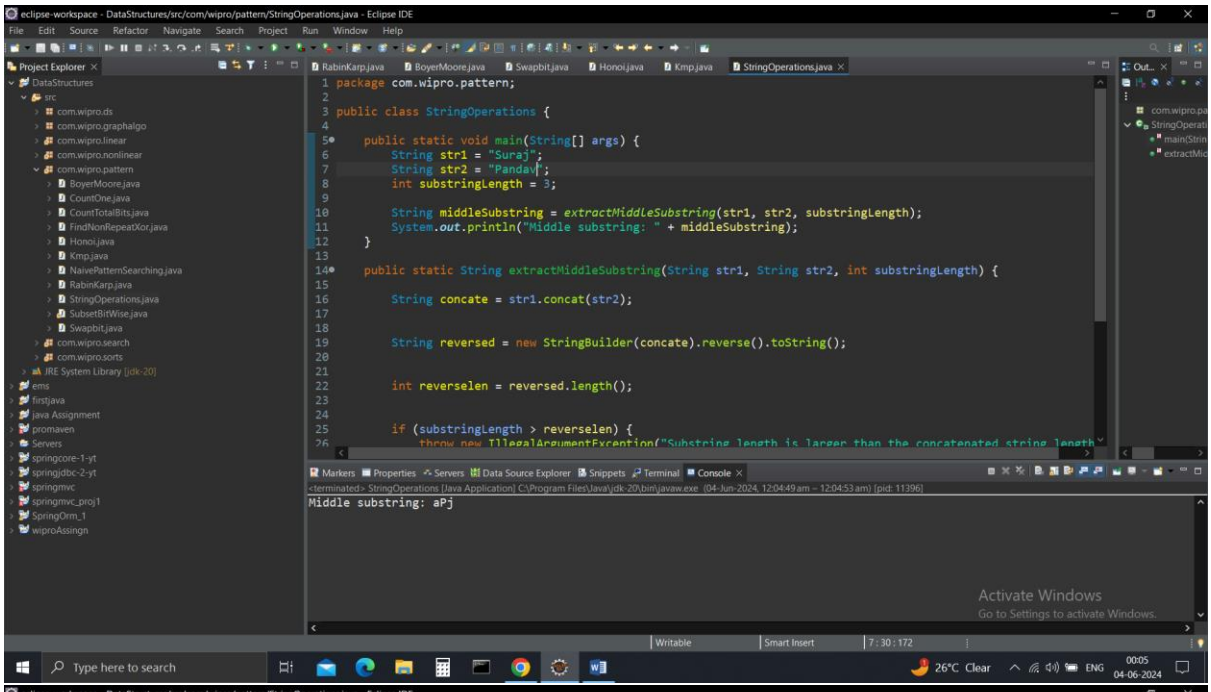
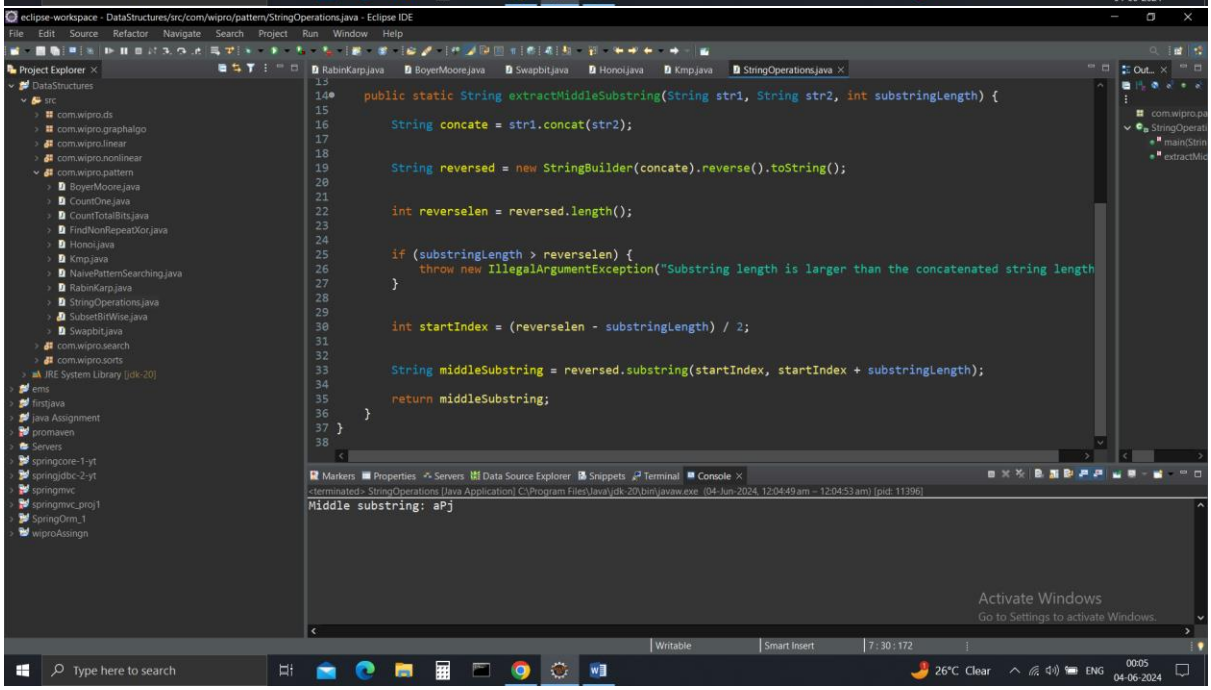
Task 1: String Operations

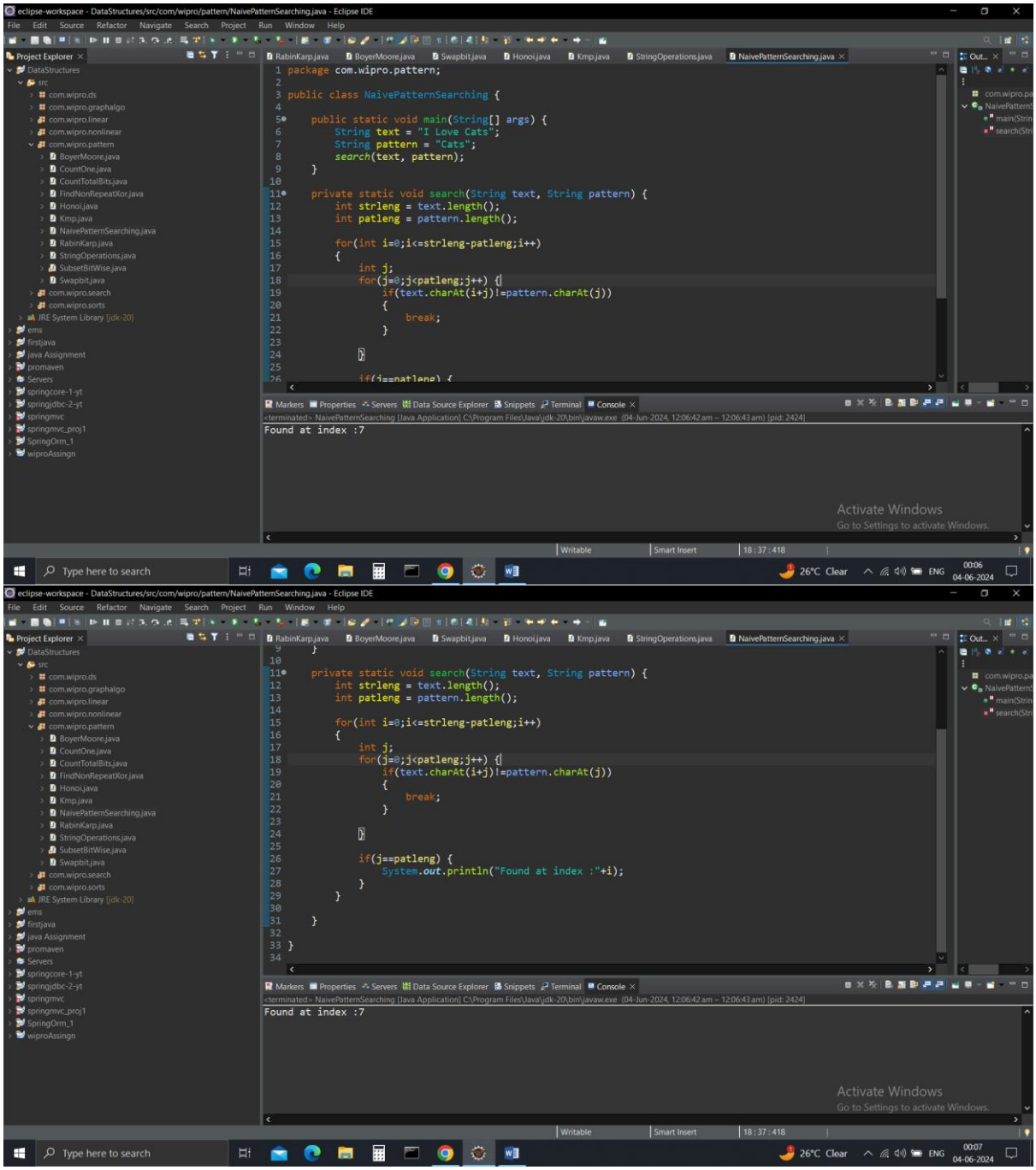
Write a method that takes two strings, concatenates them, reverses the result, and then extracts the middle substring of the given length. Ensure your method handles edge cases, such as an empty string or a substring length larger than the concatenated string.





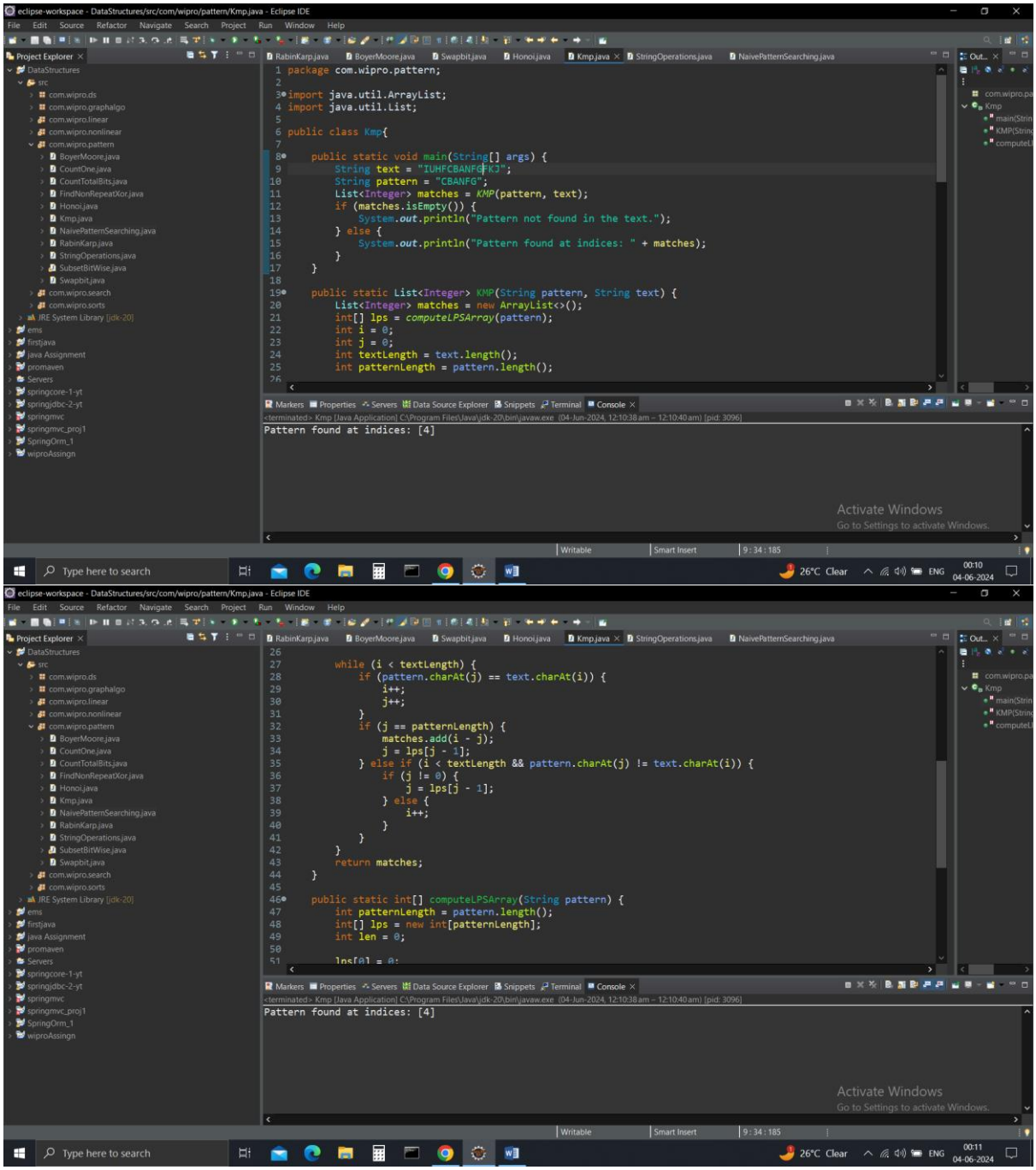
Task 2: Naive Pattern Search

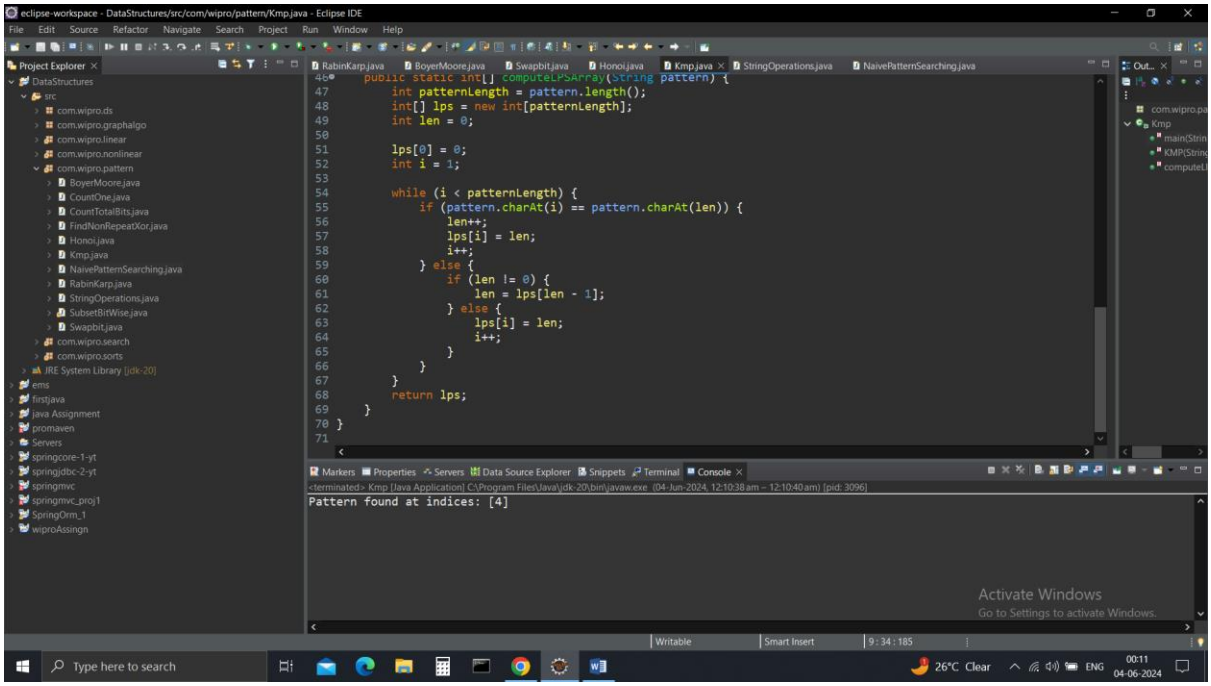
Implement the naive pattern searching algorithm to find all occurrences of a pattern within a given text string. Count the number of comparisons made during the search to evaluate the efficiency of the algorithm



Task 3: Implementing the KMP Algorithm.

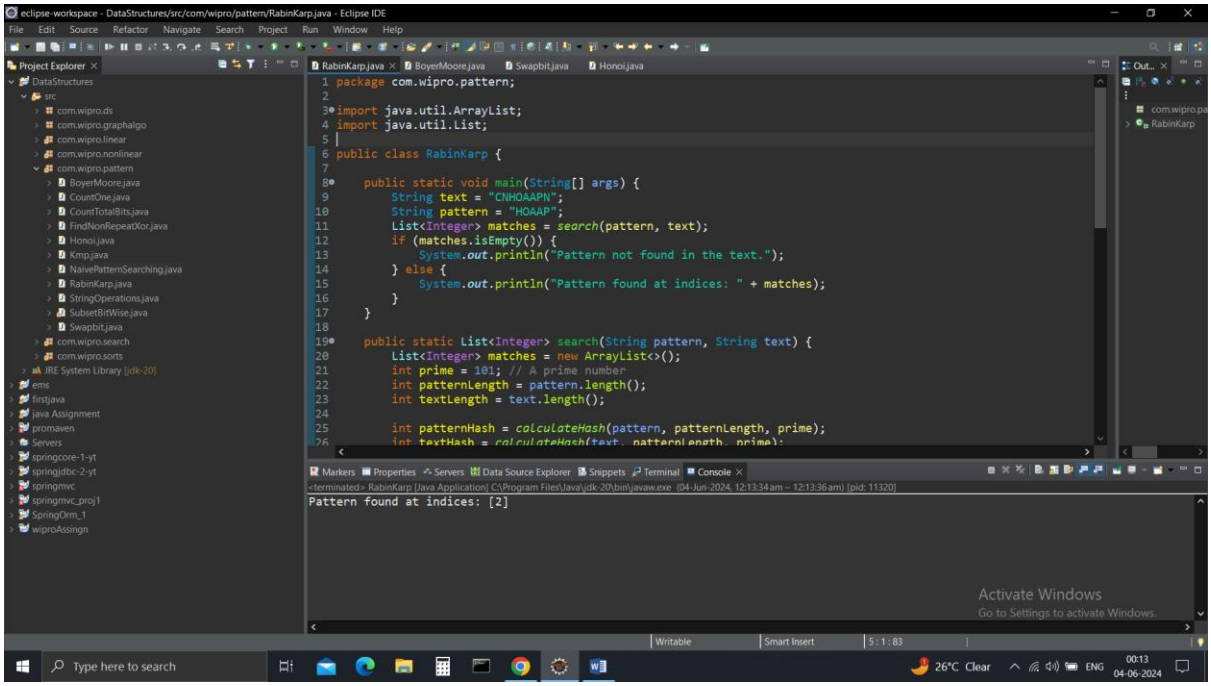
Code the Knuth-Morris-Pratt (KMP) algorithm in C# for pattern searching which pre-processes the pattern to reduce the number of comparisons. Explain how this pre-processing improves the search time compared to the naive approach

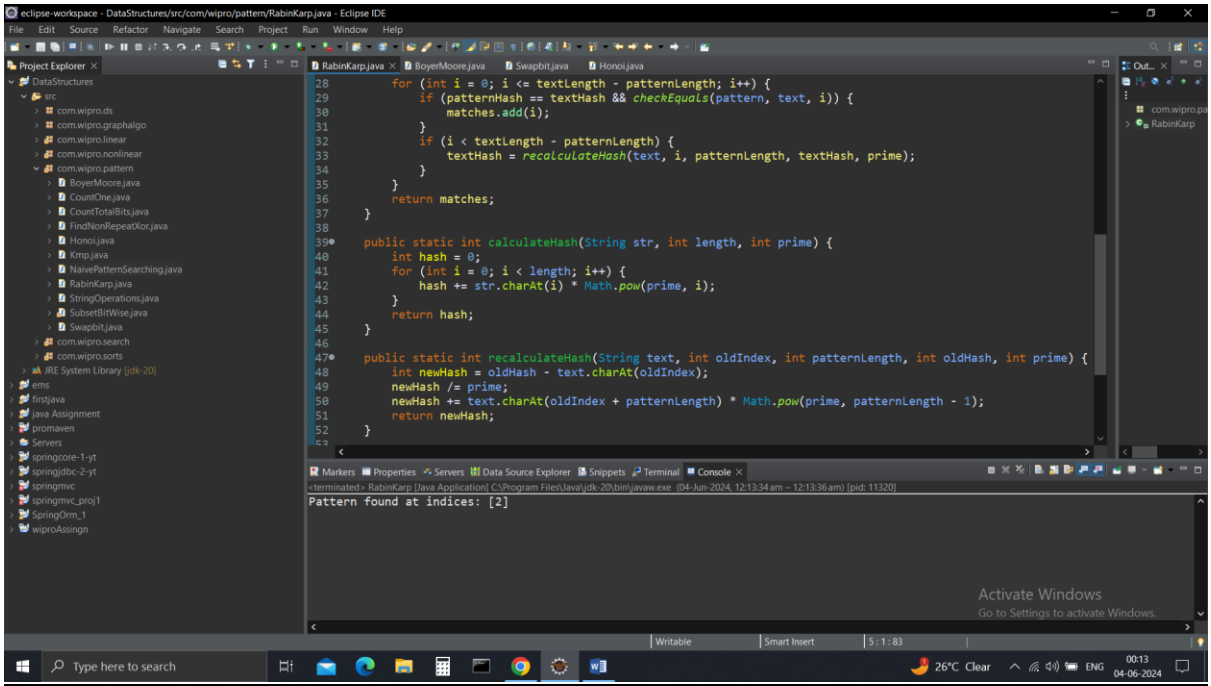


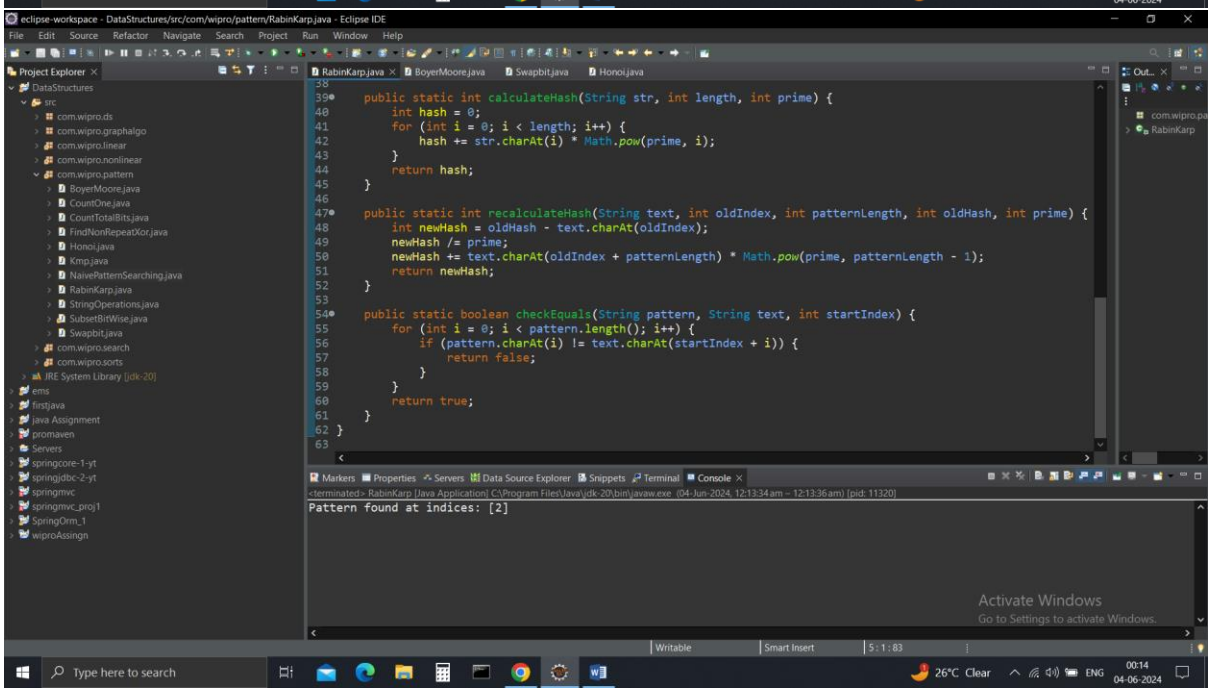


Task 4:Rabin-Karp Substring Search

Implement the Rabin-Karp algorithm for substring search using a rolling hash. Discuss the impact of hash collisions on the algorithm's performance and how to handle them.

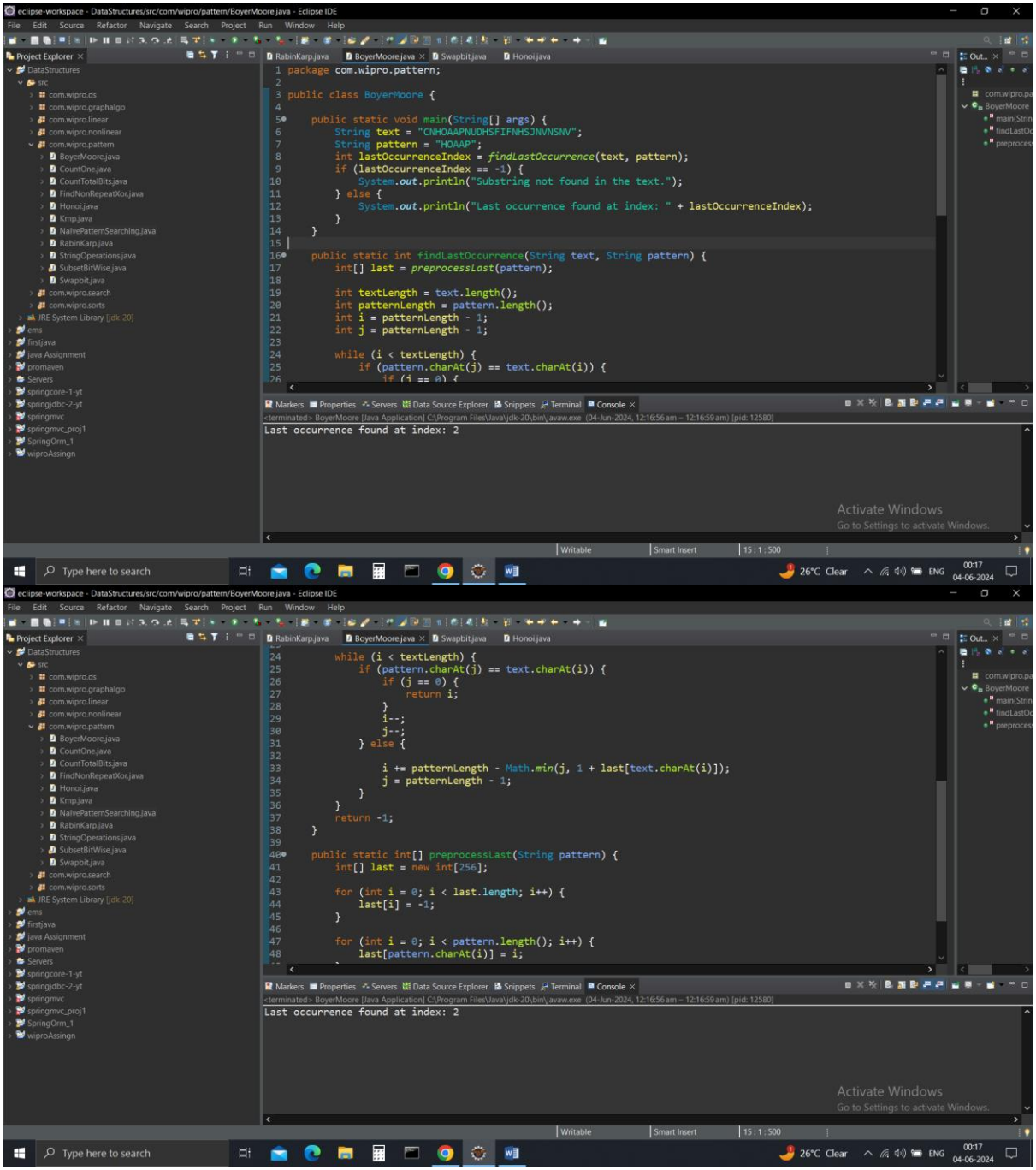


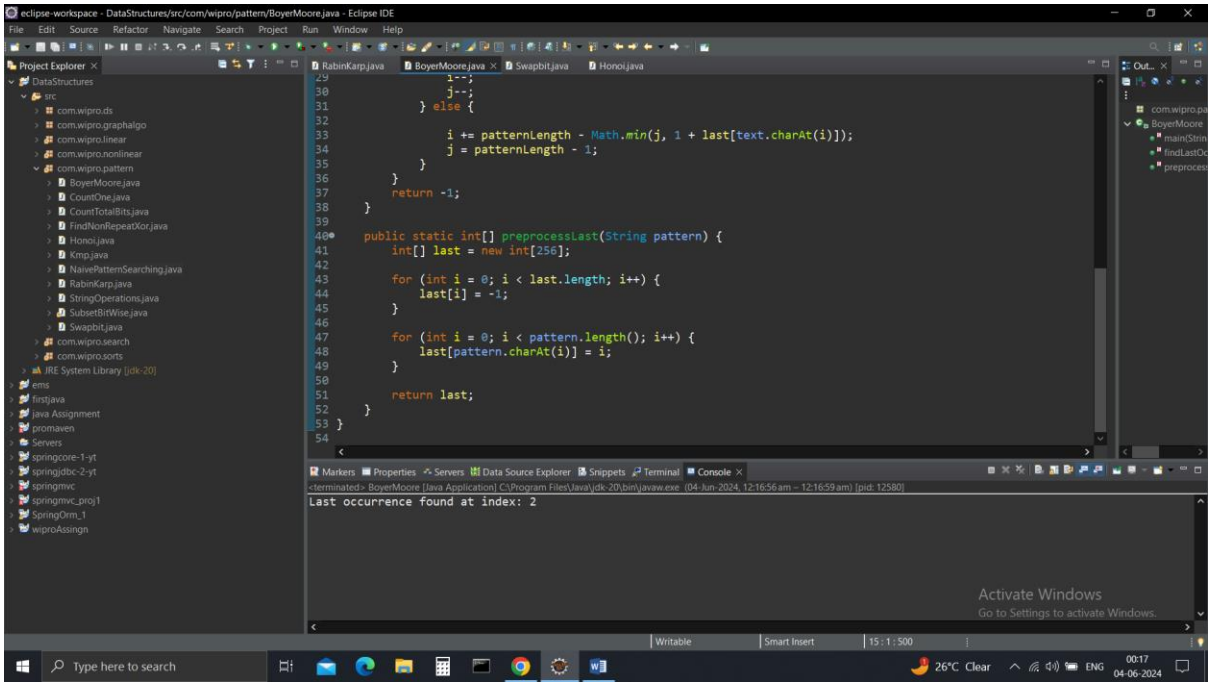

eclipse-workspace - DataStructures/src/com/wipro/pattern/RabinKarp.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Project Explorer
DataStructures
src
com.wipro.ds
com.wipro.graphalgo
com.wipro.linear
com.wipro.nonlinear
com.wipro.pattern
BoyerMoore.java
CountOne.java
CountTotalBits.java
FindNonRepeatXor.java
Honoi.java
Kmp.java
NaivePatternSearching.java
RabinKarp.java
StringOperations.java
SubsetBitWise.java
Swapbit.java
com.wipro.search
com.wipro.sorts
JRE System Library [jdk-20]
ems
firstjava
java Assignment
promaven
Servers
springcore-1-yt
springjdbc-2-yt
springmvc
springmvc_proj1
SpringOrm_1
wiproAssingn
RabinKarp.java
BoyerMoore.java
Swapbit.java
Honoi.java
for (int i = 0; i <= textLength - patternLength; i++) {
if (patternHash == textHash && checkEquals(pattern, text, i)) {
matches.add(i);
}
if (i < textLength - patternLength) {
textHash = recalculateHash(text, i, patternLength, textHash, prime);
}
}
return matches;
}
public static int calculateHash(String str, int length, int prime) {
int hash = 0;
for (int i = 0; i < length; i++) {
hash += str.charAt(i) * Math.pow(prime, i);
}
return hash;
}
public static int recalculateHash(String text, int oldIndex, int patternLength, int oldHash, int prime) {
int newHash = oldHash - text.charAt(oldIndex);
newHash /= prime;
newHash += text.charAt(oldIndex + patternLength) * Math.pow(prime, patternLength - 1);
return newHash;
}
Markers Properties Servers Data Source Explorer Snippets Terminal Console
<terminated> RabinKarp [Java Application] C:\Program Files\Java\jdk-20\bin\javaw.exe (04-Jun-2024, 12:13:34 am – 12:13:36 am) [pid: 11320]
Pattern found at indices: [2]
Activate Windows
Go to Settings to activate Windows.
Writable
Smart Insert
5 : 1 : 83
Type here to search
26°C Clear
ENG
00:13
04-06-2024


eclipse-workspace - DataStructures/src/com/wipro/pattern/RabinKarp.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Project Explorer
DataStructures
src
com.wipro.ds
com.wipro.graphalgo
com.wipro.linear
com.wipro.nonlinear
com.wipro.pattern
BoyerMoore.java
CountOne.java
CountTotalBits.java
FindNonRepeatXor.java
Honoi.java
Kmp.java
NaivePatternSearching.java
RabinKarp.java
StringOperations.java
SubsetBitWise.java
Swapbit.java
com.wipro.search
com.wipro.sorts
JRE System Library [jdk-20]
ems
firstjava
java Assignment
promaven
Servers
springcore-1-yt
springjdbc-2-yt
springmvc
springmvc_proj1
SpringOrm_1
wiproAssingn
RabinKarp.java
BoyerMoore.java
Swapbit.java
Honoi.java
public static int calculateHash(String str, int length, int prime) {
int hash = 0;
for (int i = 0; i < length; i++) {
hash += str.charAt(i) * Math.pow(prime, i);
}
return hash;
}
public static int recalculateHash(String text, int oldIndex, int patternLength, int oldHash, int prime) {
int newHash = oldHash - text.charAt(oldIndex);
newHash /= prime;
newHash += text.charAt(oldIndex + patternLength) * Math.pow(prime, patternLength - 1);
return newHash;
}
public static boolean checkEquals(String pattern, String text, int startIndex) {
for (int i = 0; i < pattern.length(); i++) {
if (pattern.charAt(i) != text.charAt(startIndex + i)) {
return false;
}
}
return true;
}
}
Markers Properties Servers Data Source Explorer Snippets Terminal Console
<terminated> RabinKarp [Java Application] C:\Program Files\Java\jdk-20\bin\javaw.exe (04-Jun-2024, 12:13:34 am – 12:13:36 am) [pid: 11320]
Pattern found at indices: [2]
Activate Windows
Go to Settings to activate Windows.
Writable
Smart Insert
5 : 1 : 83
Type here to search
26°C Clear
ENG
00:14
04-06-2024

Task 5: Boyer-Moore Algorithm Application

Use the Boyer-Moore algorithm to write a function that finds the last occurrence of a substring in a given string and returns its index. Explain why this algorithm can outperform others in certain scenarios.




eclipse-workspace - DataStructures/src/com/wipro/pattern/BoyerMoore.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Project Explorer
DataStructures
src
com.wipro.ds
com.wipro.graphalgo
com.wipro.linear
com.wipro.nonlinear
com.wipro.pattern
BoyerMoore.java
CountOne.java
CountTotalBits.java
FindNonRepeatXor.java
Honoi.java
Kmp.java
NaivePatternSearching.java
RabinKarp.java
StringOperations.java
SubsetBitWise.java
Swapbit.java
com.wipro.search
com.wipro.sorts
JRE System Library [jdk-20]
ems
firstjava
java Assignment
promaven
Servers
springcore-1-yt
springjdbc-2-yt
springmvc
springmvc_proj1
SpringOrm_1
wiproAssingn
RabinKarp.java
BoyerMoore.java
Swapbit.java
Honoi.java
i--;
j--;
} else {
i += patternLength - Math.min(j, 1 + last[text.charAt(i)]);
j = patternLength - 1;
}
}
return -1;
}
public static int[] preprocessLast(String pattern) {
int[] last = new int[256];
for (int i = 0; i < last.length; i++) {
last[i] = -1;
}
for (int i = 0; i < pattern.length(); i++) {
last[pattern.charAt(i)] = i;
}
return last;
}
}
Markers Properties Servers Data Source Explorer Snippets Terminal Console
<terminated> BoyerMoore [Java Application] C:\Program Files\Java\jdk-20\bin\javaw.exe (04-Jun-2024, 12:16:56 am – 12:16:59 am) [pid: 12580]
Last occurrence found at index: 2
Activate Windows
Go to Settings to activate Windows.
Writable
Smart Insert
15 : 1 : 500
Type here to search
26°C Clear
ENG
00:17
04-06-2024