```
import random
import string
import time
import matplotlib.pyplot as plt

def computeArray(p, m, lpArray):
    len = 0
    lpArray[0]
    i = 1
    while i < m:
        if p[i]== p[len]:
            len += 1
            lpArray[i] = len
            i += 1
        else:
            if len != 0:
                len = lpArray[len-1]
            else:
                lpArray[i] = 0
                i += 1

def kmpSearch(text,pattern):
    m = len(pattern)
    n = len(text)
    lpArray = [0]*m
    j = 0
    computeArray(pattern, m, lpArray)
    i = 0
    while i < n:
        if pattern[j] == text[i]:
            i += 1
            j += 1

        if j == m:
            print ("Pattern is present at index ",str(i-j))
            j = lpArray[j-1]
        elif i < n and pattern[j] != text[i]:
            if j != 0:
                j = lpArray[j-1]
            else:
                i += 1

text = "ABABDABACDABAAB"
pattern = "ABA"
kmpSearch(text,pattern)
```

```
Pattern is present at index  0
Pattern is present at index  5
Pattern is present at index  10
```

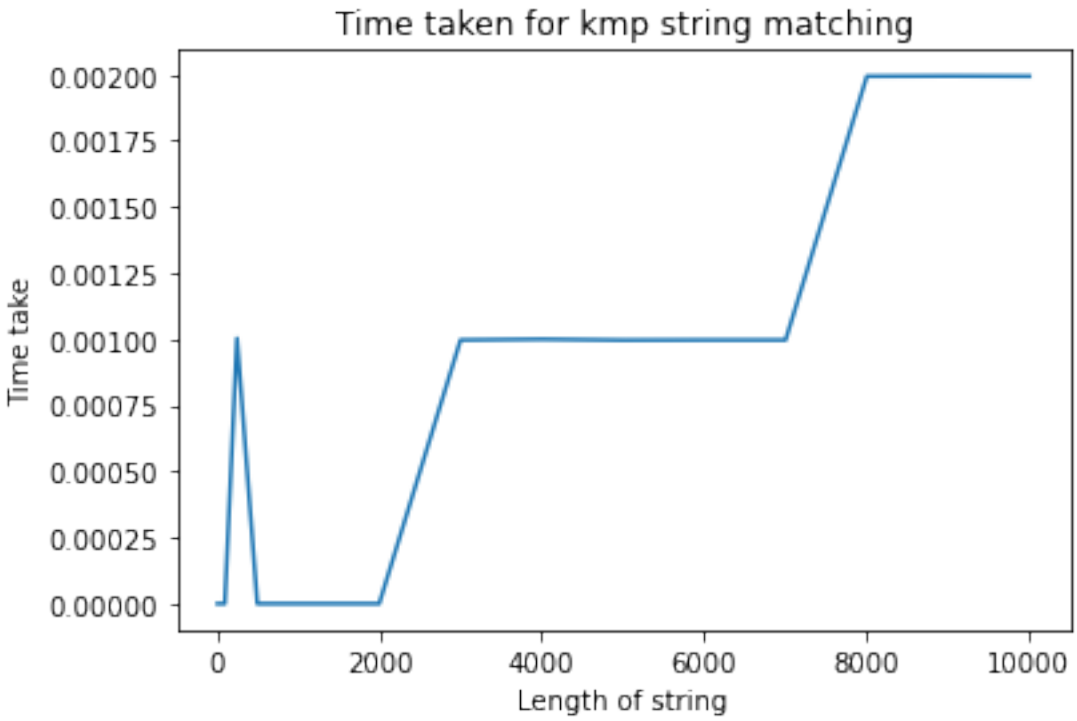
Time complexity

```
runTime=[]
size=[10,20,50,100,250,500,1000,2000,3000,4000,5000,6000,7000,8000,900
0,10000]
for i in size:
    N=i

text=''.join(random.choices(string.ascii_uppercase+string.digits+strin
g.ascii_lowercase, k = N))

pattern=''.join(random.choices(string.ascii_uppercase+string.digits+st
ring.ascii_lowercase, k = 3))
    start=time.time()
    kmpSearch(text,pattern)
    stop=time.time()
    runTime.append(stop-start)

plt.plot(size,runTime)
plt.xlabel("Length of string")
plt.ylabel("Time take")
plt.title("Time taken for kmp string matching")
plt.show()
```



Time complexity of kmp string matching is of the o