In [5]:
```python
import nltk
from nltk.tokenize import word_tokenize
from nltk.tag import pos_tag
from nltk.chunk import RegexpParser
```

In [6]:
```python
def bio_tag(sentence):
    words = word_tokenize(sentence)
    tagged_words = pos_tag(words)
    bio_tagged=[]
    grammar=r"""
            NP:{<DT|PP\$>?<JJ>*<NN>}
            {<NNP>+}
            {<NN>+}
            """
    cp=RegexpParser(grammar)
    named_entities=cp.parse(tagged_words)

    for subtree in named_entities:
        if type(subtree) == nltk.tree.Tree:
            if subtree.label() != "S":
                bio_tagged.extend(["B"] + ["I"]*(len(subtree)-1))
            else:
                bio_tagged.extend(["O"]*(len(subtree)))
        else:
            bio_tagged.append("O")
    return list(zip(words,bio_tagged))
```

In [11]:
```python
text=input("Enter the text")
print(bio_tag(text))
```

```
Enter the textMy favorite movie is Jaws and series is Breaking Bad.
[('My', 'O'), ('favorite', 'B'), ('movie', 'I'), ('is', 'O'), ('Jaws', 'B'), ('and', 'O'), ('serie
s', 'B'), ('is', 'O'), ('Breaking', 'O'), ('Bad', 'B'), ('.', 'O')]
```