Exercices LYOU008 Serie 3

December 2, 2019

0.1 Donner la longueur de la plus longue ligne du fichier.

[8]:
```
with open("demo-file-latin.txt", "r") as f:
    l_max = 0
    for line in f:
        if len(line) > l_max:
            l_max = len(line)
print(l_max)
```

```
2649
```

0.2 Combien de lignes commencent par "–" ?

[13]:
```
with open("demo-file-latin.txt", "r") as f:
    nb_lignes = 0
    for ligne in f:
        if ligne.startswith('--'):
            nb_lignes += 1
print(nb_lignes)
```

```
1179
```

0.3 Donner la longueur moyenne des lignes. Même question avec lignes non vides.

[18]:
```
with open("demo-file-latin.txt", "r") as f:
    longueur_cumulee = 0
    nb_lignes = 0
    for ligne in f:
        longueur_cumulee += len(ligne)
        nb_lignes += 1
print("Le fichier comprend %d lignes pour une longueur totale de %d caractères␣
 ↪ce qui fait une moyenne de %.2f caractères par ligne"%
     (nb_lignes, longueur_cumulee, longueur_cumulee/nb_lignes))
```

```
Le fichier comprend 5823 lignes pour une longueur totale de 497719 caractères ce
qui fait une moyenne de 85.47 caractères par ligne
```

[3]:
```
# Avec les lignes non vides
with open("demo-file-latin.txt", "r") as f:
    longueur_cumulee = 0
    nb_lignes = 0
    for ligne in f:
        if len(ligne.strip()) != 0:
            longueur_cumulee += len(ligne)
            nb_lignes += 1
print("Le fichier comprend %d lignes non vides pour une longueur totale de %d␣
 ↪caractères ce qui fait une moyenne de %.2f caractères par ligne"%
     (nb_lignes, longueur_cumulee, longueur_cumulee/nb_lignes))
```

```
Le fichier comprend 2837 lignes non vides pour une longueur totale de 494657
caractères ce qui fait une moyenne de 174.36 caractères par ligne
```

0.4 Afficher les lignes qui contiennent le mot *maintenant*

[8]:
```
f = open("demo-file-latin.txt", "r")
for lgn in f:
    if lgn.find("maintenant") != -1:
        print(lgn.strip()[:75])
f.close()
```

```
Est-ce que ce n'est pas de lui ? Voyager à travers les airs ! Le voilà jalo
Voyons maintenant ce que firent les lieutenants Burton et Speke dans l'Afri
--Et maintenant cherche sur la côte l'île de Zanzibar, par 6° de latitude s
--Suis maintenant ce parallèle et arrive à Kazeh.
« Voici maintenant des chiffres très exacts.
« Je doutais, dit-il en tendant la main à Samuel Fergusson, mais maintenant
« Nous nous sommes attardés, dit le docteur. Il nous faut maintenant traver
--Es-tu convaincu maintenant !
« Et maintenant, dit-il, au ballon !
--Peu nous importe maintenant ! Que le vent nous pousse dans le nord pendan
Mais, ainsi qu'il l'avait fait comprendre à Kennedy, par suite de sa perte
--Ce prêtre, qui avait fait v u de pauvreté, repose maintenant dans une min
En effet, une bande épaisse et maintenant distincte s'élevait lentement au-
A quinze cents pieds environ du sol, il rencontra la masse opaque du nuage,
« Et maintenant, Joe, dit 1e docteur, jette-moi en dehors une cinquantaine
--Oui, cette contrée est fatale ! Nous marchons directement vers le royaume
Le docteur essaya d'en fixer la configuration actuelle, bien différente déj
--Surtout maintenant que nous sommes fixés sur la qualité de l eau du Tchad
--Nous essayerons de regagner la partie septentrionale du lac, en nous main
--Pauvre Joe ! bonne et excellente nature ! c ur brave et franc ! Un moment
« Et maintenant, dit le docteur, le ciel nous conduise où il lui plaira !
--Oh ! ce que j'en ai fait; répondit celui-ci, ce n'est pas pour vous; c'es
```

0.5 Afficher les lignes qui contiennent le mot *maintenant* indépendamment de la casse (majuscule ou minuscule)

[9]:
```
f = open("demo-file-latin.txt", "r")
for lgn in f:
    if lgn.lower().find("maintenant") != -1:
        print(lgn.strip()[:75])
f.close()
```

```
Est-ce que ce n'est pas de lui ? Voyager à travers les airs ! Le voilà jalo
Voyons maintenant ce que firent les lieutenants Burton et Speke dans l'Afri
--Et maintenant cherche sur la côte l'île de Zanzibar, par 6° de latitude s
--Suis maintenant ce parallèle et arrive à Kazeh.
« Maintenant, Messieurs, comme détail pratique, j'ajouterai ceci.
« Voici maintenant des chiffres très exacts.
« Je doutais, dit-il en tendant la main à Samuel Fergusson, mais maintenant
Maintenant, dit Fergusson, prends deux fusils, ami Dick, l'un pour toi, l a
Maintenant, mes amis, dit le docteur Fergusson, il faut tout prévoir nous p
« Nous nous sommes attardés, dit le docteur. Il nous faut maintenant traver
« Maintenant, mes amis, soyez prêts à tout hasard.
--Es-tu convaincu maintenant !
« Et maintenant, dit-il, au ballon !
--Peu nous importe maintenant ! Que le vent nous pousse dans le nord pendan
Mais, ainsi qu'il l'avait fait comprendre à Kennedy, par suite de sa perte
--Ce prêtre, qui avait fait v u de pauvreté, repose maintenant dans une min
« Maintenant, Joe, dit le docteur, il te reste encore une jolie fortune, si
En effet, une bande épaisse et maintenant distincte s'élevait lentement au-
A quinze cents pieds environ du sol, il rencontra la masse opaque du nuage,
« Et maintenant, Joe, dit 1e docteur, jette-moi en dehors une cinquantaine
--Oui, cette contrée est fatale ! Nous marchons directement vers le royaume
Maintenant nous pouvons dormir tranquilles, dit le docteur.
Le docteur essaya d'en fixer la configuration actuelle, bien différente déj
--Surtout maintenant que nous sommes fixés sur la qualité de l eau du Tchad
--Je l'espère. Maintenant, Dick, tu vas chasser aux environs, sans t éloign
--Nous essayerons de regagner la partie septentrionale du lac, en nous main
--Pauvre Joe ! bonne et excellente nature ! c ur brave et franc ! Un moment
« Et maintenant, dit le docteur, le ciel nous conduise où il lui plaira !
--Oh ! ce que j'en ai fait; répondit celui-ci, ce n'est pas pour vous; c'es
```

0.6 Compter le nombre de mots (ou plutôt de tokens) que contient le texte.

[10]:
```
f = open("demo-file-latin.txt", "r")
nb_mots = 0
for lgn in f:
    liste_mots = lgn.split()
    nb_mots += len(liste_mots)
f.close()
```

```
print("Le texte comprend %d tokens." % nb_mots)
```

```
Le texte comprend 81701 tokens.
```

0.7 Compter le nombre de tokens du textes si on met tout en minuscules.

[4]:
```
# Enoncé ambigu : si on met simplement tout le texte en minuscule, cela ne␣
 ↪change rien au nombre d'occurrences de tokens. On retouve le même résultat␣
 ↪que dans l'exercice précédent
f = open("demo-file-latin.txt", "r")
nb_mots = 0
for lgn in f:
    liste_mots = lgn.lower().split()
    nb_mots += len(liste_mots)
f.close()
print("Le texte comprend %d tokens." % nb_mots)

# On pourrait aussi vouloir compter le nombre de tokens différents que l'on␣
 ↪trouve si on mets tout le texte en minuscules
# (supprimant ainsi la différence entre 'Le' et 'le')
# La méthode choisie ci-dessous est très peu efficace: elle peut prendre␣
 ↪quelques secondes, ce qui est énorme vue la taille réduite du texte manipulé.
liste_tokens = []
f = open("demo-file-latin.txt", "r")
for lgn in f:
    liste_mots = lgn.lower().split()
    for mot in liste_mots:
        if mot not in liste_tokens:
            liste_tokens.append(mot)
f.close()
print("Le texte comprend %d (types de) tokens différents." % len(liste_tokens))

# Affichage des 20 premiers tokens trouvés pour se faire idée du contenu de la␣
 ↪liste
for t in liste_tokens[:20]:
    print (t, " ", end="")
```

```
Le texte comprend 81701 tokens.
Le texte comprend 15587 (types de) tokens différents.
jules verne cinq semaines en ballon voyage de découvertes afrique par
3 anglais chapitre premier la fin d'un discours très
```

0.8 Afficher tous les mots qui commencent par *voya*

[15]:
```
# Pour répondre à cette question je choisis de construire d'abord la liste des␣
 ↪mots/tokens du texte en mémoire
# Ce n'est pas indispensable, on peut répondre à la question en scannant tout␣
 ↪le fichier sans construire de structure de données en mémoire.
tokens_du_texte = []
with open("demo-file-latin.txt", "r") as f:
    for ligne in f:
        liste_tok = ligne.strip().split()
        tokens_du_texte.extend(liste_tok)

# Affichage des premiers tokens pour une vérification sommaire
print(tokens_du_texte[:50])

# Nouvelle boucle qui parcourt la liste des tokens
for tok in tokens_du_texte:
    if tok.startswith("voya"):
        print(tok, end="/")
print()
```

```
['JULES', 'VERNE', 'CINQ', 'SEMAINES', 'EN', 'BALLON', 'VOYAGE', 'DE',
'DÉCOUVERTES', 'EN', 'AFRIQUE', 'PAR', '3', 'ANGLAIS', 'CHAPITRE', 'PREMIER',
'La', 'fin', "d'un", 'discours', 'très', 'applaudi.--Présentation', 'du',
'docteur', 'Samuel', 'Fergusson--«', 'Excelsior.', '»--Portrait', 'en', 'pied',
'du', 'docteur.--Un', 'fataliste', 'convaincu.--Dîner', 'au', "Traveller's",
'club.--Nombreux', 'toasts', 'de', 'circonstance', 'Il', 'y', 'avait', 'une',
'grande', 'affluence', "d'auditeurs,", 'le', '14', 'janvier']
voyageurs/voyageurs/voyageur,/voyageur/voyages,/voyage/voyages,/voyages,/voyageu
rs/voyageurs/voyageur/voyage/voyage/voyages/voyage,/voyage./voyage/voyage./voyag
e/voyage,/voyage/voyage/voyage/voyageurs/voyage/voyageur./voyage/voyageurs;/voya
geurs/voyageur/voyage,/voyage/voyage…/voyageurs/voyageur/voyageur,/voyage./voy
ages,/voyage./voyage/voyage/voyage,/voyage/voyage/voyageurs/voyageurs/voyage/voy
age;/voyage/voyageant/voyaient/voyage/voyage/voyage-là/voyages/voyageurs/voyage/
voyageurs/voyage/voyage,/voyageurs/voyageurs/voyage/voyageurs/voyageurs/voyait/v
oyagerons/voyageurs,/voyage/voyageurs/voyage/voyageurs/voyait/voyageurs/voyager,
/voyageurs/voyageurs/voyant/voyageurs/voyantes,/voyageurs,/voyage,/voyage/voyant
/voyageurs/voyager,/voyageur./voyageurs/voyage,/voyageurs./voyageurs/voyageurs./
voyager/voyageurs;/voyait/voyage/voyageurs/voyage;/voyageurs/voyageurs/voyait/vo
yageurs/voyait/voyait/voyageurs/voyage/voyageur/voyant/voyageurs/voyageur,/voyag
eurs/voyage/voyageurs,/voyageur/voyage,/voyageurs/voyageurs./voyageur/voyage,/vo
yageurs/voyant/voyageurs,/voyage/voyage/voyage,/voyageurs/voyage/voyageurs;/voya
ger/voyage/voyage/voyait/voyage/voyageurs/voyageurs/voyageurs;/voyageurs/voyageu
rs,/voyageurs/voyageurs;/voyant/voyageurs/voyageur/voyageurs/voyageurs/voyageurs
/voyageurs/voyages/voyage/voyage:/voyage/voyant/voyageur/voyageurs/voyageurs/voy
ageurs/voyageurs/voyageurs,/voyageurs/voyage./voyage,/voyageurs/voyage./voyageur
s,/voyageur/voyageurs/voyageurs/voyait/voyage/voyages/voyageurs/voyageurs/voyant
/voyageurs./voyageurs/voyageurs,/voyageurs/voyageurs/voyage/voyageurs/voyageurs/
```

voyage/voyageurs/voyageurs/voyageurs/voyait/voyageur/voyant/voyageurs/voyait/voyant/voyage/voyage/voyageurs,/voyage/voyageurs,/voyageurs/voyageur/voyageur./voyages/voyage/voyageur/voyageurs,/voyage,/voyage/voyage/voyageurs/voyageurs./voyageurs;/voyage./voyage/voyage./voyageurs/voyage,/voyagé/voyageurs/voyageurs/voyageurs/voyageurs./voyageurs/voyage,/voyageurs/voyageurs/voyageurs,/voyageurs/voyage./voyageurs,/voyageurs/voyageurs/voyageurs./voyage/voyageurs/voyage./voyage/

0.9 Dans la liste des mots du texte, supprimer les signes de ponctuation qui apparaissent à la fin des mots.

[5]:
```
# Liste des signes de ponctuation
l_ponct = ['.', ',',';',':', '!','?','-']

tokens_du_texte = []
with open("demo-file-latin.txt", "r") as f:
    for ligne in f:
        liste_tok = ligne.strip().split()
        for i in range(len(liste_tok)):
            if liste_tok[i][-1] in l_ponct:
                liste_tok[i] = liste_tok[i][:-1]
        tokens_du_texte.extend(liste_tok)

# Affichage des premiers tokens pour une vérification sommaire
print(tokens_du_texte[:50])
```

```
['JULES', 'VERNE', 'CINQ', 'SEMAINES', 'EN', 'BALLON', 'VOYAGE', 'DE',
'DÉCOUVERTES', 'EN', 'AFRIQUE', 'PAR', '3', 'ANGLAIS', 'CHAPITRE', 'PREMIER',
'La', 'fin', "d'un", 'discours', 'très', 'applaudi.--Présentation', 'du',
'docteur', 'Samuel', 'Fergusson--«', 'Excelsior', '»--Portrait', 'en', 'pied',
'du', 'docteur.--Un', 'fataliste', 'convaincu.--Dîner', 'au', "Traveller's",
'club.--Nombreux', 'toasts', 'de', 'circonstance', 'Il', 'y', 'avait', 'une',
'grande', 'affluence', "d'auditeurs", 'le', '14', 'janvier']
```

0.10 En construisant un dictionnaire, indiquez le nombre de types différents que l'on trouve dans le texte.

[16]:
```
# l'énoncé demandait qu'on construise un dictionnaire, mais
# on peut répondre aussi à la question en utilisant un set():
# la structure de donnée 'set' (ensemble): la méthode .add()
# n'ajoute un élément que s'il n'y est pas déjà.
tokens = set()
with open("demo-file-latin.txt", "r") as f:
    for ligne in f:
        liste_tok = ligne.strip().split()
        for t in liste_tok:
            tokens.add(t)
print(len(tokens))
```

```
# Version avec dictionnaire (on va en profiter pour
# compter le nombre d'occurrences de chaque token)
dico_tokens = {}
with open("demo-file-latin.txt", "r") as f:
    for ligne in f:
        liste_tok = ligne.strip().split()
        for t in liste_tok:
            dico_tokens[t] = dico_tokens.get(t,0) + 1
print(len(dico_tokens))

# Affichage des premiers items du dictionnaire pour une vérification sommaire
c = 0
for k in dico_tokens.keys():
    print(k, dico_tokens[k], end=' / ')
    c += 1
    if c > 50:
        break
```

```
16023
16023
JULES 1 / VERNE 1 / CINQ 1 / SEMAINES 1 / EN 2 / BALLON 1 / VOYAGE 1 / DE 1 /
DÉCOUVERTES 1 / AFRIQUE 1 / PAR 1 / 3 4 / ANGLAIS 1 / CHAPITRE 44 / PREMIER 1 /
La 126 / fin 15 / d'un 183 / discours 8 / très 64 / applaudi.--Présentation 1 /
du 942 / docteur 373 / Samuel 65 / Fergusson--« 1 / Excelsior. 1 / »--Portrait 1
/ en 781 / pied 27 / docteur.--Un 1 / fataliste 2 / convaincu.--Dîner 1 / au 454
/ Traveller's 3 / club.--Nombreux 1 / toasts 3 / de 3580 / circonstance 4 / Il
213 / y 133 / avait 145 / une 710 / grande 60 / affluence 2 / d'auditeurs, 1 /
le 2073 / 14 4 / janvier 3 / 1862, 1 / à 1584 / la 1494 /
```

0.11 Donnez les 10 mots les plus fréquents du texte.

[29]:
```
dico_tokens = {}
with open("demo-file-latin.txt", "r") as f:
    for ligne in f:
        liste_tok = ligne.strip().split()
        for t in liste_tok:
            dico_tokens[t] = dico_tokens.get(t,0) + 1

# Ce dictionnaire donne la fréquence de chaque token
# Il faut trouver les plus fréquents:
dico_trie = sorted(dico_tokens.items(), key = lambda x: x[1], reverse=True)
for k,v in dico_trie[:10]:
    print("le mot %5s apparaît %d fois" % (k,v))
```

```
le mot    de apparaît 3580 fois
le mot    le apparaît 2073 fois
```

```
le mot    et apparaît 1603 fois
le mot     à apparaît 1584 fois
le mot    la apparaît 1494 fois
le mot   les apparaît 1247 fois
le mot     ! apparaît 1066 fois
le mot    du apparaît 942 fois
le mot    il apparaît 906 fois
le mot    un apparaît 822 fois
```