Lab 1: Analysis of Tokenization and N-grams in Natural Language Processing\

```
import nltk
nltk.download('punkt')

from nltk.tokenize import word_tokenize
from nltk.util import ngrams
from nltk.probability import FreqDist

# Sample document
document = "Natural language processing (NLP) is a subfield of artificial intelligence (AI)

# Tokenization
tokens = word_tokenize(document.lower())

# Total number of tokens
total_tokens = len(tokens)

# Number of unique tokens
unique_tokens = set(tokens)
num_unique_tokens = len(unique_tokens)

# Token frequency distribution
token_freq = FreqDist(tokens)

# Generate N-grams
n = 2  # Change to desired n-gram size
n_grams = list(ngrams(tokens, n))

# N-gram frequency distribution
n_gram_freq = FreqDist(n_grams)

# Report
print("Total number of tokens:", total_tokens)
print("Number of unique tokens:", num_unique_tokens)

print("\nToken frequency distribution:")
for token, freq in token_freq.items():
    print(f"{token}: {freq}")

print("\nTop 5 Tokens:")
for token, freq in token_freq.most_common(5):
    print(f"{token}: {freq}")

print("\nN-gram frequency distribution:")
for n_gram, freq in n_gram_freq.items():
    print(f"{' '.join(n_gram)}: {freq}")

print("\nTop 5 Bi-grams:")
for n_gram, freq in n_gram_freq.most_common(5):
    print(f"{' '.join(n_gram)}: {freq}")
```