

3/ le nom de domaine est-il unique ? Qui l'attribue ?

ACTIVITE N°2

≈ 10 min

Allez sur la page avec le navigateur Firefox : <https://fr.wikipedia.org/wiki/Informatique>

Appuyer sur F12, se rendre dans l'onglet « Réseau » (et actualiser la page),

1/ Tous les éléments de la page sont-ils chargés simultanément ?

2/ Comment savoir si un élément a bien été chargé ?

3/ Combien de requêtes a-t-il fallu pour charger toute la page ?

1/ Regarder la vidéo suivante :

https://www.youtube.com/watch?time_continue=286&v=0A5fQER40Wg&feature=emb_logo

Avec près de 92 % de parts de marché de la recherche web en France, Google est le moteur de recherche incontournable. Pourtant des alternatives existent. Essayons de tester trois moteurs de recherche placés sous le signe de la confidentialité :

- [DuckDuckGo](#)
- [Qwant](#)
- [StartPage](#)

2/ Quelles sont les différences observées par rapport aux moteurs de recherche établis (Google, Bing, ...) ? (Avantages / inconvénients)

Ressources :

- <https://www.archimag.com/vie-numerique/2015/06/24/duckduckgo-pleine-ascension>
- <https://www.archimag.com/veille-documentation/2019/05/15/qwant-causes-transforme-clics-moteur-recherche-dons-associations>

1/ Après avoir regardé les 2 vidéos ci-dessous (n'hésitez pas à prendre des notes pendant le visionnage des vidéos), rédigez un texte de quelques lignes qui résumera les thèmes abordés dans ces vidéos. Ce résumé devra être produit à l'aide d'un traitement de texte.

- <https://www.dailymotion.com/video/x16lt53>
- https://www.youtube.com/watch?time_continue=5&v=fm5MSdPU8tY&feature=emb_title

2/ À l'aide des informations présentes sur le site de la [CNIL](#) et dans la vidéo ci-dessous, paramétrez votre navigateur web afin de limiter vos traces lors de vos navigations sur le web.

Lien de la vidéo : <https://www.dailymotion.com/video/xw48jr>

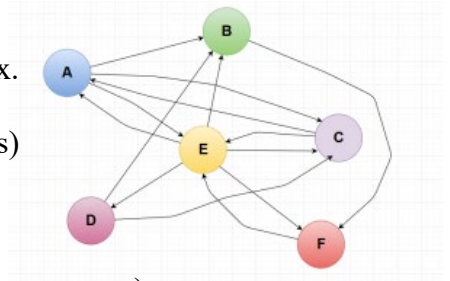
« En 1998, deux jeunes doctorants de l'université de Stanford, Larry Page et Sergey Brin (en collaboration avec Motwani et Wynograd) publiaient un article intitulé "The PageRank Citation Ranking : Bringing Order to the Web" présentant les résultats d'un nouvel algorithme permettant de classer les pages web selon leur popularité et montrant la précision de cet algorithme sur un nouveau moteur de recherche appelé ... Google !

- On simule le comportement d'un internaute.
- On démarre au hasard sur une page
- On suit de façon aléatoire un lien se trouvant sur cette page
- Et ainsi de suite

On impose le fait que dans 15% des cas, l'utilisateur abandonne sa navigation pour repartir d'une page au hasard. Notre utilisateur va se balader de pages en pages, en faisant son petit tour du net. A chaque fois que l'utilisateur tombe sur une page donnée, cette page gagne un point. Et à la fin, la page ayant le plus de point est alors la page la plus populaire du réseau ! »

Considérons 6 sites web : A, B, C, D, E, F ayant des liens hypertexte entre eux.

- Le site A contient un lien vers les sites "B", "C" et "E" (3 liens sortants)
 - Le site B contient un lien vers le site "F" (1 lien sortant)
 - Le site C contient un lien vers les sites "A" et "E" (2 liens sortants)
 - Le site D contient un lien vers les sites "B" et "C" (2 liens sortants)
 - Le site E contient un lien vers les sites "A", "B", "C", "D" et "F" (5 liens sortants)
- Le site F contient un lien vers le site "E" (1 lien sortant)



```
import random
# création d'un dictionnaire Hypertexte
Hypertext = {}
# création d'un dictionnaire pour le nombre de visite
Walk_Number = {}
# une variable pour le nombre total de visite
Total_Walk = 0
#liste des sites web
Websites = ["A","B","C","D","E","F"]
# les liens hypertextes, le dictionnaire possède des clés (noms des sites) qui contiennent des listes (liens
hypertextes)
Hypertext["A"] = ["B","C","E"]
Hypertext["B"] = ["F"]
Hypertext["C"] = ["A","E"]
Hypertext["D"] = ["B","C"]
Hypertext["E"] = ["A","B","C","D","F"]
Hypertext["F"] = ["E"]
print(Hypertext)
# On initialise à 0.0 les visites des sites
Walk_Number["A"] = 0.0
Walk_Number["B"] = 0.0
Walk_Number["C"] = 0.0
Walk_Number["D"] = 0.0
Walk_Number["E"] = 0.0
Walk_Number["F"] = 0.0
```

Et la partie calcul, l'algorithme :

```
i = 0
while i < 1000:
    x = random.choice(Websites)
    while random.random() < 0.85:
        Walk_Number[x] = Walk_Number[x] + 1
        Total_Walk = Total_Walk + 1
        x = random.choice(Hypertext[x])
    i = i + 1
print (Walk_Number)
print(Total_Walk)
```

1/ Après avoir entré le code ci-dessus. Donner le classement obtenu ? Était-ce prévisible ?

2/ Faire afficher dans la console les sites classés par ordre de popularité.