

```

1  ''' Lycée Vaugelas MP 2023-2024
2      TP 2 : dictionnaires
3      12 septembre 2023
4  '''
5
6  # ----- Exercice 1 -----
7  def comptage(s):
8      d = {}
9      for c in s:
10         if c in d:
11             d[c] += 1
12         else:
13             d[c] = 1
14     return d
15
16 print(comptage('Je programme donc je suis'))
17 print(comptage([2,3,5,2,1,6,8,4,5,9,7,6,3,2,5,0,8,7,41,2,3,6]))
18
19 # ----- Exercice 2 -----
20 def eleves(d):
21     dico = {}
22     for num, (e1,e2,e3) in d.items():
23         dico[e1] = dico[e2] = dico[e3] = num
24     return dico
25
26 def classe(d):
27     dico = eleves(d)
28     t = []
29     for nom, num in dico.items():
30         t.append((nom,num))
31     t.sort()
32     return t
33
34 d = {4:('Durand', 'Martin', 'Dupont'), 2:('Petit', 'Dubois', 'Dumoulin'), 10:(
    'Richard',
35     'Thomas', 'Roux'), 7:('Bernard', 'Robert', 'Simon')}
36 print(eleves(d))
37 print(classe(d))
38
39 # ----- Exercice 3 -----
40 points = {}
41 groupes = 'AEILNORSTU', 'DGM', 'BCP', 'FHV', 'JQ', 'KWXYZ'
42 val = 1,2,3,4,8,10
43 for i, g in enumerate(groupe):
44     for lettre in g:
45         points[lettre] = val[i]
46
47 def score(mot):
48     s = 0
49     for c in mot:
50         s += points[c.upper()]
51     return s
52
53 print(score('Vaugelas'))
54
55 # ----- Exercice 4 -----
56 def deriv(p):
57     d = {}
58     for deg, coef in p.items():
59         if deg != 0:
60             d[deg - 1] = deg * coef
61     return d or {0:0}
62
63 print(deriv({0:8,1:3,4:-7}))
64 print(deriv({1:5}))
65 print(deriv({0:5}))
66
67 # ----- Exercice 5 -----
68 def occurrence(t):
69     d = {}
70     for c in t:
71         if c in d:

```

```
72         d[c] += 1
73     else:
74         d[c] = 1
75     return d
76
77 def compare(t1,t2):
78     d = occurrence(t1)
79     for x in t2:
80         if x not in d: return False
81         if d[x] == 1: del d[x]
82         else: d[x] -= 1
83     return not d
84
85 print(compare([1,2,3,2,1],[3,2,2,1,1]))
86 print(compare([1,2,3,2,1],[3,2,2,2,1,1]))
87 print(compare([1,2,3,2,1],[3,2,1,1,1]))
88
89 # La complexité est linéaire alors qu'on ne fait pas mieux que n.log(n) en triant
90
91
92
93
```