

**Instituto de
Computação**

UNIVERSIDADE ESTADUAL DE CAMPINAS



MC102 - Aula 11

Exemplos: Strings, Sets e Dicionários

Algoritmos e Programação de Computadores

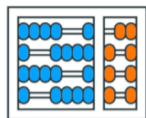
Turmas
OVXZ

Prof. Lise R. R. Navarrete

lrommel@ic.unicamp.br

Quinta-feira, 28 de abril de 2022

19:00h - 21:00h (CB06)



**Instituto de
Computação**

UNIVERSIDADE ESTADUAL DE CAMPINAS



UNICAMP

MC102 – Algoritmos e Programação de Computadores

Turmas

OVXZ

<https://ic.unicamp.br/~mc102/>

Site da Coordenação de MC102

Aulas teóricas:

Terça-feira, 21:00h - 23:00h (CB06)

Quinta-feira, 19:00h - 21:00h (CB06)

Conteúdo

- Exemplo 1
- Exemplo 2
- Exemplo 3
- Exemplo 4
- Exemplo 5

Exemplo 1

Exemplo 1

Dada uma lista de letras,
criar uma lista aleatoria contendo 5 cadeias
tal que, cada cadeia está formada por tais letras e
cada cadeia deve ter no máximo 20 letras.

exemplo01.py > ...

<https://shell.cloud.google.com/>

```
1 import random
```

```
2
```

```
3 print(random.random())
```

```
$ python3 exemplo01.py
```

```
0.807475549991721
```

```
$
```

exemplo01.py > {} random

<https://shell.cloud.google.com/>

```
1 import random
2
3 mylist = ["apple", "banana", "cherry"]
4
5 print(random.choice(mylist))
6
```

cloudshell x + ▾

```
$ python3 exemplo01.py
cherry
$ █
```

exemplo01.py > {} random

<https://shell.cloud.google.com/>

```
1 import random
2
3 letras = "abcdefghijklmnopqrstuvwxy"
4
5 print(random.choice(letras))
6
```

cloudshell X + ▾

```
$ python3 exemplo01.py
s
$ python3 exemplo01.py
f
$ █
```


exemplo01.py > ...

<https://shell.cloud.google.com/>

```
1 import random
2
3 letras = "abc"
4
5 print(random.choices(letras, k = 10))
```

cloudshell X + ▾

```
$ python3 exemplo01.py
['b', 'b', 'b', 'b', 'a', 'b', 'a', 'b', 'c', 'a']
$
```

exemplo01.py > ...

<https://shell.cloud.google.com/>

```
1 import random
2
3 letras = "abc"
4
5 print("".join(random.choices(letras, k = 10)))
```

cloudshell X + ▾

```
$ python3 exemplo01.py
acbcccbbca
$ █
```

exemplo01.py > ...

<https://shell.cloud.google.com/>

```
1 import random
2
3 letras = "abcdefghijklmnopqrstuvwxy"
4
5 print("".join(random.choices(letras, k = 10)))
```

cloudshell X + ▾

```
$ python3 exemplo01.py
dywhqckayd
$
```

exemplo01.py > ...

<https://shell.cloud.google.com/>

```
1 import random
2
3 letras = "abcdefghijklmnopqrstuvwxy"
4
5 print("".join(random.choices(letras, k = 10)))
```

cloudshell X + ▾

```
$ python3 exemplo01.py
vtmontxfs
$ █
```

exemplo01.py > ...

<https://shell.cloud.google.com/>

```
1 import random
2
3 letras = "abcdefghijklmnopqrstuvwxyz"
4
5 print("".join(random.choices(letras, k = random.choice(range(10)))))
```

cloudshell X + ▾

```
$ python3 exemplo01.py
jugwwb
$ python3 exemplo01.py
orazv
$ █
```

exemplo01.py > ...

<https://shell.cloud.google.com/>

```
1 import random
2
3 letras = "abcdefghijklmnopqrstuvwxy"
4
5 print("".join(random.choices(letras, k = random.choice(range(20)))))
```

cloudshell X + ▾

```
$ python3 exemplo01.py
ercsfz xu
$ python3 exemplo01.py
oiy
$
```

exemplo01.py > {} random

<https://shell.cloud.google.com/>

```
1 import random
2
3 letras = "abcdefghijklmnopqrstuvwxy"
4
5 print(["".join(random.choices(letras, k = random.choice(range(20)))) for i in range(5)])
```

cloudshell X + ▾

```
$ python3 exemplo01.py
['xkwpon', '', 'gujpb', 'wyikzycqdvu', 'ekylfhx']
$ python3 exemplo01.py
['ofnsjkj', '', 'nqbejnuns', 'szipllzescaaqntlruq', 'olxgsvtjhwtea']
$
```

exemplo01.py > ...

<https://shell.cloud.google.com/>

```
1 import random
2
3 letras = "abcdefghijklmnopqrstuvwxy"
4
5 P = ["".join(random.choices(letras, k = random.choice(range(20)))) for i in range(5)]
6
```

cloudshell X + ▾

```
$ python3 exemplo01.py
['cabmmowdpyebp', 'whygsxwfaqyjn', 'lhfinbpigehrdkattyf', 'bfdrojem', 'dbgdzko']
$
```


Exemplo 2

Exemplo 2

Dada uma lista de cadeias,
criar uma lista com as letras usadas
e
contar quantas vezes foi usada cada uma dessas letras.

exemplo01.py > ...

<https://shell.cloud.google.com/>

```
1 import random
2 letras = "abcdefghijklmnopqrstuvwxy"
3 P = ["".join(random.choices(letras, k = random.choice(range(20)))) for i in range(100)]
4
5 print(P)
```

cloudshell X + ▾

```
pikbrodmc', 'y', 'idhsvqmcryxswmbpua', 'tbtaxafqvuwjrwp', 'vexqfcbulwhovhmsjo',
'odwpfxfsdcdvydnouw', 'gyocijqtncrerl', 'ujfgzv', 'cxbufsmtdddpzv', 'rqchmiimouw
p', 'tqjze', 'fipvojbxz', 'j', 'rdthbwswdcomveawmxc', 'qbqllpavet', 'bggyvbwlgls
rba', 'nmaoquevcnpl', 'godaykepgelnuiqnegy', 'brmdlyuxxn', 't', 'ahlklsfansu', '
qzxv', '', 'zprgre', 'fgfewro', 'jexwushurxxizj', 'xtj', 'lxvvexkofjfgkc']
$
```

exemplo01.py > ...

<https://shell.cloud.google.com/>

```
1 import random
2 letras = "abcdefghijklmnopqrstuvwxy"
3 P = ["".join(random.choices(letras, k = random.choice(range(20)))) for i in range(100)]
4
5 print(P)
```

cloudshell X + v

```
wmaafxoy', 'lngpezvm', 'pudi', 'ovaxpzbh', 'vkldxslrskvtetydct', 'sfyaqfgnoadv
tlclfo', 'bhmqolgkzvgvsgh', 'kahisesxhnjofwsdv', 'uafacdlleo', 'u', 'mrqcf', '
bh', 'gactyqrvidwtwgeid', 'jjlv', 'wiotzdn', 'nadhbeurkqkqns', 'pzdskaqfgcky',
'lwolghdb', 'mdxbvzqugqbftlmf', 'pms', 'cyjtidgsooaheg', 'vxkpkfkzobtdtfmsy', '
zzlagqpl', 'aprthapaepoppebmy', 'kucrx', 'jkhmzb']
```

\$

exemplo01.py > ...

<https://shell.cloud.google.com/>

```
1 import random
2 letras = "abcdefghijklmnopqrstuvwxyz"
3 P = ["".join(random.choices(letras, k = random.choice(range(20)))) for i in range(100)]
4 print(P)
5
6 letras_usadas = set()
7 print(letras_usadas)
8
```

cloudshell X + ▾

```
pklbfdxneviuve', 'mrrnftgh', 'snawmpvunwobhfxb', 'ugrprocu', 'wfkmdwkypqylabql'
, 'ylzzw', 'wzovfqp', 'yqqevbb', 'llgyedsdwmym', 'fqfcpkxkasrjyoon', 'voezpl', '
cu', 'vikmkf', 'shf', 'gddxfgo', 'eldlqpykvtxboc', 'u', 'sxjijatwtiqwpcg', 'dkop
nqv']
set()
$
```

exemplo01.py > ...

<https://shell.cloud.google.com/>

```
5
6 letras_usadas = set()
7 for palavra in P:
8     letras_usadas.update(palavra)
9
10 print(letras_usadas)
11
12
```

cloudshell X + ▾

X

```
czdo', 'mfbxvjwe', '', 'hkvurfrkbovk', 'useqapjdgzmexths', 'wtusoiukycemiykm',
'osrrakdahxrsaimrate', 'rkfkrq', 'zw', 'wbn', 'dxhwueryvanjxubv', '', 'kyehkxrh
cctfntkexjv', 'yrttblfox', 'ra']
{'m', 'x', 'l', 'c', 'g', 'e', 'u', 't', 'h', 'w', 'y', 'd', 'f', 'z', 'a', 'q',
 'i', 'o', 'b', 'p', 'v', 'r', 'j', 's', 'k', 'n'}
$
```

exemplo01.py > ...

```
5
6 letras_usadas = set()
7 for palavra in P:
8     letras_usadas.update(palavra)
9
10 print(list(letras_usadas))
11
12
```

<https://shell.cloud.google.com/>

cloudshell X + ▾

```
db', '', 'yicgmbdxvgrwlyutib', 'dchipjwnnjcmrqjci', 'ysnjpexhnp', 'wawzihxdbn',
'mlzlbbhzmcbbggbbcyb', 'zl', 'lfjnfwdyy', 'uvs', 'iup', 'slrxhzhjmhbw', 'btmmh',
'uwhbwgxo', 'rystrkdyrtbahmh', 'hun', 'kopklacuhrs', 'wb', 'fqnqnldrdyq']
['a', 'j', 'w', 'u', 'l', 'n', 'c', 'd', 'e', 't', 'r', 'i', 'z', 'b', 'g', 'x',
'v', 'o', 'k', 's', 'y', 'p', 'h', 'f', 'q', 'm']
$
```

exemplo01.py > ...

```
5
6 letras_usadas = set()
7 for palavra in P:
8     letras_usadas.update(palavra)
9
10 print(list(letras_usadas).sort())
11
12
```

<https://shell.cloud.google.com/>

cloudshell X + ▾

```
iodoisam', 'jnmnhzjg', 'rnh', 'kmnor', '', 'qwvfjhirom', 'ujjqtht', 'lwaym', 'ix
xjofceynyy', 'qkehkw', 'jcecewfcifbxn', 'wddxrwndbvjjyhi', 'lxmviulgppylozpf', '
sbedsgnmrlzzxsufs', 'cdos', 'f', 'lnikwhgd', 'agastqfridvbdtmlcw', '', 'joglgqjc
h', 'vgqeycrrocvsfbfpcp']
None
$ █
```


exemplo01.py > ...

```
5
6 letras_usadas = set()
7 for palavra in P:
8     letras_usadas.update(palavra)
9
10 print(sorted(list(letras_usadas)))
11
12
```

<https://shell.cloud.google.com/>

cloudshell X + ▾

```
fbhqicolaeeyh', 'rxbfkutksiehhgk', 'eycuwbwpaquvjxz', 'pm', 'sj', 'zwelzpvsv', 'w
ovqmmzkvehia', 'pa', 'bxxothjxfd', 'sgvwphjhlkmo', 'nnprnujchceeki', 'eofofrsalr
myamdkr', '', 'lxholqkwc', 'cgpkrgrwnnogecrlevb']
['a', 'b', 'c', 'd', 'e', 'f', 'g', 'h', 'i', 'j', 'k', 'l', 'm', 'n', 'o', 'p',
 'q', 'r', 's', 't', 'u', 'v', 'w', 'x', 'y', 'z']
$
```

exemplo01.py > ...

<https://shell.cloud.google.com/>

```
5
6 letras_usadas = set()
7 for palavra in P:
8     letras_usadas.update(palavra)
9
10 print("".join(sorted(list(letras_usadas))))
11
12
```

cloudshell X + ▾

```
'oxrqrtgqiww', 'vxcienkyuseyymapvqi', 'vdzfyo', 'kfjzuzxjnickjamrh', 'tdowi', '
', 'z', 'gsro', 'fsmcdqvheidhzkt', 'sbqgtnehcgqlra', 'w', 'eoouitlvr', 'oclwzdw
j', 'didsg', 'gkoclkhmbe', 'cbjisqmiqochgvces', 'yzxoxakaucnkdpksbi', 'xxuaxwfs
jvb', 'vdvhothypii', 'dcdvablrxhdfkz', 'cf', 'jzytoknqstwyldmgoi']
abcdefghijklmnopqrstuvwxyz
$
```

exemplo01.py > ...

<https://shell.cloud.google.com/>

```
5
6 letras_usadas = set()
7 for palavra in P:
8     letras_usadas.update(palavra)
9
10 print("".join(sorted(list(letras_usadas))))
11
12
```

cloudshell X + ▾

```
u', 'zjl', 'kva', 'zpy', 'itsca', 'qjntbhiqnyej', 'fwkxxlqkopyn', 'q', '', 'ysod
zxylcxyfu', 'sjuottrlidadmp', 'koevwcyot', 'lmqktardte', 'zbpmtnvut', 'bqgjnvad
jl', 'eitlgmmvqjeb', 'sdogiyurchlgmd', 'lfqxirhoyemrmeqnbez', 'v', 'dnffijohvof
mewdzmry', 'wed', 'bfqzllpvqy', 'itvxyjmyewpmdqwhzk']
abcdefghijklmnopqrstuvwxyz
$
```

exemplo01.py > ...

<https://shell.cloud.google.com/>

```
5
6 letras_usadas = set()
7 for palavra in P:
8     letras_usadas.update(palavra)
9
10 contador = dict(zip(letras_usadas,[0 for x in range(len(letras_usadas))]))
11 print(contador)
12
```

cloudshell X + ▾

```
h', 'pbzorzlsgt', 'vuiiuh', 'zeykmvxwrejsx', 'myicnvpvpmuflpr', 'nqvahb', 'r', 'p
mhqynkdtbm']
{'g': 0, 'h': 0, 'x': 0, 'r': 0, 's': 0, 'v': 0, 'u': 0, 'l': 0, 'k': 0, 'w': 0,
 'i': 0, 'y': 0, 'n': 0, 'a': 0, 'm': 0, 'e': 0, 'b': 0, 'd': 0, 'q': 0, 'p': 0,
 'j': 0, 'o': 0, 'c': 0, 'z': 0, 'f': 0, 't': 0}
$
```

exemplo01.py > ...

<https://shell.cloud.google.com/>

```
y
10 contador = dict(zip(letras_usadas,[0 for x in range(len(letras_usadas))]))
11
12 for palavra in P:
13     for x in palavra:
14         contador[x] += 1
15 print(contador)
16
```

cloudshell X + ▾

```
fduzi', 'eetnauizyhnjmlhs', 'zvxg', 'ecelbysvddrxdj', 'dsiqlygijubpgp', 'ksk', '
pbqwsci', 'mzalfzw', 'flgphycgf', 'kxuofgxuduqbvvuptaq', 'pdd', 'j', 'noe', '']
{'z': 34, 'o': 35, 't': 32, 'i': 41, 'k': 35, 'm': 34, 'f': 34, 'q': 28, 'w': 42
, 'u': 31, 'n': 39, 'b': 36, 'd': 28, 'h': 40, 'v': 49, 's': 48, 'g': 38, 'c': 3
6, 'r': 42, 'l': 55, 'e': 36, 'x': 37, 'y': 34, 'j': 30, 'p': 38, 'a': 27}
$
```

exemplo01.py > ...

<https://shell.cloud.google.com/>

```
y
10 contador = dict(zip(letras_usadas,[0 for x in range(len(letras_usadas))]))
11
12 for palavra in P:
13     for x in palavra:
14         contador[x] += 1
15 print(contador)
16
```

cloudshell X + ▾

```
j', 'v', 'g', 'jnkyhj', 'alzwognwd', 'zouvujonnakguxjexj', 'rvfidrs', 'nzfqfvkei
yfphgopxl']
{'c': 39, 'a': 44, 'v': 40, 'd': 40, 'y': 35, 'b': 31, 'k': 41, 'h': 40, 'e': 39
, 'p': 51, 'm': 34, 'g': 37, 'n': 37, 't': 39, 'z': 42, 'i': 39, 's': 47, 'r': 3
4, 'j': 43, 'o': 39, 'u': 47, 'f': 36, 'l': 50, 'w': 44, 'x': 43, 'q': 40}
$
```

Exemplo 3

Exemplo 3

Dado um dicionário contendo as ocorrências de determinados símbolos,
mostrar as ocorrências de tais símbolos ordenados em
forma alfabética.

exemplo01.py > ...

<https://shell.cloud.google.com/>

```
y
10 contador = dict(zip(letras_usadas,[0 for x in range(len(letras_usadas))]))
11
12 for palavra in P:
13     for x in palavra:
14         contador[x] += 1
15 print(contador)
16
```

cloudshell X + ▾

```
qntjta', 'jpl', 'zjsg', 'cvgfdgox', 'ttsfuxnemrlj', 'yhczhgnyhlw', 'prxmmlwru',
'tnkczipusrd', 'gooqgkyzbdjcfjwyx', 'kco', 'ffjmpaxhdpsjh']
{'j': 51, 'c': 39, 'f': 37, 'd': 38, 'q': 26, 'h': 44, 'a': 34, 'z': 37, 'b': 35
, 'l': 39, 'n': 46, 'u': 37, 'g': 34, 'w': 42, 'k': 36, 'v': 40, 'r': 36, 'o': 2
6, 'p': 34, 'y': 33, 'i': 30, 'm': 29, 't': 38, 'e': 24, 'x': 37, 's': 37}
$
```

exemplo01.py > ...

<https://shell.cloud.google.com/>16
17
18
19
20

cloudshell X +

X

```
jylnwnqwepqd', 'gaiqvjnryyn', 'rienym', '']  
{'t': 31, 's': 25, 'g': 36, 'd': 35, 'm': 24, 'y': 35, 'k': 58, 'r': 40, 'a': 35  
, 'x': 32, 'l': 31, 'b': 32, 'e': 36, 'c': 44, 'p': 31, 'i': 34, 'h': 36, 'j': 2  
7, 'n': 48, 'w': 39, 'o': 33, 'q': 43, 'z': 32, 'u': 41, 'f': 40, 'v': 25}  
a:35 b:32 c:44 d:35 e:36 f:40 g:36 h:36 i:34 j:27 k:58 l:31 m:24 n:48 o:33 p:31  
q:43 r:40 s:25 t:31 u:41 v:25 w:39 x:32 y:35 z:32 $
```

exemplo01.py > ...

<https://shell.cloud.google.com/>

```
16
17 for x in sorted(letras_usadas):
18     print("{}:{}".format(x,contador[x]), end=" ")
19
20
21
22
```

cloudshell X + ▾

X

```
jylnwnqwepqd', 'gaiqvjnryyn', 'rienym', '']
{'t': 31, 's': 25, 'g': 36, 'd': 35, 'm': 24, 'y': 35, 'k': 58, 'r': 40, 'a': 35
, 'x': 32, 'l': 31, 'b': 32, 'e': 36, 'c': 44, 'p': 31, 'i': 34, 'h': 36, 'j': 2
7, 'n': 48, 'w': 39, 'o': 33, 'q': 43, 'z': 32, 'u': 41, 'f': 40, 'v': 25}
a:35 b:32 c:44 d:35 e:36 f:40 g:36 h:36 i:34 j:27 k:58 l:31 m:24 n:48 o:33 p:31
q:43 r:40 s:25 t:31 u:41 v:25 w:39 x:32 y:35 z:32 $
```

exemplo01.py > ...

```
18
19 for x in sorted(letras_usadas):
20     print("{}:{}".format(x,contador[x]), end=" ")
21
22
23
24
```

<https://shell.cloud.google.com/>

cloudshell X + ▾

```
{'j': 16, 'v': 41, 'x': 44, 'h': 39, 'u': 39, 'y': 44, 'n': 34, 'e': 37, 'a': 49
, 'z': 38, 's': 41, 'q': 36, 'c': 42, 'b': 29, 'o': 36, 'r': 42, 'f': 43, 'l': 3
6, 'd': 30, 'p': 38, 'm': 28, 't': 62, 'w': 43, 'k': 42, 'i': 37, 'g': 42}
```

```
a:49 b:29 c:42 d:30 e:37 f:43 g:42 h:39 i:37 j:16 k:42 l:36 m:28 n:34 o:36 p:38
q:36 r:42 s:41 t:62 u:39 v:41 w:43 x:44 y:44 z:38 $
```

Exemplo 4

Exemplo 4

Temos uma agenda velha armazenada num dicionário, o qual guarda o par:

'Nome': 'telefone'

Precisamos adicionar o prefixo "9" a cada número telefônico. Isto pode ser feito de forma "simples"?

<https://shell.cloud.google.com/>

exemplo04.py > ...

```
1 agenda = {'Elena': '5534-5178', 'Henry': '1234-0987', 'Carl': '6578-4321',  
2 | 'Lucas': '8485-0645', 'Lena': '8484-1256', 'Lewis': '7539-8671'}  
3  
4  
5  
6 print(agenda)
```

cloudshell X + ▾

```
$ python3 exemplo04.py  
{'Elena': '5534-5178', 'Henry': '1234-0987', 'Carl': '6578-4321', 'Lucas': '8485-  
-0645', 'Lena': '8484-1256', 'Lewis': '7539-8671'}  
$ █
```

<https://shell.cloud.google.com/>

exemplo04.py > ...

```
1 agenda = {'Elena': '5534-5178', 'Henry': '1234-0987', 'Carl': '6578-4321',
2 | 'Lucas': '8485-0645', 'Lena': '8484-1256', 'Lewis': '7539-8671'}
3 print(agenda)
4
5 for x in agenda.keys():
6 |     agenda[x] = "9" + agenda[x]
7 print(agenda)
```

cloudshell X + ▾

```
$ python3 exemplo04.py
{'Elena': '5534-5178', 'Henry': '1234-0987', 'Carl': '6578-4321', 'Lucas': '8485-0645', 'Lena': '8484-1256', 'Lewis': '7539-8671'}
{'Elena': '95534-5178', 'Henry': '91234-0987', 'Carl': '96578-4321', 'Lucas': '98485-0645', 'Lena': '98484-1256', 'Lewis': '97539-8671'}
$
```


<https://shell.cloud.google.com/>

exemplo04.py > ...

```
1 agenda = {'Elena': '5534-5178', 'Henry': '1234-0987', 'Carl': '6578-4321',  
2           'Lucas': '8485-0645', 'Lena': '8484-1256', 'Lewis': '7539-8671'}  
3 print(agenda)  
4  
5 agenda = { x:"9" + agenda[x] for x in agenda.keys() }  
6  
7 print(agenda)
```

cloudshell X + ▾

```
$ python3 exemplo04.py  
{'Elena': '5534-5178', 'Henry': '1234-0987', 'Carl': '6578-4321', 'Lucas': '8485-  
-0645', 'Lena': '8484-1256', 'Lewis': '7539-8671'}  
{'Elena': '95534-5178', 'Henry': '91234-0987', 'Carl': '96578-4321', 'Lucas': '9  
8485-0645', 'Lena': '98484-1256', 'Lewis': '97539-8671'}  
$
```

Exemplo 5

Exemplo 5

Temos uma agenda armazenada num dicionário,
o qual guarda o par: 'Nome': 'CPF'
Precisamos normalizar os CPFs,
dado que foram armazenados sem respeitar o formato:

123.456.789-00

Isto pode ser feito de forma "simples"?

<https://shell.cloud.google.com/>

exemplo05.py > ...

```
1 agenda = {'Elena': '65788762890', 'Henry': '12348720987', 'Carl': '657.899.843.21',  
2           'Lucas': '111 848 506-45', 'Lena': '848-125-602.21', 'Lewis': '753.986.710-21'}  
3 print(agenda)  
4  
5
```

cloudshell X + ▾

```
$ python3 exemplo05.py  
{'Elena': '65788762890', 'Henry': '12348720987', 'Carl': '657.899.843.21', 'Luca  
s': '111 848 506-45', 'Lena': '848-125-602.21', 'Lewis': '753.986.710-21'}  
$
```

<https://shell.cloud.google.com/>

exemplo05.py > ...

```
1 agenda = { 'Elena': '65788762890', 'Henry': '12348720987', 'Carl': '657.899.843.21',  
2           'Lucas': '111 848 506-45', 'Lena': '848-125-602.21', 'Lewis': '753.986.710-21'}  
3 print(agenda)  
4  
5 for x in agenda.keys():  
6     agenda[x] = agenda[x]  
7 print(agenda)
```

cloudshell X + ▾

```
$ python3 exemplo05.py  
{'Elena': '65788762890', 'Henry': '12348720987', 'Carl': '657.899.843.21', 'Luca  
s': '111 848 506-45', 'Lena': '848-125-602.21', 'Lewis': '753.986.710-21'}  
{'Elena': '65788762890', 'Henry': '12348720987', 'Carl': '657.899.843.21', 'Luca  
s': '111 848 506-45', 'Lena': '848-125-602.21', 'Lewis': '753.986.710-21'}  
$
```

<https://shell.cloud.google.com/>

exemplo05.py > ...

```
1 agenda = {'Elena': '65788762890', 'Henry': '12348720987', 'Carl': '657.899.843.21',  
2 'Lucas': '111 848 506-45', 'Lena': '848-125-602.21', 'Lewis': '753.986.710-21'}  
3 print(agenda['Lena'])
```

cloudshell X + ▾

```
$ python3 exemplo05.py  
848-125-602.21  
$
```

<https://shell.cloud.google.com/>

exemplo05.py > ...

```
1 agenda = {'Elena': '65788762890', 'Henry': '12348720987', 'Carl': '657.899.843.21',  
2           'Lucas': '111 848 506-45', 'Lena': '848-125-602.21', 'Lewis': '753.986.710-21'}  
3 print(agenda['Lena'])  
4  
5 tmp = [x for x in agenda['Lena'] if x in "0123456789"]  
6 print(tmp)
```

cloudshell X + ▾

```
$ python3 exemplo05.py  
848-125-602.21  
['8', '4', '8', '1', '2', '5', '6', '0', '2', '2', '1']  
$
```

<https://shell.cloud.google.com/>

exemplo05.py > ...

```
1 agenda = {'Elena': '65788762890', 'Henry': '12348720987', 'Carl': '657.899.843.21',  
2 | 'Lucas': '111 848 506-45', 'Lena': '848-125-602.21', 'Lewis': '753.986.710-21'}  
3 print(agenda['Lena'])  
4  
5 tmp = "".join([x for x in agenda['Lena'] if x in "0123456789"])  
6 print(tmp)
```

cloudshell X + ▾

```
$ python3 exemplo05.py  
848-125-602.21  
84812560221  
$ █
```


<https://shell.cloud.google.com/>

exemplo05.py > ...

```
1 agenda = {'Elena': '65788762890', 'Henry': '12348720987', 'Carl': '657.899.843.21',
2           'Lucas': '111 848 506-45', 'Lena': '848-125-602.21', 'Lewis': '753.986.710-21'}
3 print(agenda['Lena'])
4
5 tmp = "".join([x for x in agenda['Lena'] if x in "0123456789"])
6 print("{}.{}.{}-{}".format(tmp[0:3],tmp[3:6],tmp[6:9],tmp[9:11]))
```

cloudshell X + ▾

```
$ python3 exemplo05.py
848-125-602.21
848.125.602-21
$ █
```

<https://shell.cloud.google.com/>

exemplo05.py > ...

```
4
5 tmp = [x for x in agenda['Lena'] if x in "0123456789"]
6 tmp.insert(3, ".")
7 tmp.insert(7, ".")
8 tmp.insert(11, "-")
9 print(tmp)
```

cloudshell X + ▾

```
$ python3 exemplo05.py
848-125-602.21
['8', '4', '8', '.', '1', '2', '5', '.', '6', '0', '2', '-', '2', '1']
$
```

<https://shell.cloud.google.com/>

exemplo05.py > ...

```
4
5 tmp = [x for x in agenda['Lena'] if x in "0123456789"]
6 tmp.insert(3, ".")
7 tmp.insert(7, ".")
8 tmp.insert(11, "-")
9 print("".join(tmp))
```



cloudshell X + ▾



```
$ python3 exemplo05.py
848-125-602.21
848.125.602-21
$ █
```

<https://shell.cloud.google.com/>

exemplo05.py > ...

```
1 agenda = {'Elena': '65788762890', 'Henry': '12348720987', 'Carl': '657.899.843.21',
2 | 'Lucas': '111 848 506-45', 'Lena': '848-125-602.21', 'Lewis': '753.986.710-21'}
3
4 for x in agenda.keys():
5 |     tmp = "".join([x for x in agenda[x] if x in "0123456789"])
6 |     agenda[x] = "{}.{}.{}-{}".format(tmp[0:3],tmp[3:6],tmp[6:9],tmp[9:11])
7 print(agenda)
```

cloudshell X + ▾

```
$ python3 exemplo05.py
{'Elena': '657.887.628-90', 'Henry': '123.487.209-87', 'Carl': '657.899.843-21',
 'Lucas': '111.848.506-45', 'Lena': '848.125.602-21', 'Lewis': '753.986.710-21'}
$
```

Perguntas

Referências

- Zanoni Dias, MC102, Algoritmos e Programação de Computadores, IC/UNICAMP, 2021. <https://ic.unicamp.br/~mc102/>
 - Aula Introdutória [[slides](#)] [[vídeo](#)]
 - Primeira Aula de Laboratório [[slides](#)] [[vídeo](#)]
 - Python Básico: Tipos, Variáveis, Operadores, Entrada e Saída [[slides](#)] [[vídeo](#)]
 - Comandos Condicionais [[slides](#)] [[vídeo](#)]
 - Comandos de Repetição [[slides](#)] [[vídeo](#)]
 - Listas e Tuplas [[slides](#)] [[vídeo](#)]
 - Strings [[slides](#)] [[vídeo](#)]
 - Dicionários [[slides](#)] [[vídeo](#)]
 - Funções [[slides](#)] [[vídeo](#)]
 - Objetos Multidimensionais [[slides](#)] [[vídeo](#)]
 - Algoritmos de Ordenação [[slides](#)] [[vídeo](#)]
 - Algoritmos de Busca [[slides](#)] [[vídeo](#)]
 - Recursão [[slides](#)] [[vídeo](#)]
 - Algoritmos de Ordenação Recursivos [[slides](#)] [[vídeo](#)]
 - Arquivos [[slides](#)] [[vídeo](#)]
 - Expressões Regulares [[slides](#)] [[vídeo](#)]
 - Execução de Testes no Google Cloud Shell [[slides](#)] [[vídeo](#)]
 - Numpy [[slides](#)] [[vídeo](#)]
 - Pandas [[slides](#)] [[vídeo](#)]
- Panda - Cursos de Computação em Python (IME -USP) <https://panda.ime.usp.br/>
 - Como Pensar Como um Cientista da Computação <https://panda.ime.usp.br/pensepy/static/pensepy/>
 - Aulas de Introdução à Computação em Python <https://panda.ime.usp.br/aulasPython/static/aulasPython/>
- Fabio Kon, Introdução à Ciência da Computação com Python <http://bit.ly/FabioKon/>
- Socratica, Python Programming Tutorials <http://bit.ly/SocraticaPython/>
- Google - online editor for cloud-native applications (Python programming) <https://shell.cloud.google.com/>
- w3schools - Python Tutorial <https://www.w3schools.com/python/>
- Outros, citados nos Slides.