

```

"""This program search ip address pattern in a string.
For Ip address the rules are
1. It is of type X.X.X.X
2. We only need to define a pattern for X and all other bytes are of same
pattern
3. We can have this pattern as "X." repeating three time followed by "X"
or X followed by ".X" repeating 3 times
4. For X the rules are it can only be between 1-255.
5. The rules defined are as follow
    a. if two digits are 25 then last digit can only be from 0-5 .
        This can be written as 25[0-5]
    b. else if first digit is 2 then second digit can be between 0-4 and
third digit can be 0-9 .
        This can be written as 2[0-4][0-9]
    c. else if first digit present and is 0 or 1 then second digit can be
0-9 and third digit can be 0-9.
        This can be written as [01]?[0-9][0-9]?
    d. So adding all condition the X can be written as 25[0-5]|2[0-4][0-
9]|[01]?[0-9][0-9]?
6. Once X is defined we can write ip address pattern as X followed by .X
three times i.e X(.X){3}
7. We also need to declare that ip address must be followed by space \s
or end of string $ i.e \s|$
"""

```

```

import re
pat = r"((25[0-5]|2[0-4][0-9]|[01]?[0-9][0-9]?)(\. (25[0-5]|2[0-4][0-
9]|[01]?[0-9][0-9]?)){3})(\s|$)"

data = "adfd sdf 12 25.25 5.255.255 55.5 192.169.1.02 1.1.1.1 adf 2.0 "

match = re.search(pat,data)
if match:
    print match.group(1)
else:
    print "No match found"

```