**Kmp pattern matching alorithm**

#include<bits/stdc++.h>

using namespace std;

void find\_lps(string pattern, int lps[])

{

int i=1, j=0;

lps[0]=0;

while(i<pattern.size())

{

if(pattern[i]==pattern[j])

{

j++;

lps[i]=j;

i++;

}

else if(j!=0)

{

j=lps[j-1];

}

else

{

lps[i]=0;

i++;

}

}

return ;

}

int kmpSearch(string text, string pattern, int lps[])

{

int i=0, j=0, numberOfOccurance=0;

while(i<text.size())

{

if(text[i]==pattern[j])

{

i++;

j++;

if(j>=pattern.size())

{

numberOfOccurance++;

j=lps[j-1];

}

}

else if(j!=0)

{

j=lps[j-1];

}

else i++;

}

return numberOfOccurance;

}

int main()

{

ios\_base::sync\_with\_stdio(false);

cin.tie(NULL);

string text, pattern;

cin>>text>>pattern;

int lps[10000];

find\_lps(pattern, lps);

int numberOfOccurance=kmpSearch(text, pattern, lps);

cout<<numberOfOccurance<<endl;

return 0;

}