

University of Science and Technology HOUARI BOUMEDIENE
Faculty of Electronics and Computer Science



Département d'informatique

Examen Python –L2 S4

USTHB/F.E.I/Département Informatique/L2

ACAD : A, B et C/Examen : PYTHON - 2019/2020

Fait par l'étudiant :

NOM : KHITER.

PRENOM : Mohamed Achraf.

Matricule : 171731032520.

Section : B.

Groupe : 03.

Exercice 01 :

1- Explication de chaque ligne de code du script suivant, permettant de lire tout le contenu d'un répertoire :

le script suivant permet de lister tout le contenu d'un répertoire recursivement jusqu'à trouver le dernier sous fichier (condition d'arrêt)

- import os,sys : ça veut dire importer les deux modules (bibliotheques) standards du python , pour qu'on puisse profiter de l'utilisation de ses fonctions et sous modules . Les modules `os` et `sys` fournissent de nombreux outils pour gérer les noms de fichiers, les chemins et les répertoires. Ces modules sont des `wrappers` (emballages) pour des modules spécifiques à la plate-forme, donc des fonctions comme : `os.path.isdir(i)` , `sys.argv[i]` , sous UNIX, Windows, Mac OS et toute autre plate-forme prise en charge par Python.
- def R(d): c'est la définition d'une fonction qui s'appelle (R) avec une variable (d) comme paramètre d'entrée. (d) est du type (string) fait la référence au nom du répertoire / fichier.
- for i in os.listdir(d): le debut de la boucle (for) en utilisant une variable (i) aussi du type (string) qui fait référence au nom du répertoires / fichiers situe sous le repertoire courant (d) un par un dans chaque iteration. l'utilisation de la fonction standard `os.listdir(d)` permet de lister le contenu du repertoire (d) de A a Z : fichiers, sous-repertoires
- i=d+"/"+i : la variable (i) recoit la concaténation du 3 chaines de caractere (d,i et "/") .tel que (d) : repertoire courant , (i):un objet quelconque situe sous le rep (d) . et le "/" pour generer le nouveau chemin relatif a tester .
- if os.path.isdir(i): commencer une condition(combinaison de if et de else) pour tester a l'aide de la fonction standard `isdir(i)` du module OS si l'objet (i) est un sous-repertoire ou pas .
si oui on fait:
- R(i),print(i) : cette ligne contient deux instructions différentes : la premiere celle de `R(i)` : c est un appel recursif a la fonction predfinie (R)

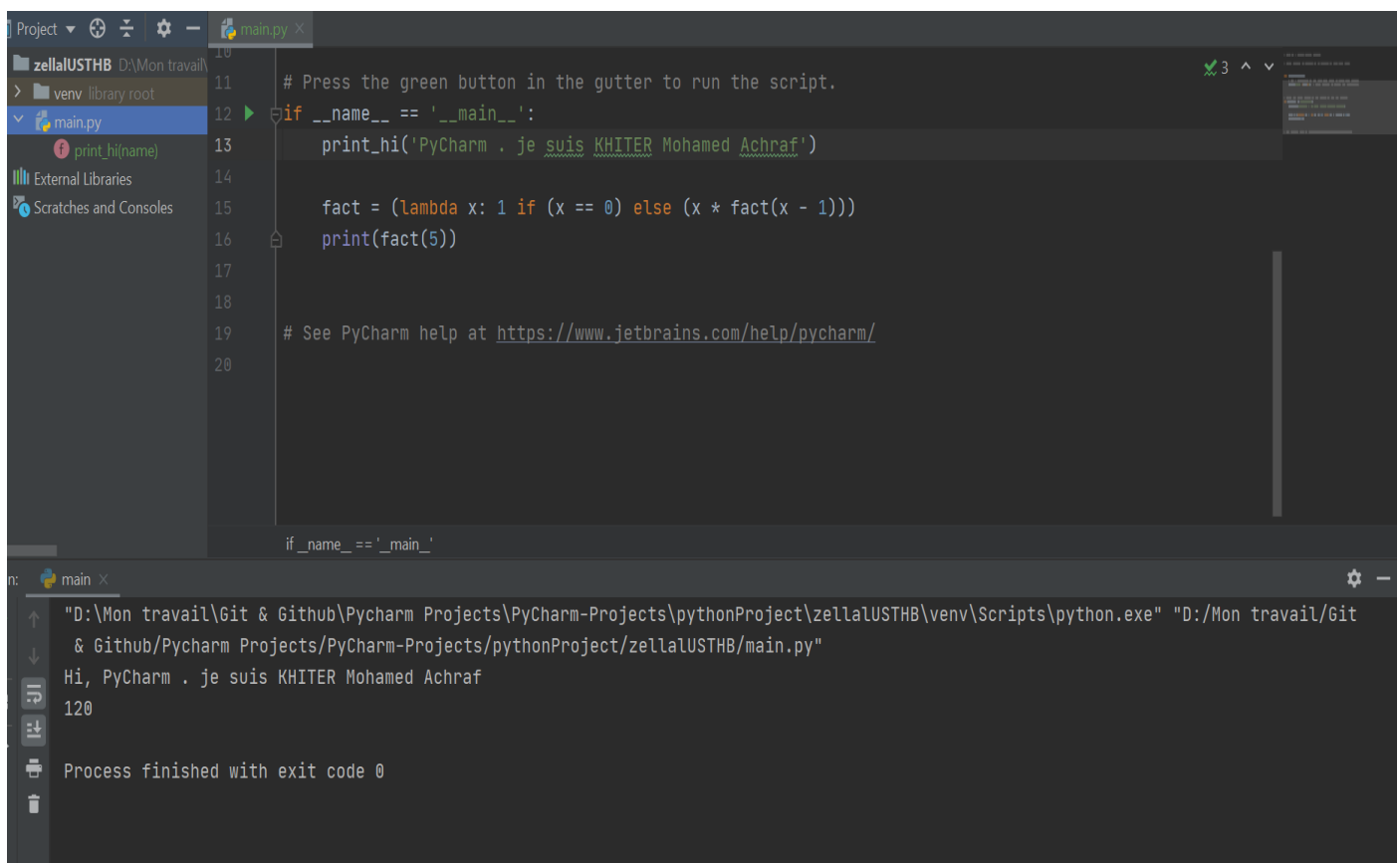
avec l'argument (i) . qui est dans ce cas un sous repertoire. pour la deuxième ligne `print(i)` . c est facile c est la fonction standard print qui nous aide a afficher un message dans la console .dans ce cas est le dernier sous-repertoire empiler dans la pile durant les appels recursifs,precisementle sous-repertoire qui ne contient plus des sous-repertoires, apres ca on commence a depiler .

- Else : sinon on fait ca :
- print(i) : afficher la variable (i) qui est une chaine de caractere du nom du fichiers testes sous un chemin relatif au repertoire principale .
- R(sys.argv[1]) : c est un appel a la fonction (R) avec un parametre d entree qui est un argument (I/O) numero 1 . il presente le repertoire de depart .

Exercice 03 :

1- reprendre le script ci-dessous, en supprimant les deux boucles « for » dans la partie en gras (partie du calcul de factorielle). Le calcul de factorielle doit passer de 7 à 2 lignes de code suivant :

- `factorielle = (lambda x: 1 if (x == 0) else (x * fact(x - 1)))`
- `[print(factorielle(i)) for i in [x-1 , y-1]]` # x et y : sont des nombres donnees .



The screenshot shows the PyCharm IDE with a Python script named `main.py` open. The script contains the following code:

```
10 # Press the green button in the gutter to run the script.
11
12 if __name__ == '__main__':
13     print_hi('PyCharm . je suis KHITER Mohamed Achraf')
14
15     fact = (lambda x: 1 if (x == 0) else (x * fact(x - 1)))
16     print(fact(5))
17
18
19 # See PyCharm help at https://www.jetbrains.com/help/pycharm/
20
```

The output console at the bottom shows the execution of the script:

```
main
"D:\Mon travail\Git & Github\Pycharm Projects\Pycharm-Projects\pythonProject\zellaLUSTHB\venv\Scripts\python.exe" "D:/Mon travail/Git & Github/Pycharm Projects/Pycharm-Projects/pythonProject/zellaLUSTHB/main.py"
Hi, PyCharm . je suis KHITER Mohamed Achraf
120
Process finished with exit code 0
```

2- Expliquer à quoi sert l'étoile * placée avant le « x » dans la ligne de code 10 :

Mettre (* args) comme dernier élément dans la liste d'arguments de notre définition de fonction permet à cette fonction d'accepter un nombre arbitraire d'arguments anonymes et / ou de mots-clés.

Ces arguments sont appelés arguments de mots-clés. En fait, ce sont des espaces réservés pour plusieurs arguments, et ils sont particulièrement utiles lorsque nous devons passer un nombre différent d'arguments à chaque fois que nous appelons la fonction.

{

et voilà l'ensemble des arguments entres se forment un tuple (tableau des valeurs) comme parametre d'entree . la meme chose avec la sortie . ca sera un tuple de resultats a parcourir.

}

Collecting arguments

```
>>> def f(*args):  
    print(args)  
  
>>> f()  
()  
>>> f(10)  
(10,)  
>>> f(10, 20, 30)  
(10, 20, 30)
```

Exercice 02 :

1- expliquer l'expression régulière, qui se trouve entre guillemets en gras (ligne de code 5)

- `x=re.search("[.!]??(. $)",i,re.S)`

- `[.!]??(. $)`

`[.!]` :

Faites correspondre un seul caractère présent dans la liste ci-dessous.

?? :

Quantificateur - Correspond entre zéro(0) et une fois(1), aussi peu de fois que possible, en s'étendant au besoin (paresseux) ,

1st Capturing Group (.\$) :

. correspond à n'importe quel caractère

\$ affirme sa position à la fin d'une ligne

REGEX FLAGS : s (re.S): faire (.) correspond également à la nouvelle ligne.

- en générale cette expression reguliere capture toujours le dernier caractère de chaque lignes(espace blanc " " inclu), et aussi capture l'avant dernier caractère si il est équivalent a (.) ou (!) comme on peut voir dans l'exemple dessous : # quand il capte un caractere il fait pas de saut de ligne dans l'affichage . si il capte rien (ligne vide) il fait un saut de ligne(\n)

The screenshot shows the regex101.com website with the following details:

- URL:** regex101.com/r/gJ1aH3/2/
- REGULAR EXPRESSION:** `[.!]??(.$)`
- TEST STRING:**

```
KHITER
khiter4$%^&***()_+!@
mohamed,
achraf..
Mr Zellaal...
USTHB!!!
!
123456
.
examan python .!.!
```
- Matches:** 13 matches, 469 steps (~2ms)
- Left Sidebar:**
 - LIVE & SHARE:** Update Regex (ctrl+s), Fork Regex, Delete Regex
 - FAVOR:** PCRE (PHP), ECMAScript (JavaScript), Python (checked), Golang
 - FUNCTION:** Match (checked), Substitution, Unit Tests
 - TOOLS:** Code Generator

Exercice 04 :

1- Expliquer la ligne de code 3 en gras du script ci-dessous.

- `for j in zip(*[re.split("\W+", open(sys.argv[1], 'r', encoding="utf-8").read().lower())[i:] for i in range(int(sys.argv[2]))]):`
- `d[j]=d.get(j,0)+1`

Cette ligne dans ce script elle va prendre à l'entrée un fichier ex : (test.txt) Comme (sys.argv[1]) et un nombre entier comme (sys.argv[2]) qui représente longueur de co-occurrence par ex (2 : ['khiter', 'achraf']) avec un espace blanc entre eux, alors cette portion de code génère un dictionnaire (cle, valeur) la cle représente un tuple des mots séparés par un espace blanc dans le texte principal. Le nombre des mots dans le tuple dépend de l'entier d'entrée (sys.argv[2]) par exemple si (sys.argv[2]) = 3 alors le tuple soit comme ça : ('khiter', 'mohamed', 'achraf') il prend les trois premiers mots séparés par un espace et calcule la fréquence de cette cooccurrence dans tout le fichier. Et retourner cette fréquence comme la valeur de la cle dans le dictionnaire. Après revenir au début et prendre la deuxième cooccurrence trouvée en avançant le curseur avec un mot. Car pour ça on va trouver plusieurs cooccurrences. Voici un exemple qui explique ça :

```
15 print_hi('PyCharm')
16
17 fact = (lambda x: 1 if (x == 0) else (x * fact(x - 1)))
18 [print(fact(i)) for i in [6,7]]
19
20 #try to compile a simple regex for Zella USTHB exam exo 4 :
21 d = {}
22 for j in zip(*[re.split("\W+", open(sys.argv[1], 'r', encoding="utf-8").read().lower())[i:] for i in range(int(sys.argv[2]))]):
23     d[j] = d.get(j, 0) + 1
24
25 print(type(d))
26 print(d)
27 # See PyCharm help at https://www.jetbrains.com/pycharm/
28
if __name__ == '__main__':
```

test.txt - Bloc-notes

khiter khiter khiterkhiter khiter khiter

```
{('khiter', 'khiter'): 3, ('khiter', 'khiterkhiter'): 2, ('khiterkhiter', 'khiter'): 2}
```

2- expliquer l'expression régulière, qui se trouve entre guillemets en gras (ligne de code 5).

- `"^\\w{" + sys.argv[4] + "} (.+)?\\w{" + sys.argv[5] + "}$"` :

Remarque : `sys.argv[4]` = longueur du 1^{er} token d'une cooccurrence et `sys.argv[5]` = longueur du dernier token d'une cooccurrence.

L'expression régulière va extraire : une chaîne de caractères qui commence par une sous chaîne (un mot) de longueur `argv[4]` et se termine par une autre sous chaîne (mot) de longueur `argv[5]` et entre ses 2 chaînes on peut trouver toutes autres caractères tout sauf l'espace blanc ou le saut de ligne

3- Modifier ce script, en supprimant la ligne de code 2 (création du dictionnaire vide « d »). Après modification, ce script doit passer de 5 à 4 lignes de code, tout en conservant les 3 boucles « for » et la condition « if » :

```

20 #try to compile a simple regex for Zellaal USTHB exam exo 4 :
21 d = {}
22 for j in zip(*[re.split("\\W+", open(sys.argv[1], 'r').read().lower())[i:] for i in range(int(sys.argv[4]) - 1)]):
23     d[j] = d.get(j, 0) + 1
24 [print(" ".join(k) + " " + str(d.get(k)) + "\\n") for k in sorted(d) if d.get(k) == int(sys.argv[5])]
25
26 # See PyCharm help at https://www.jetbrains.com/help/pycharm/
27
if __name__ == '__main__':

```

Terminal: Local × +

```

(venv) D:\Mon travail\Git & Github\Pycharm Projects\PyCharm-Projects\pythonProject\zellaalUSTHB>python main.py test.txt 3 3 3 5
Hi, PyCharm . je suis KHITER Mohamed Achraf
720
5040
sur des bases 3
sur la place 3
sur les bancs 3

```

execution du script

resultats

Le nouveau code est :

- import sys,re
- for j in zip(*[re.split("\W+",open(sys.argv[1],'r',encoding="utf-8").read().lower())[i:] for i in range(int(sys.argv[2]))]):
- d[j]=d.get(j,0)+1
- [print(" ".join(k)+" "+str(d.get(k))+"\n") for k in sorted(d) if d.get(k)==int(sys.argv[3]) and re.search("^\\w{"+sys.argv[4]+"}"(.+)?\\w{"+sys.argv[5]+"}\$"," ".join(k))] .

FIN .

MERCI POUR LA
Correction.