


```

#####

#exercise_one_
re1 = r'[a-zA-Z]+'
#exercise_two
re2 = r'[a-z]*d'
# exercise three. Need word boundaries. See below.
# \b should not be in capture parens.
re3 = r'\b(\w+)\b\s+\b\1\b'
re4 = r'(?:\bcharge\b.+\bcall\b)|(?:\bcall\b.+\bcharge\b)'
res0 = [re1,re2,re3,re4]

#####
###      Some example strings      ###
#####

example1 = 'abracadabra'
example2 = '1billygoat'
example3 = 'billygoat1'
example4 = 'bad apples'
example5 = 'some really really bad apples'
# Does not contain two consecutive repeated words
example6 = 'The doctor tore his sleeve.'
example7 = 'road_runner'
example8 = "Those doctors charge a fortune for a house call."
example9 = "that doctor charged a fortune for that house call."
example10 = "The trumpet call signaled the soldiers to charge."
example11 = 'sad!'
example12 = '*&%#!?'
example13 = 'John believes that that approach will fail.'

examples0 = [example1,example2,example3,example4, example5,example6,
             example7,example8, example9,example10,example11,example12,
             example13]

#####
###      Trying some matches      ###
#####

do_searches (res0, examples0)

```

```

re1 [a-zA-Z]+
=====

```

```

Match
=====

```

14. abracadabra

abracadabra

15. 1billygoat	billygoat
16. billygoat1	billygoat
17. bad apples	bad
18. some really really bad apples	some
19. The doctore tore his sleeve.	The
20. road_runner	road
21. Those doctors charge a fortune for a house call.	Those
22. that doctor charged a fortune for that house call.	that
23. The trumpet call signaled the soldiers to charge.	The
24. sad!	sad
25. *&%#!?	None
26. John believes that that approach will fail.	John

```
re2 [a-z]*d
=====
Match
=====
```

14. abracadabra	abracad
15. 1billygoat	None
16. billygoat1	None
17. bad apples	bad
18. some really really bad apples	bad
19. The doctore tore his sleeve.	d
20. road_runner	road
21. Those doctors charge a fortune for a house call.	d
22. that doctor charged a fortune for that house call.	d
23. The trumpet call signaled the soldiers to charge.	signaled
24. sad!	sad
25. *&%#!?	None
26. John believes that that approach will fail.	None

```
re3 \b(\w+)\b\s+\b\1\b
=====
Match
=====
```

14. abracadabra	None
15. 1billygoat	None
16. billygoat1	None
17. bad apples	None
18. some really really bad apples	really really
19. The doctore tore his sleeve.	None
20. road_runner	None
21. Those doctors charge a fortune for a house call.	None
22. that doctor charged a fortune for that house call.	None
23. The trumpet call signaled the soldiers to charge.	None
24. sad!	None
25. *&%#!?	None
26. John believes that that approach will fail.	that that

```
re4 (?:\bcharge\b.+?\bcall\b)|(?:\bcall\b.+?\bcharge\b) Match
```

```

=====
14. abracadabra None
15. 1billygoat None
16. billygoat1 None
17. bad apples None
18. some really really bad apples None
19. The doctore tore his sleeve. None
20. road_runner None
21. Those doctors charge a fortune for a house call. charge a fortune for a house call
22. that doctor charged a fortune for that house call. None
23. The trumpet call signaled the soldiers to charge. call signaled the soldiers to charge
24. sad! None
25. *&%#!? None
26. John believes that that approach will fail. None

```

```
In [20]: import re
```

```

#####
### Some more regular expressions ###
#####

#exercise four
re1 = r'^1+(01+)*$'
#exercise four alternative
re2 = r'1+(01+)*'
#exercise four wrong
re3 = r'^1*(101+)*$'
res1 = [re1,re2,re3]

#####
### Some example strings ###
#####

example14 = '1001'
example15 = '101101'
example16 = '11010101'
example17 = '101010'
example18 = '10101'

examples1 = [example14, example15, example16, example17, example18]

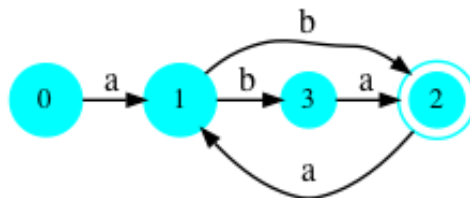
#####
### Trying some matches ###
#####

do_searches (res1, examples1)

```

re1 ^1+(01+)*\$	Match
=====	=====
14. 1001	None
15. 101101	101101
16. 11010101	11010101
17. 101010	None
18. 10101	10101
re2 1+(01+)*	Match
=====	=====
14. 1001	1
15. 101101	101101
16. 11010101	11010101
17. 101010	10101
18. 10101	10101
re3 ^1*(101+)*\$	Match
=====	=====
14. 1001	None
15. 101101	101101
16. 11010101	None
17. 101010	None
18. 10101	None

0.3 Mystery FSA



Mystery FSA

(aba?)+

0.4 Minimum Edit Distance

0.4.1 leda vs deal

l e d a
l 4

```

a 3
e 2 3
d 1 2
# 0 1 2 3 4
# 1 e d a

```

$$D(2,1) = 3, \quad D(3,1) = ?$$

```

# 1 e d a
l 4 3
a 3 4
e 2 3
d 1 2 3
# 0 1 2 3 4
# 1 e d a

```

$$D(3,2) = \min(D(3,1) + 1, \\ D(2,1) + 2, \\ D(2,2) + 1)$$

```

# 1 e d a
l 4 3 4
a 3 4 3
e 2 3 2 3
d 1 2 3 2
# 0 1 2 3 4
# 1 e d a

```

$$D(2,2) = \min(D(2,1) + 1, \\ D(1,1) + 0, \\ D(1,2) + 1)$$

```

# 1 e d a
l 4 3 4 5
a 3 4 3 4
e 2 3 2 3 4
d 1 2 3 2 3
# 0 1 2 3 4
# 1 e d a

```

```

# 1 e d a
l 4 3 4 5 4
a 3 4 3 4 3
e 2 3 2 3 4
d 1 2 3 2 3
# 0 1 2 3 4
# 1 e d a

```

```

l e d a 0
d e 0 a l
- - - - -
2 0 1 0 1

```

Total: 4

0.4.2 Drive vs brief

```
# d r i v e
f 5 6 5 4 5 4
e 4 5 4 3 4 3
i 3 4 3 2 3 4
r 2 3 2 3 4 5
b 1 2 3 4 5 6
# 0 1 2 3 4 5
# d r i v e
```

```
d r i v e 0
b r i 0 e f
- - - - -
2 0 0 1 0 1
```

Total: 4

0.4.3 Drive vs divers

```
# d r i v e
s 6 5 4 5 4 3
r 5 4 3 4 3 2
e 4 3 4 3 2 1
v 3 2 3 2 1 2
i 2 1 2 1 2 3
d 1 0 1 2 3 4
# 0 1 2 3 4 5
# d r i v e
```

```
d r i v e 0 0
d 0 i v e r s
- - - - -
0 1 0 0 0 1 1
```

Total: 3

In []: