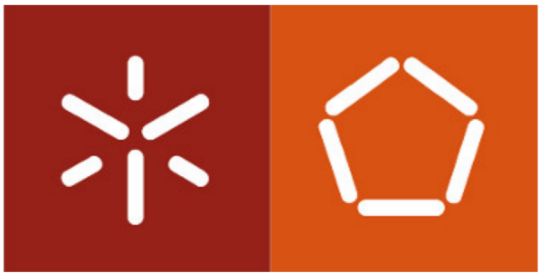

Universidade do Minho
Departamento de Informática

TP1

Processamento de Linguágens

27 de março, 2022

Índice

1. Introdução

O presente trabalho tem como objetivo desenvolver um processador de linguagens regulares, desenvolvendo expressões regulares que identifiquem padrões que permitam uma melhor organização do dataset, e a sua escrita em formato *JSON*.

Posto isto, como é evidenciado, o tema que escolhemos foi o "Ficheiros *CSV* com listas e funções de agregação", e iremos analisar o problema em questão, explicando as nossas decisões, bem como identificar todas as expressões regulares e estruturas de dados usadas para a resolução do mesmo.

2. Análise do problema

O conversor que nos foi proposto deve conseguir converter ficheiros no formato *CSV* para o formato *JSON*, efetuando uma análise da primeira linha do dataset recebido, pois esta funciona como cabeçalho, e a partir do resultado retornado pelo método *findall* da biblioteca *re* de *python*, com a seguinte expressão regular:

```
'\"?(([\w/]+)({(\d+,\d+|\d+|\d+,|,\d+)}(::(sum|media))?)?)\"?'
```

A expressão regular acima tem como objetivo identificar o padrão do *dataset* em *CSV*, identificando colunas simples, constituídas por um atributo, ou identificando colunas que deverão ser agrupadas numa só, de modo a facilitar a análise do *dataset*, como por exemplo, se o cabeçalho conter um campo com *Notas4* significa que na coluna correspondente a este campo e nas próximas 3 devemos agrupar os dígitos correspondentes numa lista. Também é possivel identificar intervalos de colunas, por exemplo *Contactos1,* ou *email,2*, ou ainda um intervalo concreto, por exemplo *Notas2,4*, sendo descartadas todas as linhas do ficheiro *CSV* que não cumpram com nenhum destes requisitos na coluna correspondente.

Complemente ao parsing de várias colunas para uma lista, é também possível efetuar opções de agregação, como a soma, identificada por *sum*, e a média, identificada por *media*, que serão detetadas pela expressão regular, e serão depois tratadas linha a linha. Devido à forma como o nosso código foi organizado, este apresenta uma grande flexibilidade, sendo fácil acrescentar novas operações sobre as listas, devendo apenas se acrescentar essa opção na expressão regular, e acrescentar algumas linhas no código apresentado a seguir:

```
# operation:
sum = False
media = False
if match[i][5] == "sum":
    sum = True
if match[i][5] == "media":
    media = True
s = 0
split = match[i][3].split(",")
num = 0
for j in range(0, int(split[1])):
    if line[i + j].isdigit():
        s += int(line[i + j])
        num += 1
    if num < int(split[0]):
        flag = False
    if media:
        s = s / num
    dic[match[i][1]] = s
```

3. Processamento do ficheiro

3.1 Parsing linha a linha

3.2 Estruturas de dados

3.3 Escrita em formato JSON

4. Testes realizados e resultados

id_aluno	nome	curso	tpc1	tpc2	tpc3	tpc4
a1	Aysha Melanie Gilberto	LEI	12	8	19	8
a2	Igor André Cantanhede	ENGFIS	12	16	18	20
a3	Laurénio Narciso	ENGFIS	8	14	15	14
a4	Jasnoor Casegas	LCC	14	20	17	11
a5	Tawseef Rebouças	ENGBIOM	13	14	13	17

Figura 4.1: Ficheiro sem operações sobre as colunas

```
[
    {
        "id_aluno": "a1",
        "nome": "Aysha Melanie Gilberto",
        "curso": "LEI",
        "tpc1": "12",
        "tpc2": "8",
        "tpc3": "19",
        "tpc4": "8"
    },
    {
        "id_aluno": "a2",
        "nome": "Igor André Cantanhede",
        "curso": "ENGFIS",
        "tpc1": "12",
        "tpc2": "16",
        "tpc3": "18",
        "tpc4": "20"
    },
    {
        "id_aluno": "a3",
        "nome": "Laurénio Narciso",
        "curso": "ENGFIS",
        "tpc1": "8",
        "tpc2": "14",
        "tpc3": "15",
        "tpc4": "14"
    },
    {
        "id_aluno": "a4",
        "nome": "Jasnoor Casegas",
        "curso": "LCC",
        "tpc1": "14",
        "tpc2": "20",
        "tpc3": "17",
        "tpc4": "11"
    },
    {
        "id_aluno": "a5",
        "nome": "Tawseef Rebouças",
        "curso": "ENGBIOM",
        "tpc1": "13",
        "tpc2": "14",
        "tpc3": "13",
        "tpc4": "17"
    },
```


Figura 4.2: Resultado de ficheiro sem operações sobre as colunas

id_aluno	nome	curso	notas{0	4}::sum
a1,"Aysha Melanie Gilberto","LEI",12,8,19,8				
a2,"Igor André Cantanhede","ENGFIS",12,16,18,20				
a3,"Laurénio Narciso","ENGFIS",8,14,15,14				
a4,"Jasnoor Casegas","LCC",14,20,17,11				
a5,"Tawseef Rebouças","ENGBIOM",13,14,13,17				

Figura 4.3: Ficheiro com a operação de soma

```
[
    {
        "id_aluno": "a1",
        "nome": "Aysha Melanie Gilberto",
        "curso": "LEI",
        "notas": "47"
    },
    {
        "id_aluno": "a2",
        "nome": "Igor André Cantanhede",
        "curso": "ENGFIS",
        "notas": "66"
    },
    {
        "id_aluno": "a3",
        "nome": "Laurénio Narciso",
        "curso": "ENGFIS",
        "notas": "51"
    },
    {
        "id_aluno": "a4",
        "nome": "Jasnoor Casegas",
        "curso": "LCC",
        "notas": "62"
    },
    {
        "id_aluno": "a5",
        "nome": "Tawseef Rebouças",
        "curso": "ENGBIOM",
        "notas": "57"
    },
```


Figura 4.4: Resultado de ficheiro com operações sobre as colunas

id_aluno	nome	curso	notas{0	4}::media
a1,"Aysha Melanie Gilberto","LEI",12,8,19,8				
a2,"Igor André Cantanhede","ENGFIS",12,16,18,20				
a3,"Laurénio Narciso","ENGFIS",8,14,15,14				
a4,"Jasnoor Casegas","LCC",14,20,17,11				
a5,"Tawseef Rebouças","ENGBIOM",13,14,13,17				

Figura 4.5: Ficheiro com a operação de média


```
[
    {
        "id_aluno": "a1",
        "nome": "Aysha Melanie Gilberto",
        "curso": "LEI",
        "notas": "11.75"
    },
    {
        "id_aluno": "a2",
        "nome": "Igor André Cantanhede",
        "curso": "ENGFIS",
        "notas": "16.5"
    },
    {
        "id_aluno": "a3",
        "nome": "Laurénio Narciso",
        "curso": "ENGFIS",
        "notas": "12.75"
    },
    {
        "id_aluno": "a4",
        "nome": "Jasnoor Casegas",
        "curso": "LCC",
        "notas": "15.5"
    },
    {
        "id_aluno": "a5",
        "nome": "Tawseef Rebouças",
        "curso": "ENGBIOM",
        "notas": "14.25"
    },
```


Figura 4.6: Resultado de ficheiro com a operação de média

5. Comentários finais e conclusão

Após a conclusão do trabalho, o grupo apresenta-se satisfeito com o resultado obtido. A ferramenta desenvolvida é capaz de ler os vários ficheiros CSV recebidos, assim como passar a informação dos mesmos para JSON. Consideramos, assim, que é aplicado o conhecimento adquirido até ao momento da unidade curricular de Processamento de Linguagens.