



Ce document est l'un des livrables à fournir lors du dépôt de votre projet : 4 pages maximum (hors documentation).

Pour accéder à la liste complète des éléments à fournir, consultez la page [Préparer votre participation](#).

Vous avez des questions sur le concours ? Vous souhaitez des informations complémentaires pour déposer un projet ? Contactez-nous à info@trophees-nsi.fr.

NOM DU PROJET : Bref, Youtube m'ennuie

> PRÉSENTATION GÉNÉRALE :

« Computer Chronicles » est une émission diffusée de 1983 à 2002 sur la chaîne de télévision américaine PBS. Ce programme, composé d'environ 400 émissions de 30 minutes, constitue une base de choix pour chercher des informations d'époque sur l'histoire des sciences du numérique, que ce soient des systèmes d'exploitation, des processeurs, des logiciels, ou des langages de programmation.

Cependant, il est impossible de chercher une information, par exemple la manière dont on présentait Linux à l'époque, sans regarder les 200 heures d'émissions.

Le but de ce projet est donc de concevoir un programme qui permet de nous placer à la seconde près où un terme recherché est prononcé par les présentateurs ou les invités en utilisant, comme base de données, les sous-titres automatiques des 400 émissions « The Computer Chronicles ».

Nous avons vu en cours un algorithme de recherche textuelle bien plus efficace que l'algorithme dit « naïf », qui est l'algorithme de Boyer-Moore, et notre professeur nous a alors proposé de le mettre en œuvre sur la thématique de l'histoire de l'informatique.

L'intérêt de ce programme est de pouvoir trouver très rapidement dans quelle vidéo et à quel moment on parle d'un sujet, permettant ainsi d'éviter un travail de recherche fastidieux et d'avoir accès en un temps très bref aux informations souhaitées. Cela nous permet **d'exploiter Youtube d'une nouvelle manière.**

> ORGANISATION DU TRAVAIL :

Nous sommes quatre. Thomas, chargé de la programmation de la fenêtre ainsi que de raccorder chaque partie du code fait par les autres. Marc, chargé de la programmation de la classe permettant de faire des recherches de mots dans un texte. Raphaël, chargé de la programmation de la fonction permettant d'extraire les données d'un fichier contenant les sous-titres des vidéos Youtube, ainsi que de la création de la musique de la vidéo. Et Antoine, chargé de réaliser et monter la vidéo.

Pour communiquer nous avons utilisé Discord en créant un groupe privé pour discuter, nous avons également travaillé ensemble directement au lycée.

> LES ÉTAPES DU PROJET :

Nous avons collecté les sous titres des vidéos de la chaîne Youtube « Computer Chronicles » en utilisant l'utilitaire en ligne de commande youtube-dl

Nous avons tout d'abord adapté notre fonction de Boyer-Moore puis nous l'avons intégrée à une classe. Nous avons ensuite créé notre fonction permettant d'extraire les données du fichier texte, ainsi que la classe de la fenêtre, tandis qu'en parallèle, Antoine réalisait un story-board de la vidéo de présentation. Enfin nous avons réuni nos travaux en un seul programme puis avons transmis le résultat à Antoine pour qu'il finalise la préparation de la vidéo.

> FONCTIONNEMENT ET OPÉRATIONNALITÉ :

Nous avons terminé le programme et pour nous assurer qu'il n'y ait pas de bugs, nous avons cherché à le faire buguer en rentrant, par exemple, un chemin de fichier vide mais le programme fonctionne correctement tant que le fichier des sous-titres est lisible et a le format attendu. En ce qui concerne la facilité d'utilisation nous avons créé une interface dont l'utilisation est facile et intuitive, en indiquant notamment à quoi correspondent les entrées textuelles ainsi qu'en affichant des messages d'erreur qui expliquent le problème à l'utilisateur quand celui-ci n'utilise pas correctement le programme.

Lors de la création du projet nous avons rencontré quelques problèmes que nous avons résolu. Le premier était que notre fonction de recherche faite pour renvoyer une liste de résultats n'en renvoyait qu'un seul. Ce fut un problème assez facile à résoudre car il suffisait de rajouter une condition permettant la sortie de la boucle lorsqu'un résultat est trouvé. Le deuxième apparut lorsque nous cherchions à extraire les données du fichier texte. En effet nous n'avions pas prévu qu'il puisse y avoir des virgules placées dans le texte des vidéos alors que nous séparions les données en faisant une opération split sur la virgule. Pour le résoudre nous avons fait en sorte de lire les données caractère par caractère en les interprétant à la manière d'un humain, ce qui ralentit un peu le programme mais évite les problèmes.

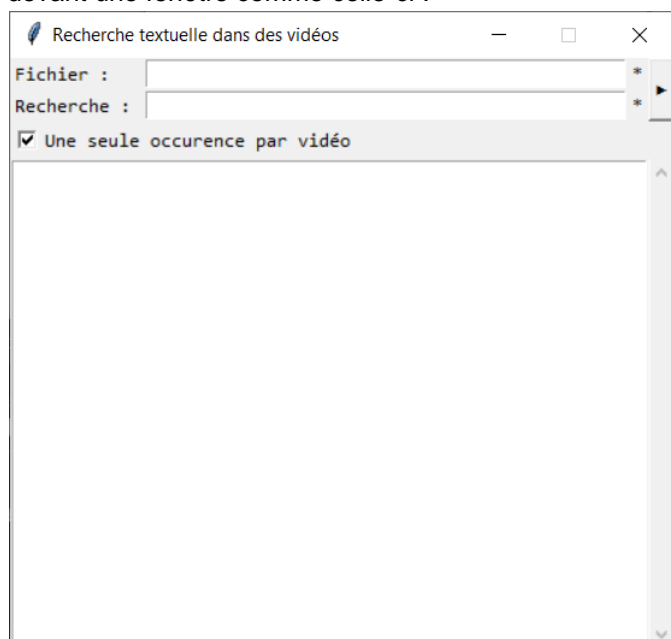
> OUVERTURE :

Nous pourrions étendre l'utilisation de notre programme à d'autres chaînes Youtube (ce qui est normalement déjà possible tant que les données sont sous le format attendu), mais également à d'autres sites, et donc, d'autres formats de fichier sous-titres.

Si c'était à refaire, nous ne pensons pas changer quelque chose car nous trouvons que nous nous sommes bien organisés.

DOCUMENTATION

Pour utiliser notre projet, lancez tout d'abord le programme python. Vous vous retrouverez alors devant une fenêtre comme celle-ci :



Le champ « Fichier » sert à indiquer le chemin du fichier texte contenant les sous-titres à utiliser.

Le champ « Recherche » sert à indiquer quelle recherche vous voulez effectuer sur le fichier.

Les astérisques situées à la suite des champs signifient que l'entrée a été modifiée ou qu'aucun choix n'a été validé. Pour valider vos choix il faut cliquer sur le bouton situé après les entrées ou faire la touche entrée lorsque vous vous trouvez dans l'une d'entre elles.

La case « Une seule occurrence par vidéo » sert à indiquer si on ne veut qu'un seul résultat, et donc qu'un seul lien par vidéo où la recherche a été trouvée. Si elle est cochée, le résultat est celui correspondant à la première occurrence trouvée et donc, celle qui apparaît le plus tôt dans la vidéo.

Le cadre blanc en dessous est une liste vide pour le moment. Les résultats des recherches s'afficheront directement dedans et la barre de défilement à droite vous permet de naviguer dedans. Si vous ne voulez pas à avoir à recopier lettre par lettre les URL trouvés il vous suffit de cliquer sur un résultat et cela vous ouvrira automatiquement la page web correspondant à l'URL.

Nous avons programmé en Python en utilisant la librairie `tkinter` pour la fenêtre ainsi que les messages d'erreur, la fonction `isfile` de la librairie `os.path` pour vérifier si le fichier saisi existe bien, et la fonction `open` de la librairie `webbrowser` pour ouvrir les pages web. Nous avons également utilisé la programmation orientée objet en créant deux classes. La première est la classe permettant de faire des recherches dans un texte à l'aide de l'algorithme de Boyer-Moore, et la seconde est la classe définissant la fenêtre qui fait appel à la première classe ainsi qu'à la fonction permettant d'extraire les données du fichier texte.

Le fichier du programme est au format `.pyw`, qui est le même que le format classique `.py`, mais il empêche l'apparition de la console lors de l'ouverture du fichier, ce qui est mieux pour nous car nous n'avons pas besoin de la console.

En ce qui concerne le stockage des données, celles-ci sont contenues dans un unique fichier texte sous le format utilisé par l'extracteur de youtube-dl qui est le suivant :

['texte de la vidéo', 'start' : timecode, nombre que nous n'utilisons pas, 'suite du texte', etc.]

<NOUVELLEVIDEO>URL</NOUVELLEVIDEO>

Ce qui donne ceci :

```
1 ['this was the very first episode of the', 'start': 2.449, 5.021, 'computer chroni
2 <NOUVELLEVIDEO>Tdj8gh9GPc4</NOUVELLEVIDEO>
3 ['this week on the computer chronicles the', 'start': 24.74, 4.77, 'y2k problem wi
4 <NOUVELLEVIDEO>nAFIsPX3_3A</NOUVELLEVIDEO>
5 ['oh', 'start': 0.0, 2.0, 'oh', 'start': 6.06, 2.449, 'oh', 'start': 11.5, 2.09, '
6 <NOUVELLEVIDEO>n9rUw-Jls8w</NOUVELLEVIDEO>
7 ['this is the nordstrom store in san', 'start': 19.31, 4.99, "francisco it's one o
8 <NOUVELLEVIDEO>cHkdQ5CG0PY</NOUVELLEVIDEO>
9 ['these are the software shelves at the', 'start': 19.55, 4.27, 'compusa store in
10 <NOUVELLEVIDEO>93qjf290Mfw</NOUVELLEVIDEO>
11 ['you', 'start': 0.0, 2.06, "if you've just converted your pc to", 'start': 28.26,
12 <NOUVELLEVIDEO>d98Xyssu_oM</NOUVELLEVIDEO>
13 ["it's not often that a new software", 'start': 19.189, 5.021, 'product gets its o
14 <NOUVELLEVIDEO>Q7AQVdmnZi4</NOUVELLEVIDEO>
15 ['you', 'start': 0.0, 2.06, 'windows 3.0 as quickly emerging as the', 'start': 24.
16 <NOUVELLEVIDEO>YewNEAIkbG4</NOUVELLEVIDEO>
```

Exemple :

Voici ce qu'on obtient avec les vidéos de la chaîne « The Computer Chronicles » et la recherche « star wars » :

