

**TCP 2101 ALGORITHM DESIGN AND ANALYSIS**

**Trimester 2 Session 2016/2017**

**REPORT**

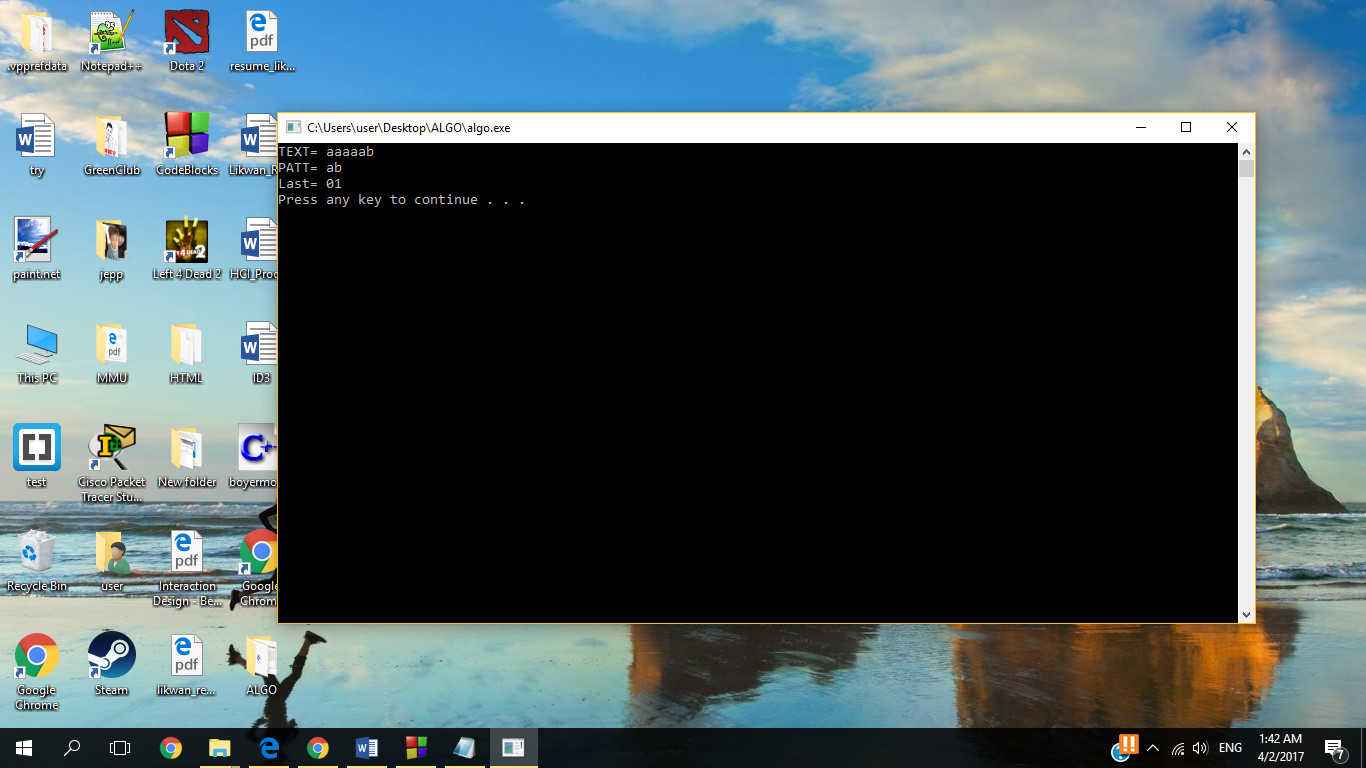
**Assignment : Boyre-Moore pattern**

**GROUP MEMBER :**

**JEFFRY HANDANI 1131121266**

**TAN LI KWAN 1122702571**

**WORST CASE SCENARIO**

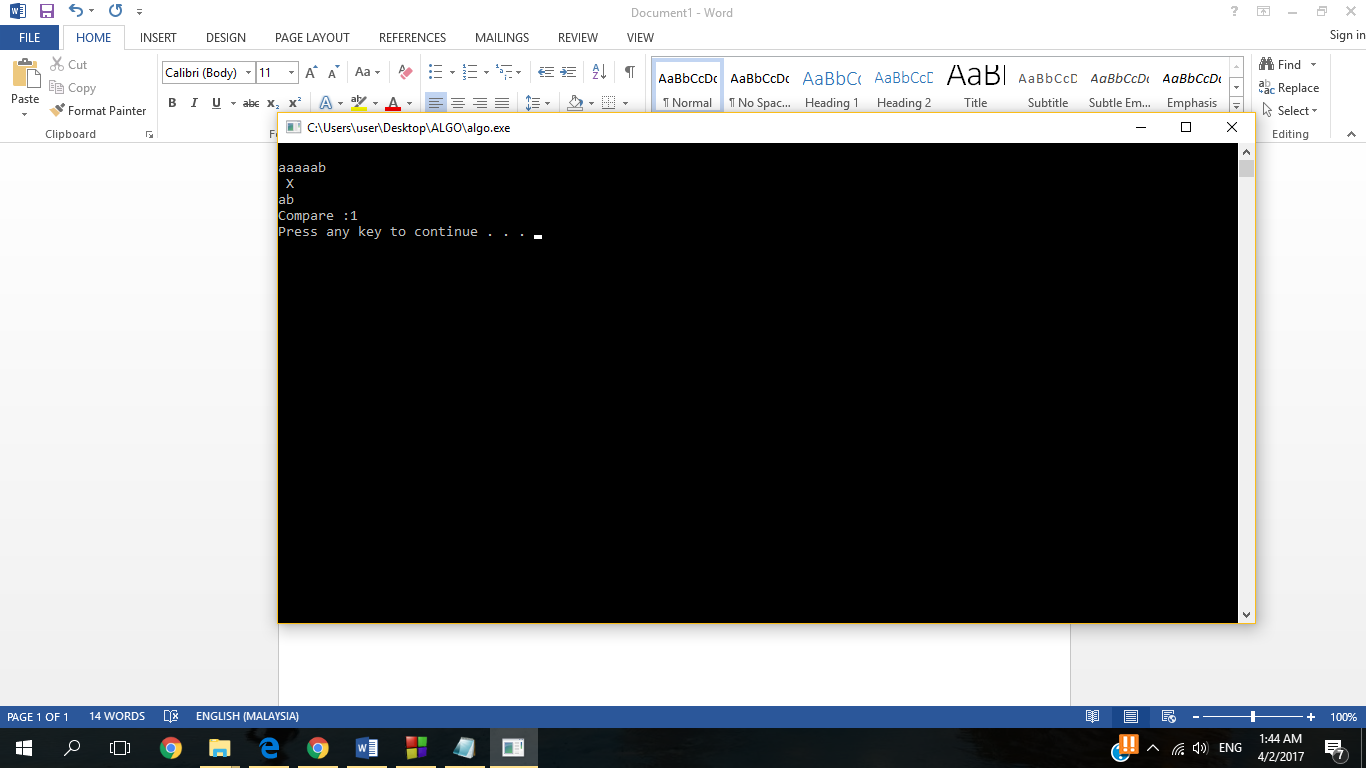


**Worst Case**

Text : aaaaab

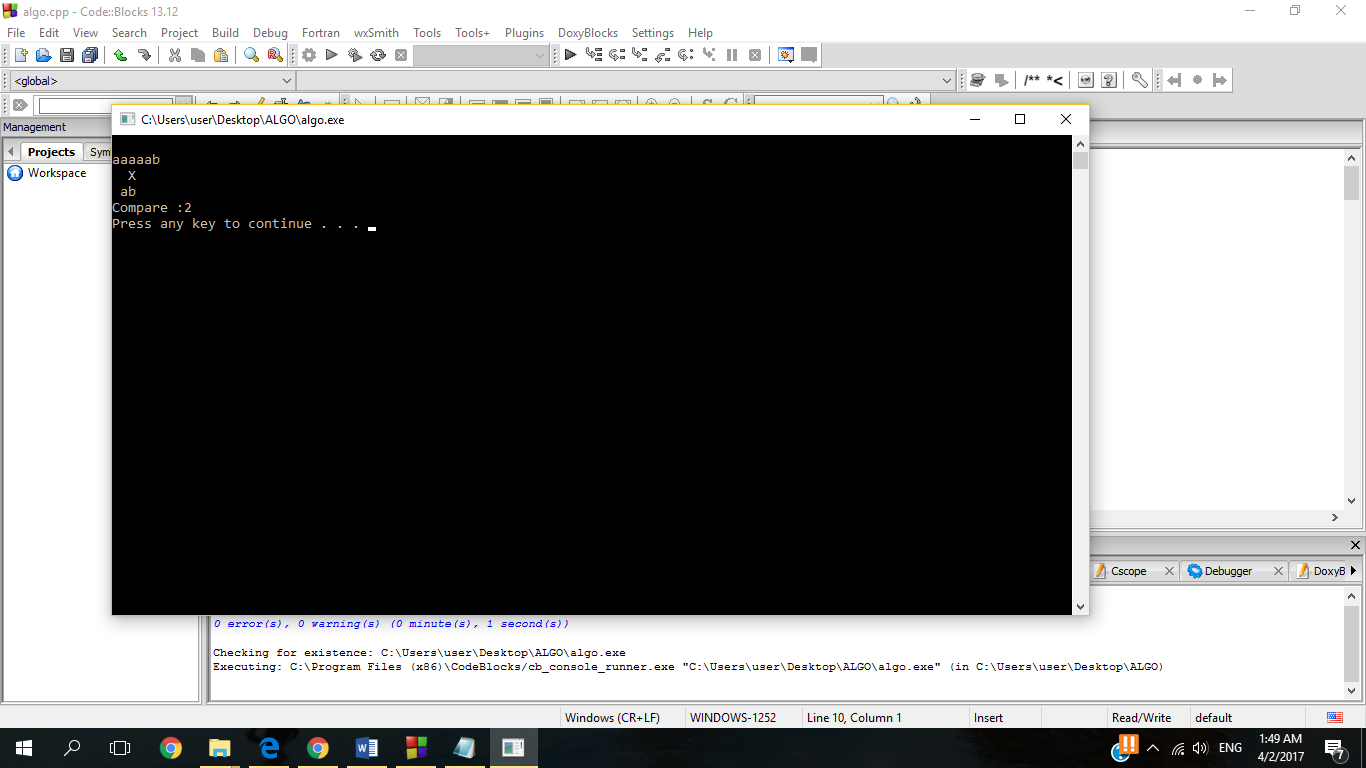
Pattern :ab

Last :01



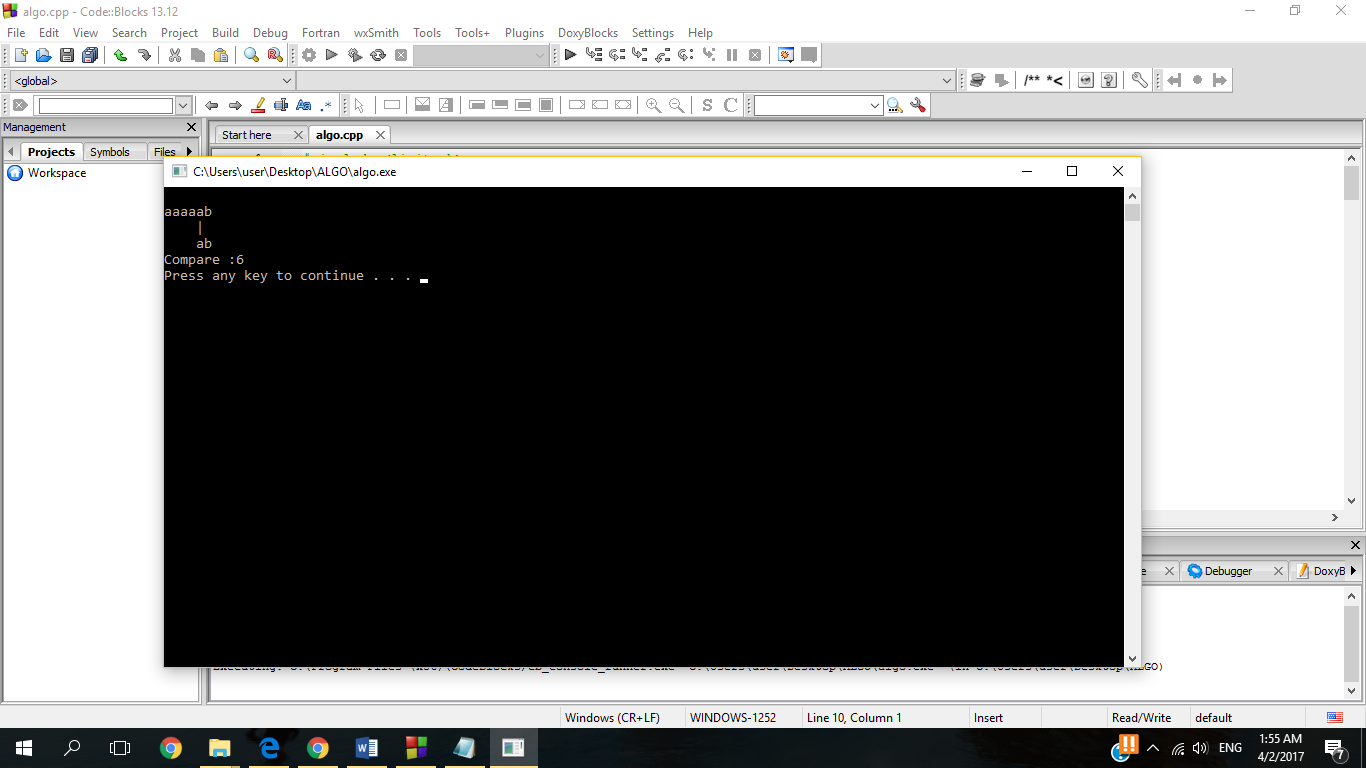
**Worst Case Walkthrough 1 :**

pattern ‘ab’ compare with text ‘aa’ and it does not match so ‘X’ sign appeared at the compared location.



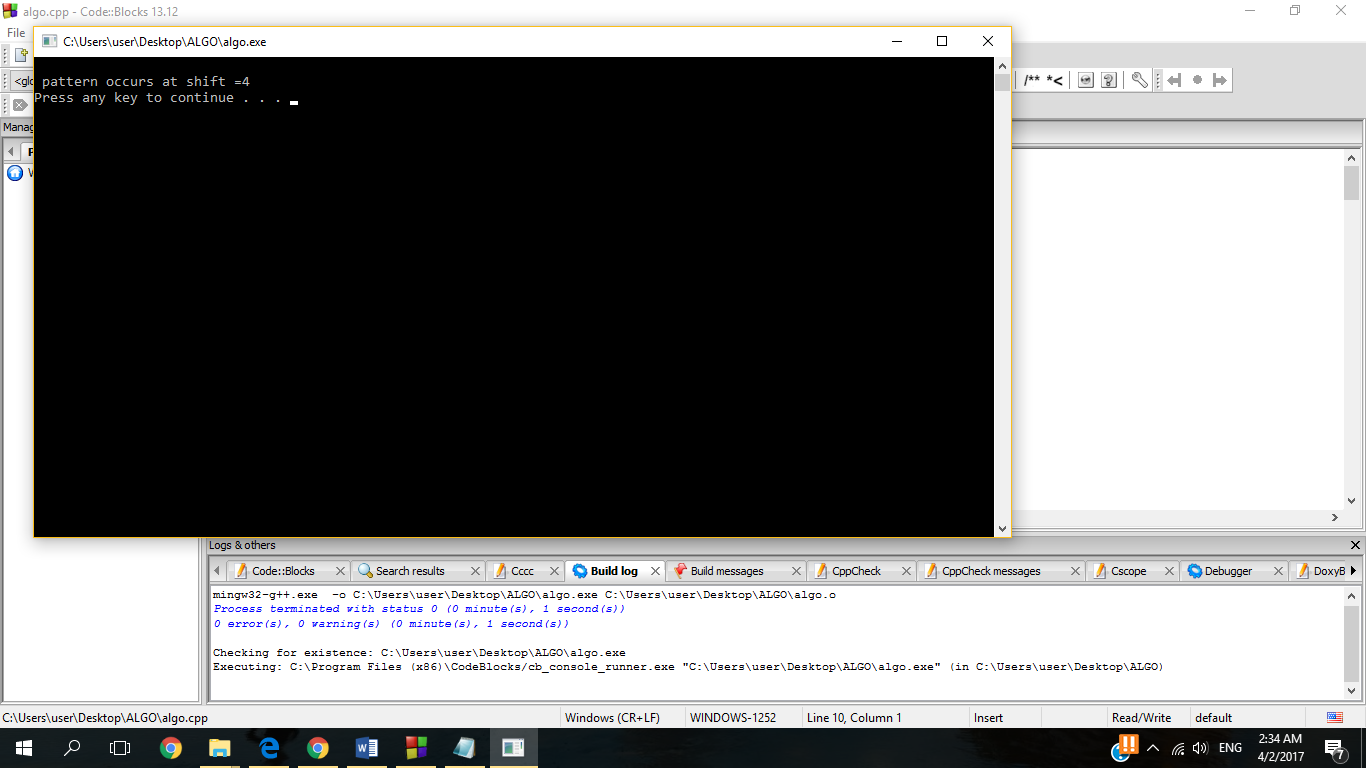
**Worst Case Walkthrough 2:**

As previous case failed to match the pattern and text , ‘ab’ shift to the nex nearest ‘a’ and it resulted failure so the ‘X’ appear in the context.



**Worst Case Walkthrough 3 :**

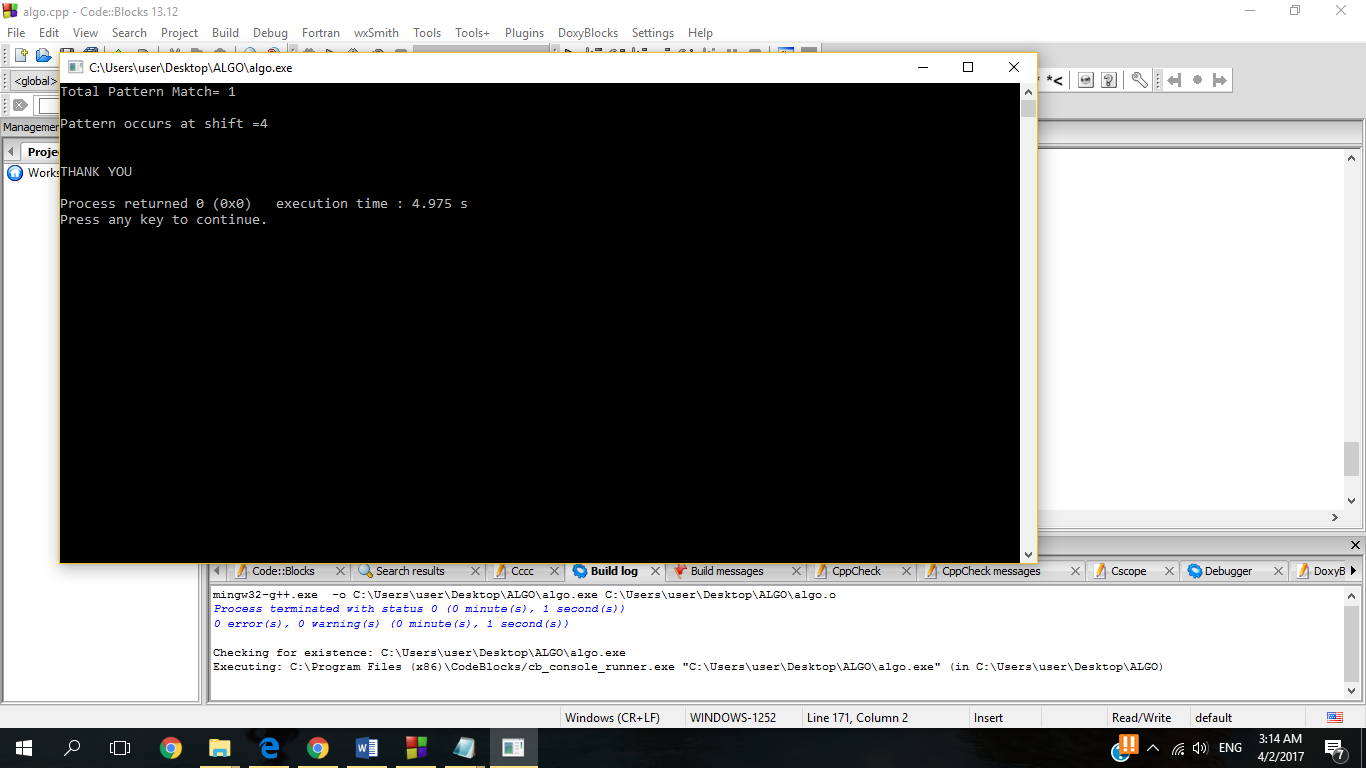
After several failure to match the pattern and text , at number 6 time it finally matches the ‘ab’ and ‘ab’ hence the ‘|’ vertical line indicate success of matching.



**Worst Case Walkthrough 4 :**

Pattern occur at shift = 4

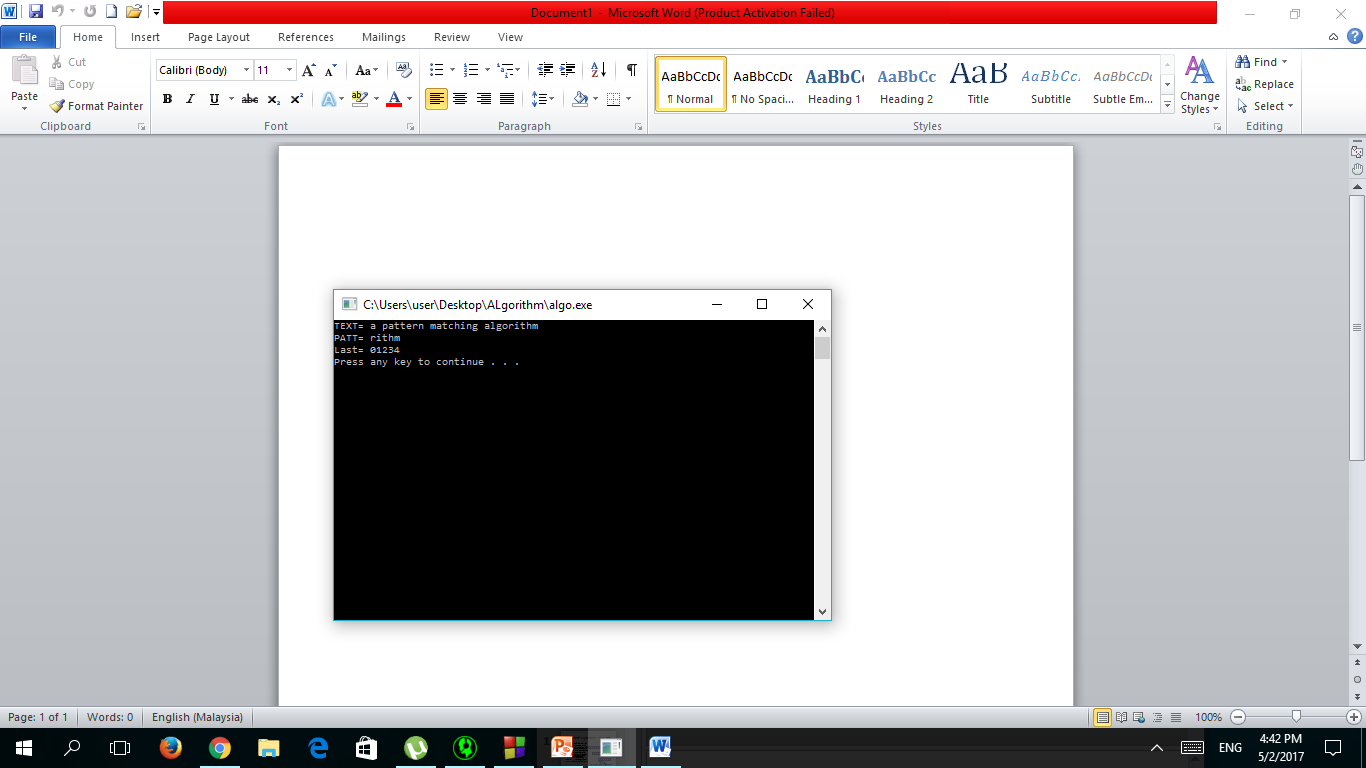
It requires several attempt to match the pattern and text hence it is consider as worst case scenario.



**Worst Case Walkthrough 5 (Final Result):**

**Total pattern match = 1 , Pattern Occurs at shift = 4. There is one successfully match pattern and text and the pattern match the text when array at 4 .**

**Average Case**

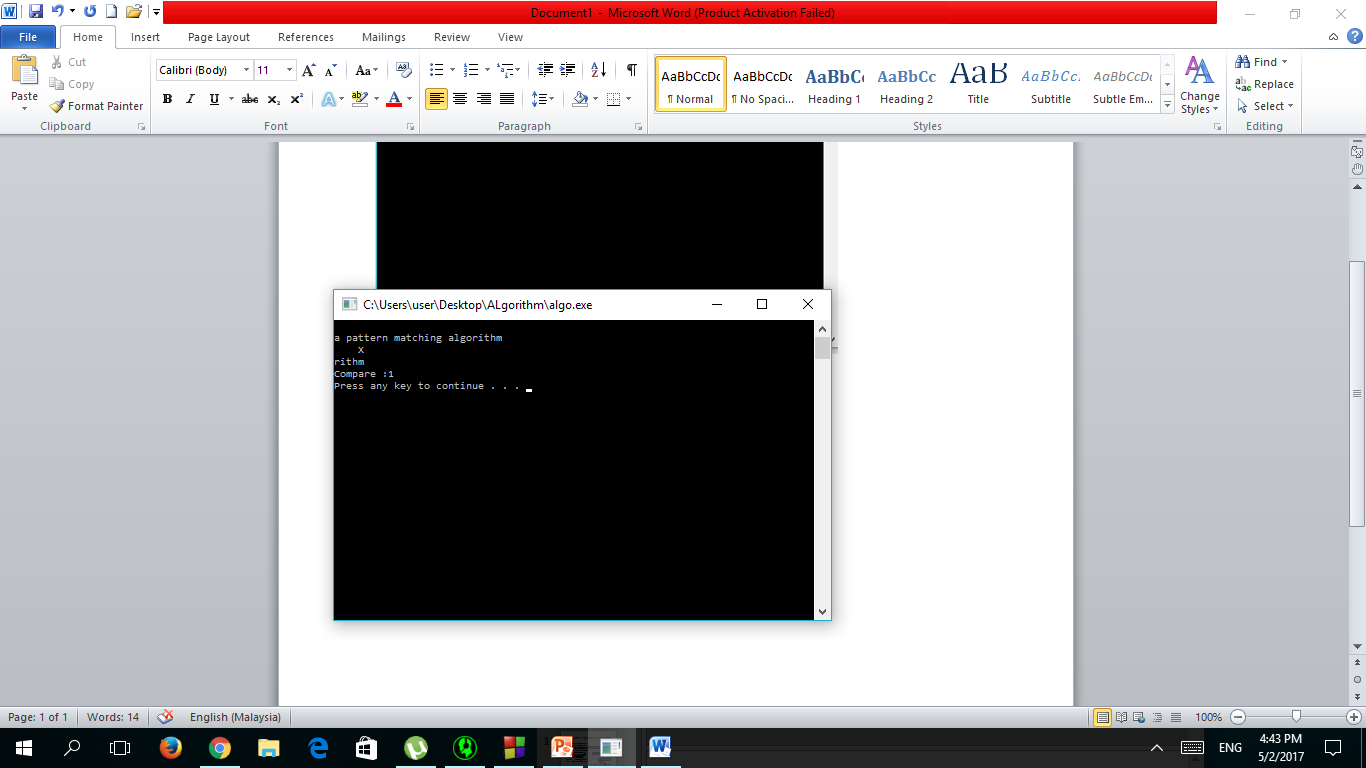


Average Case :

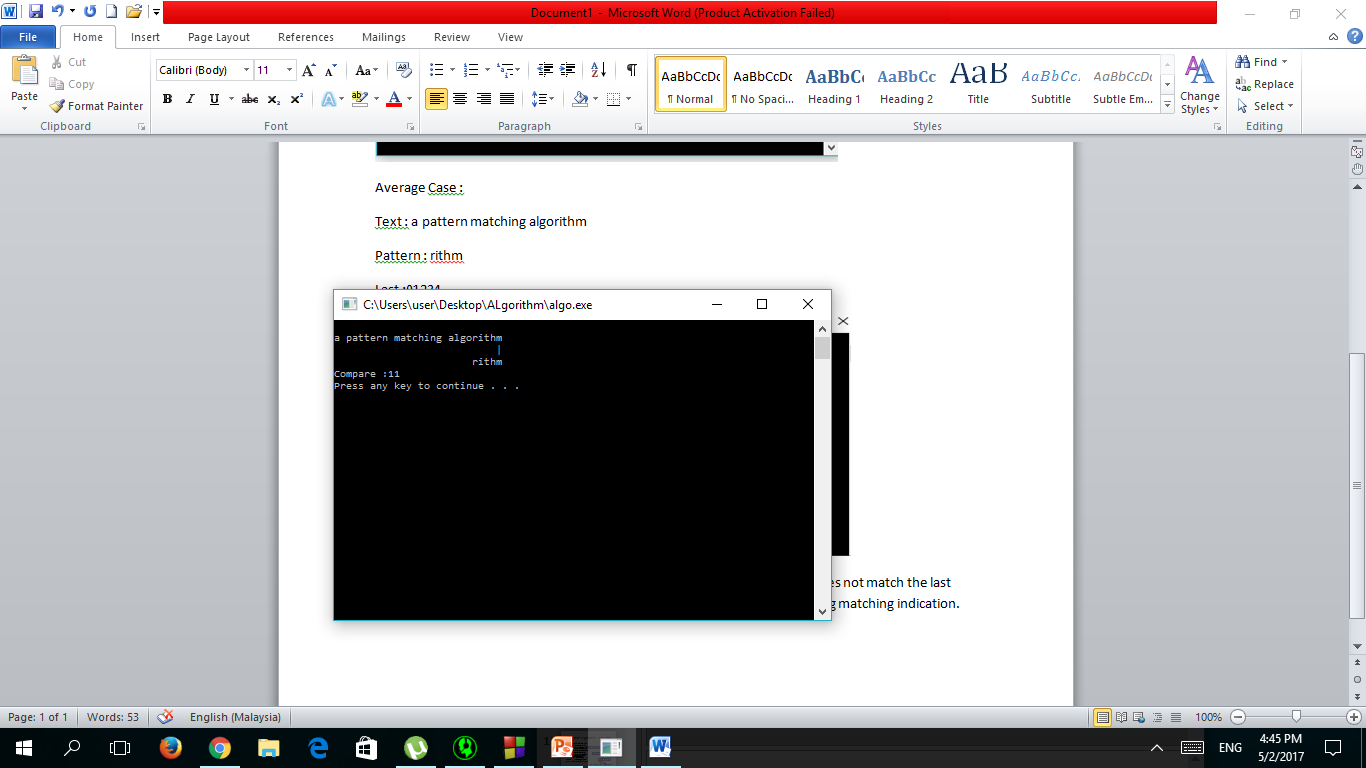
Text : a pattern matching algorithm

Pattern : rithm

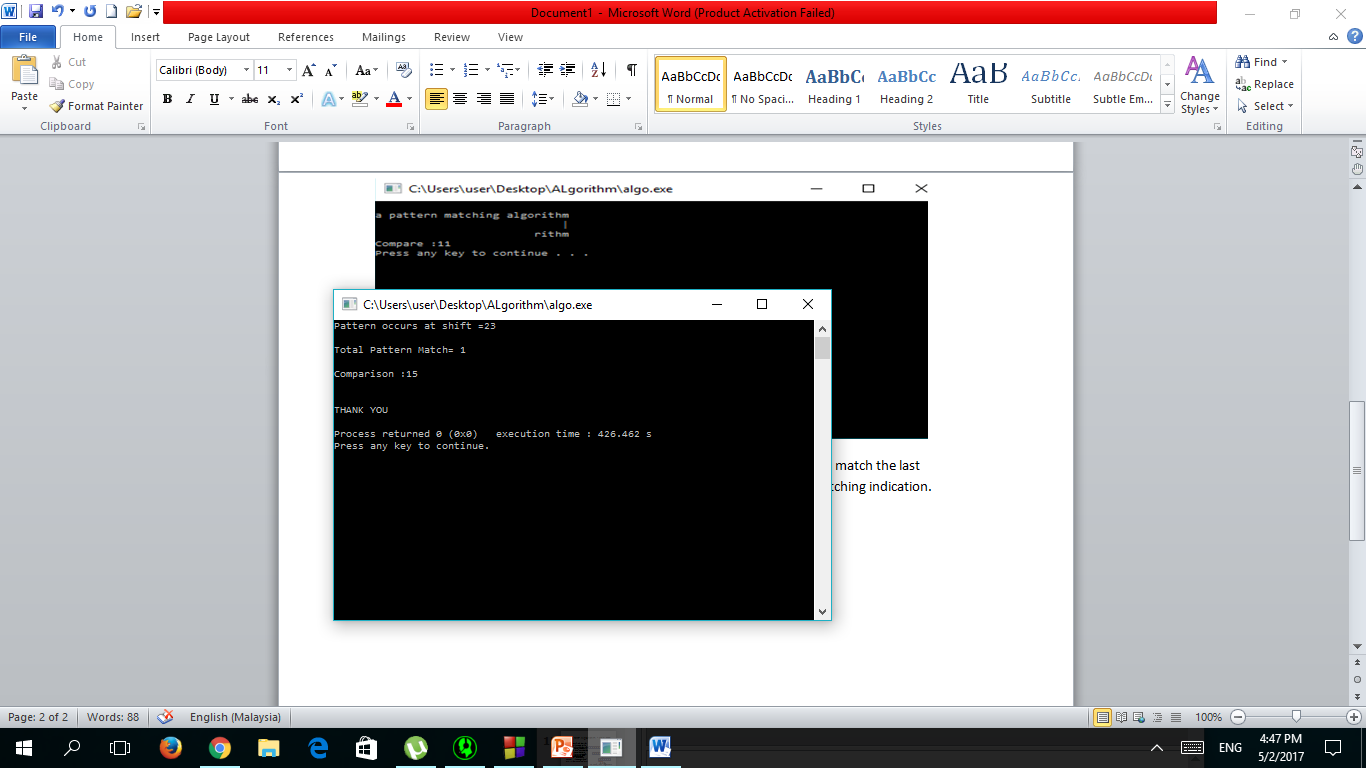
Last :01234



At the first matching , it fails as the last character of pattern ‘ rithm’ =’m’ does not match the last character of matching text of ‘a pat’ =’t’ hence it shows a ‘x’ symbol as wrong matching indication.



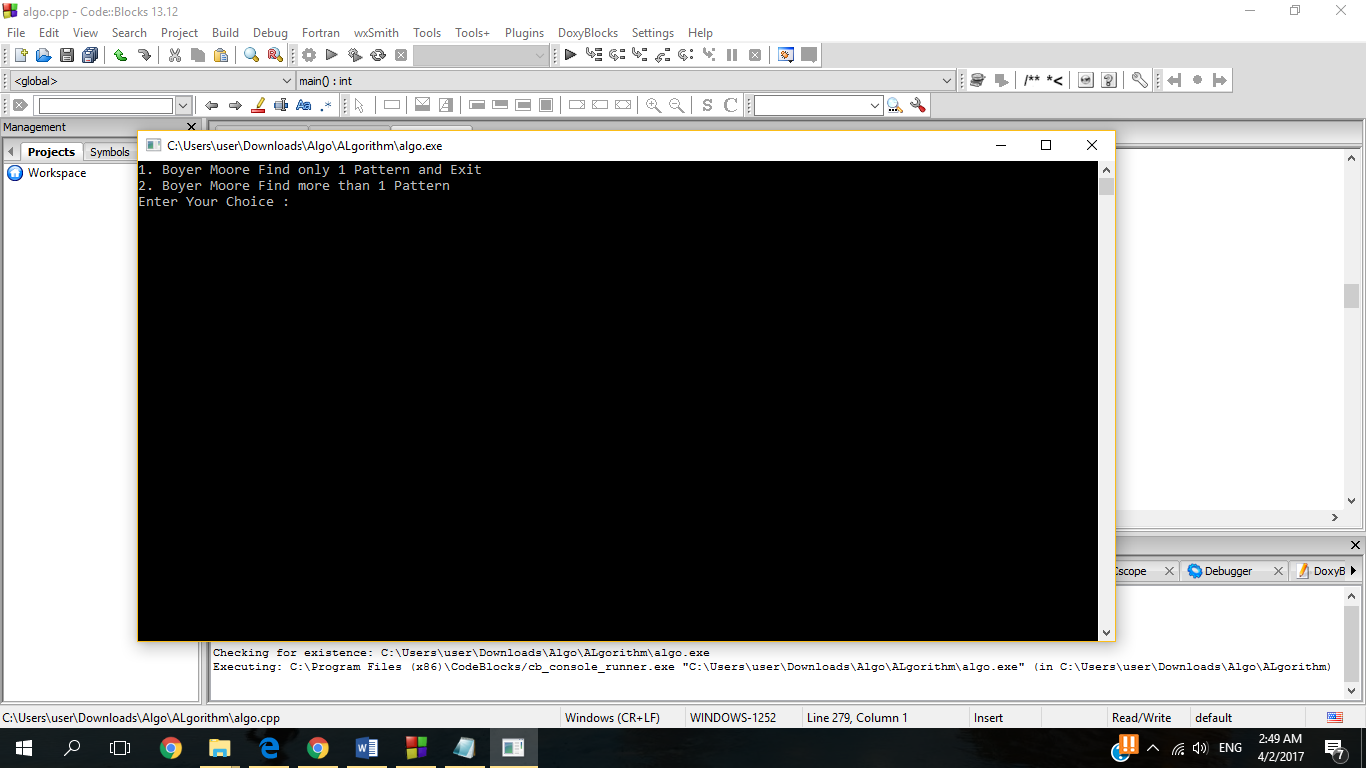
At the 11th matching , it success as the last character of pattern ‘ rithm’ =’m’ match the last character of matching text of ‘rithm’ =’m’ hence it shows a ‘|’ symbol as matching indication.



The Pattern occurs at shift 23 while total pattern match is 1 after 15 times of comparison .This resulted in an average case in between best case and worst case scenario.

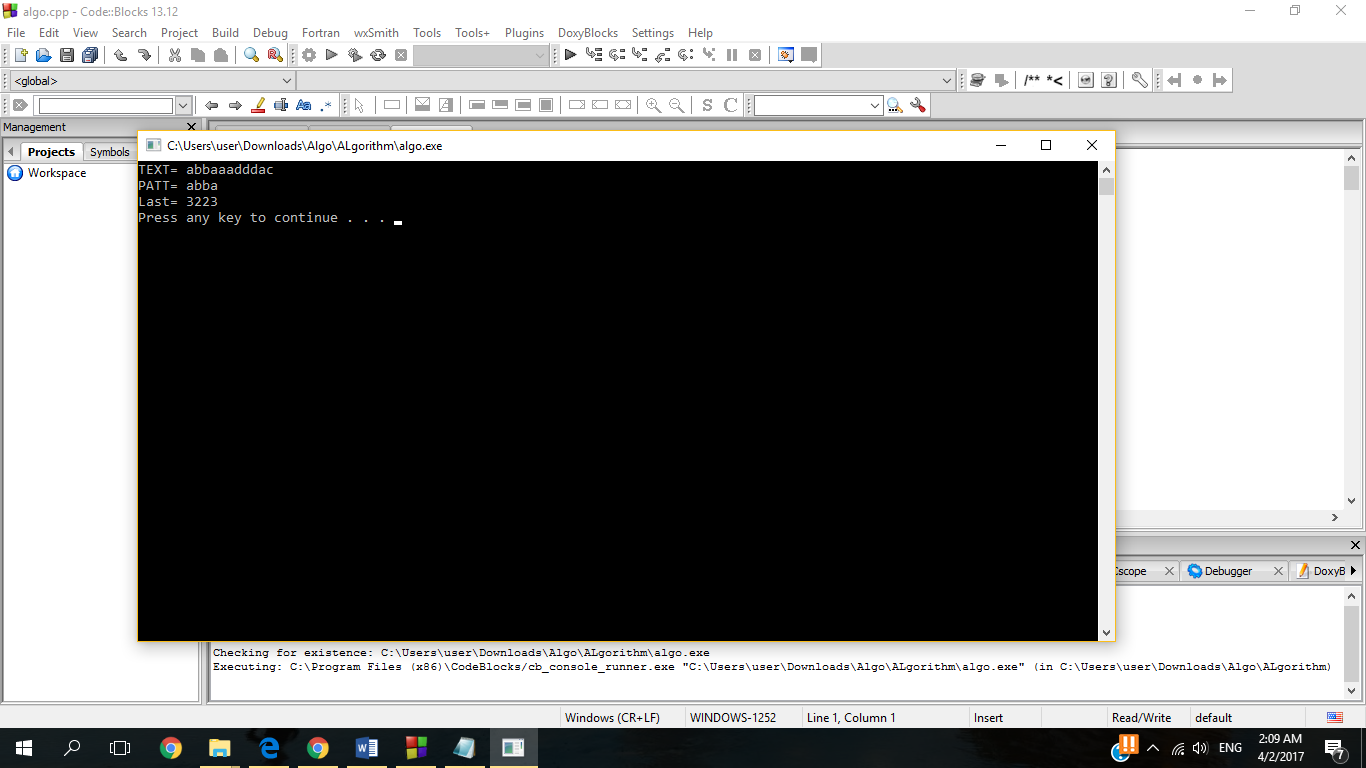
**BEST CASE**

**BEST CASE OPTION 1**



**Best Case Walkthrough (Main Menu):**

Assume select option 1 BoyerMoore find only 1 pattern and Exit

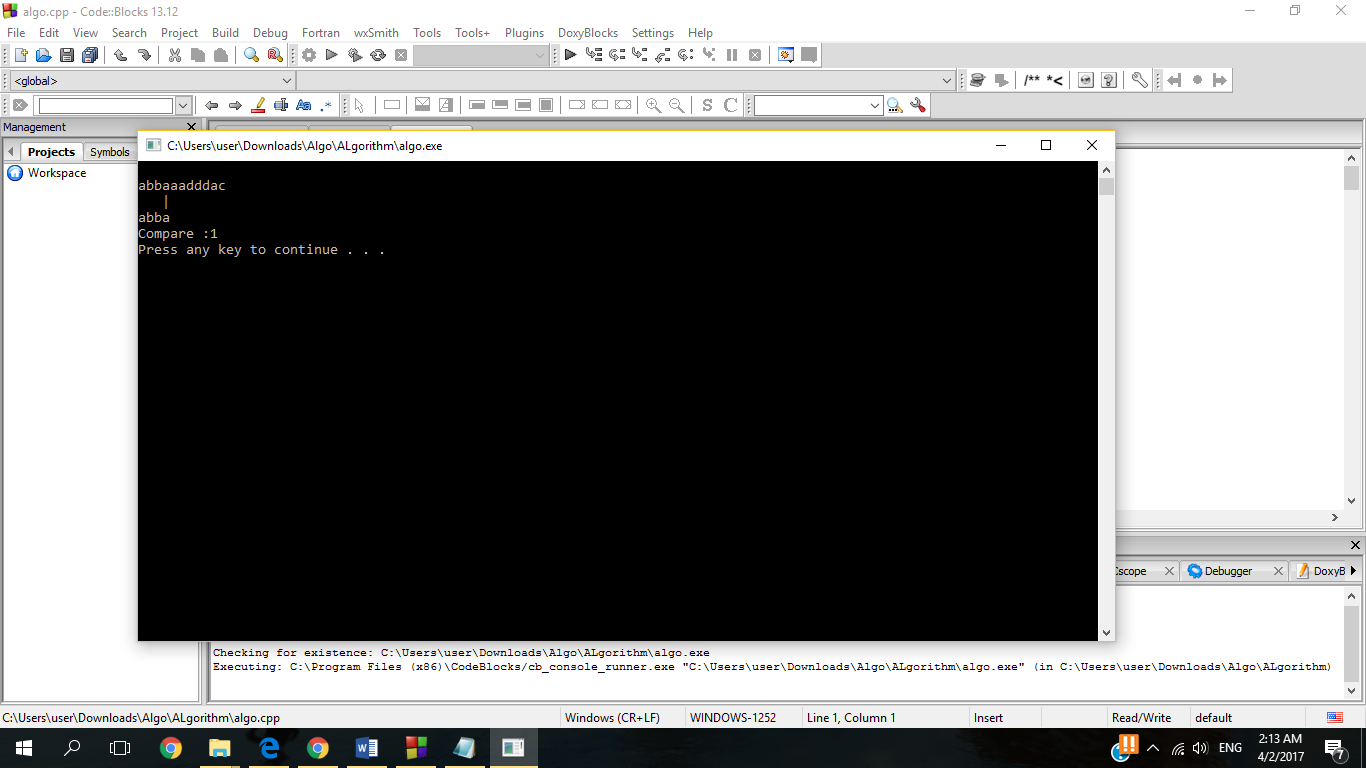


**Best Case Walkthrough 1 :**

Text =abbaaadddac

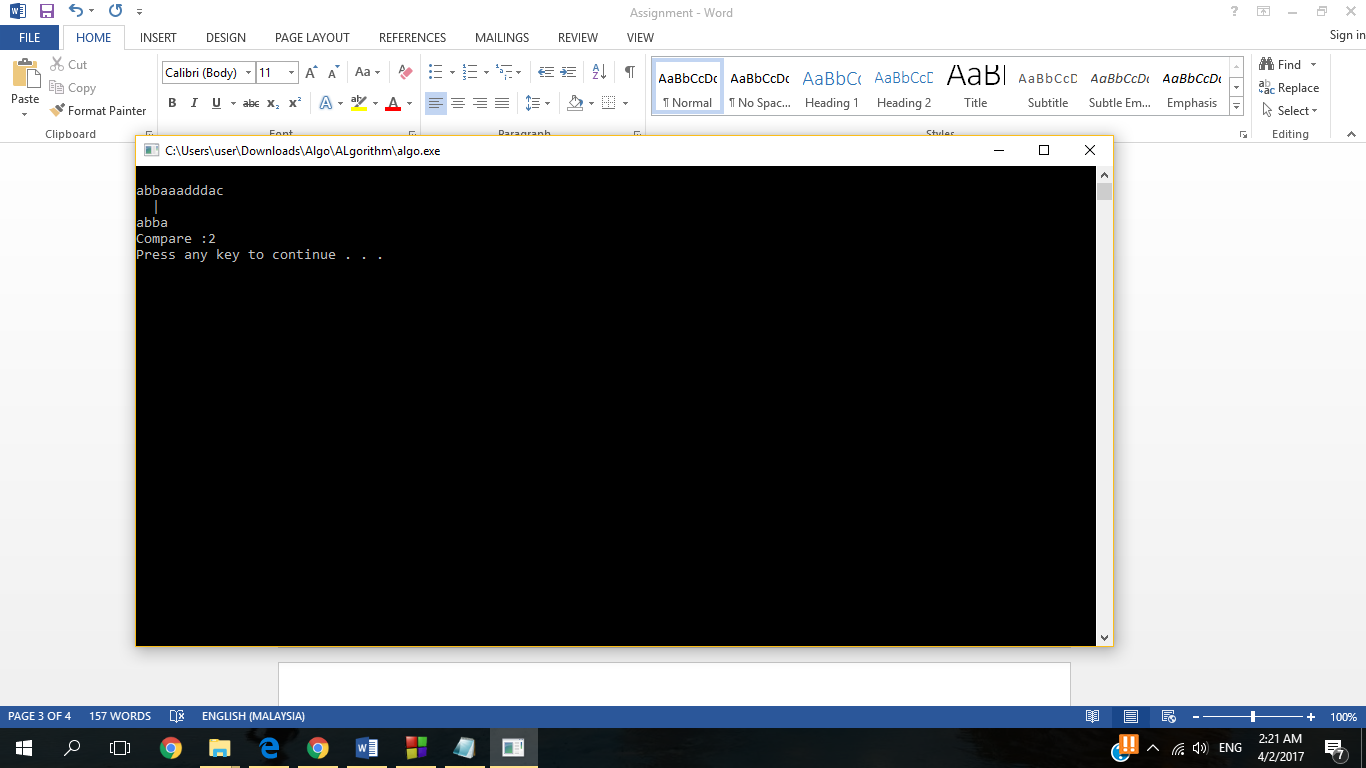
Pattern =abba

Last Occurence =3223



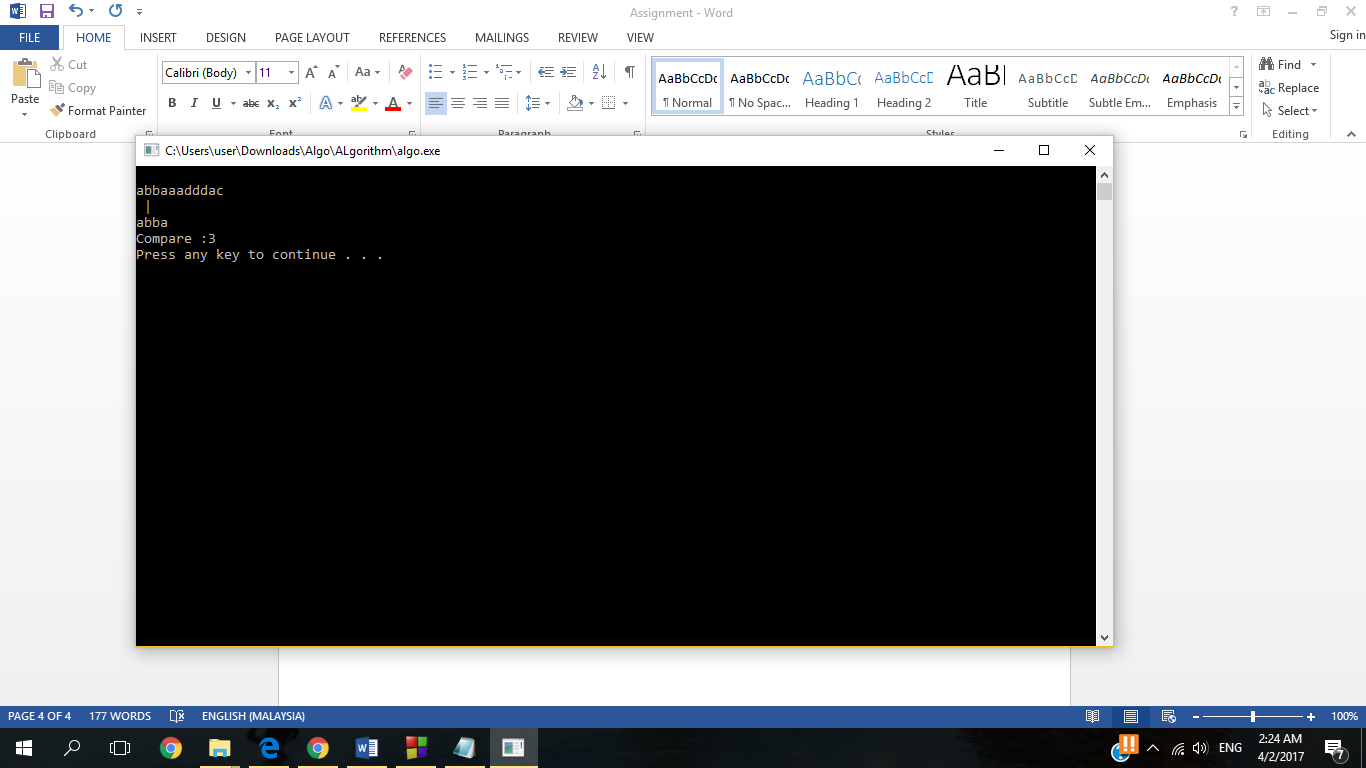
**Best Case Walkthrough 2 :**

The last character of ‘abba’ = ‘a’ is match with last character of pattern ‘abba’= ’a’ , hence the vertical line ‘|’ appear as indication of successful matched.



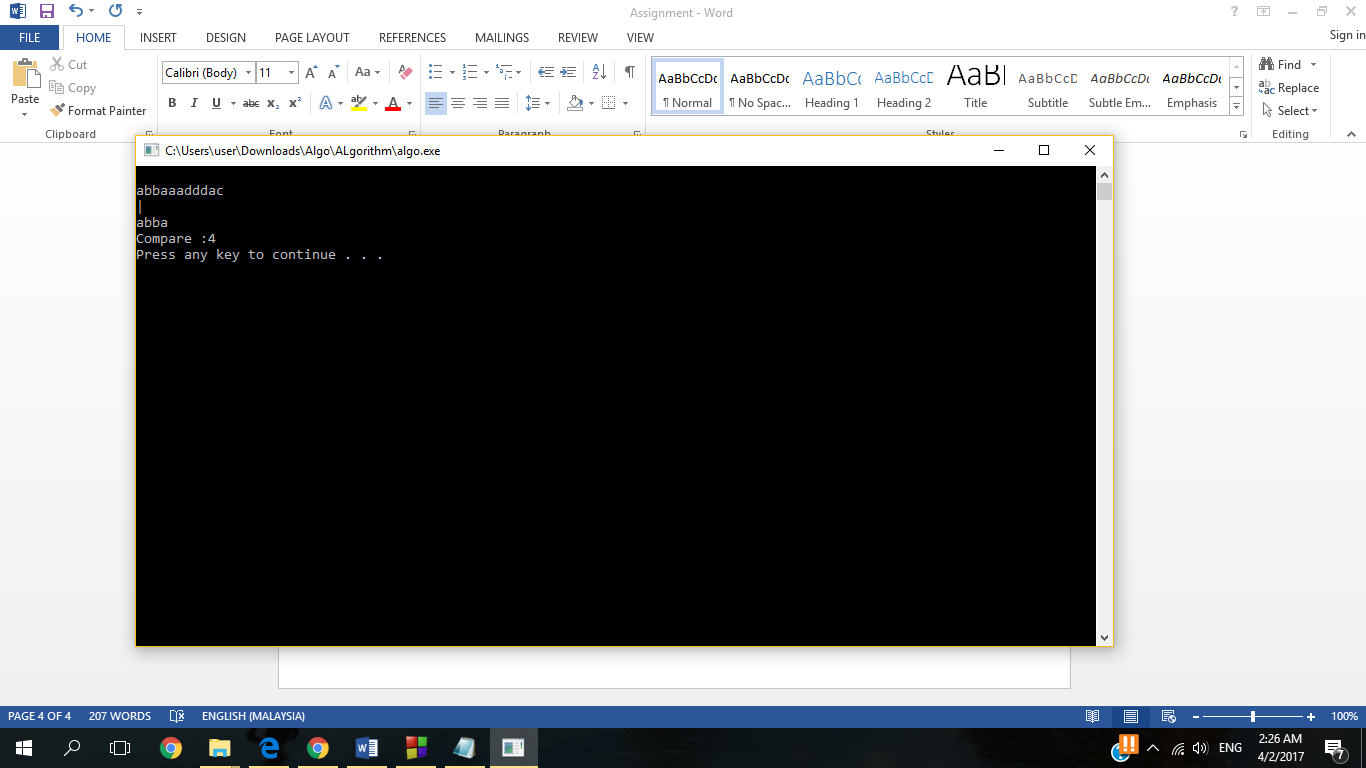
**Best Case Walkthrough 3 :**

Moving on ,the second last character of ‘abba’ = ‘b’ is match with second last character of pattern ‘abba’= ’b’ , hence the vertical line ‘|’ appear as indication of successful matched .



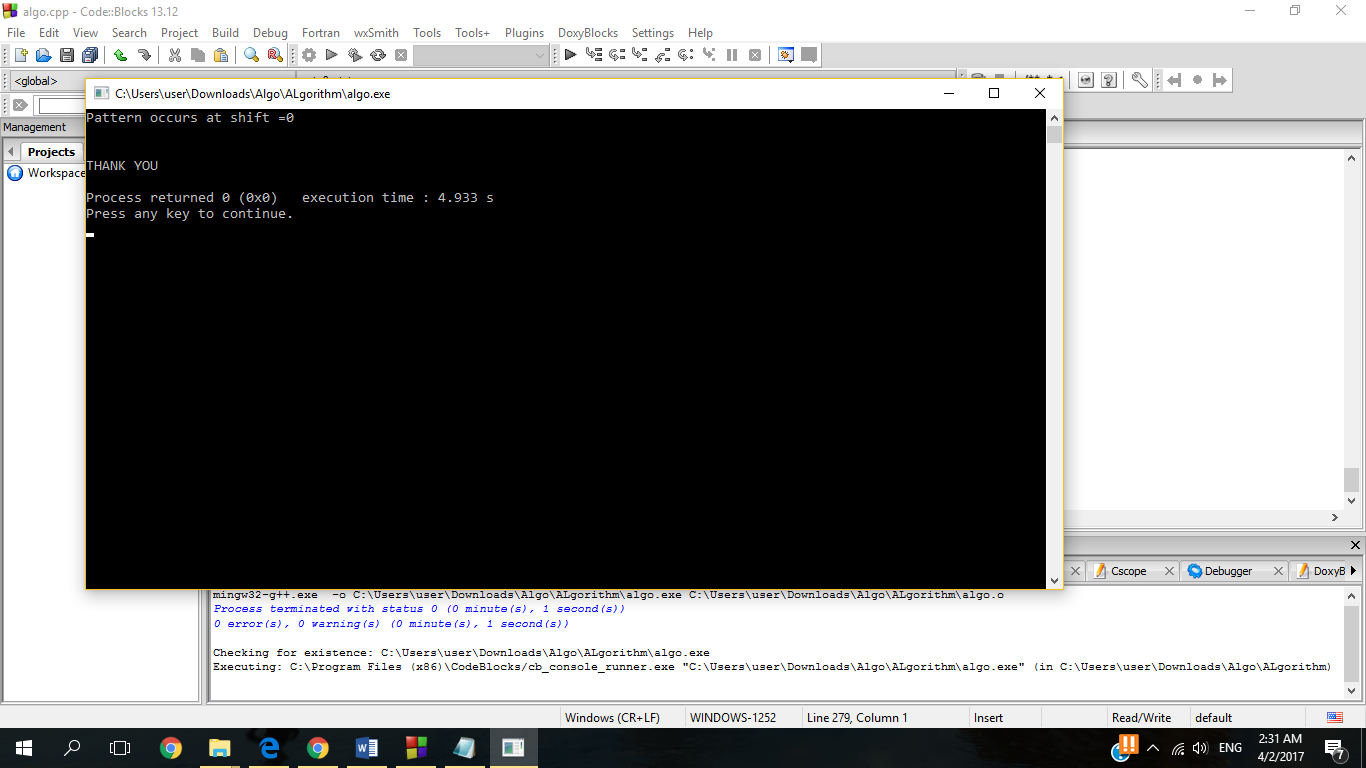
**Best Case Walkthrough 4 :**

The second character of ‘abba’ = ‘b’ is match with second second character of pattern ‘abba’= ’b’ , hence the vertical line ‘|’ appear as indication of successful matched .



**Best Case Walkthrough 5:**

The first character of ‘abba’ = ‘a’ is match with first character of pattern ‘abba’= ’a’ , hence the vertical line ‘|’ appear as indication of successful matched .

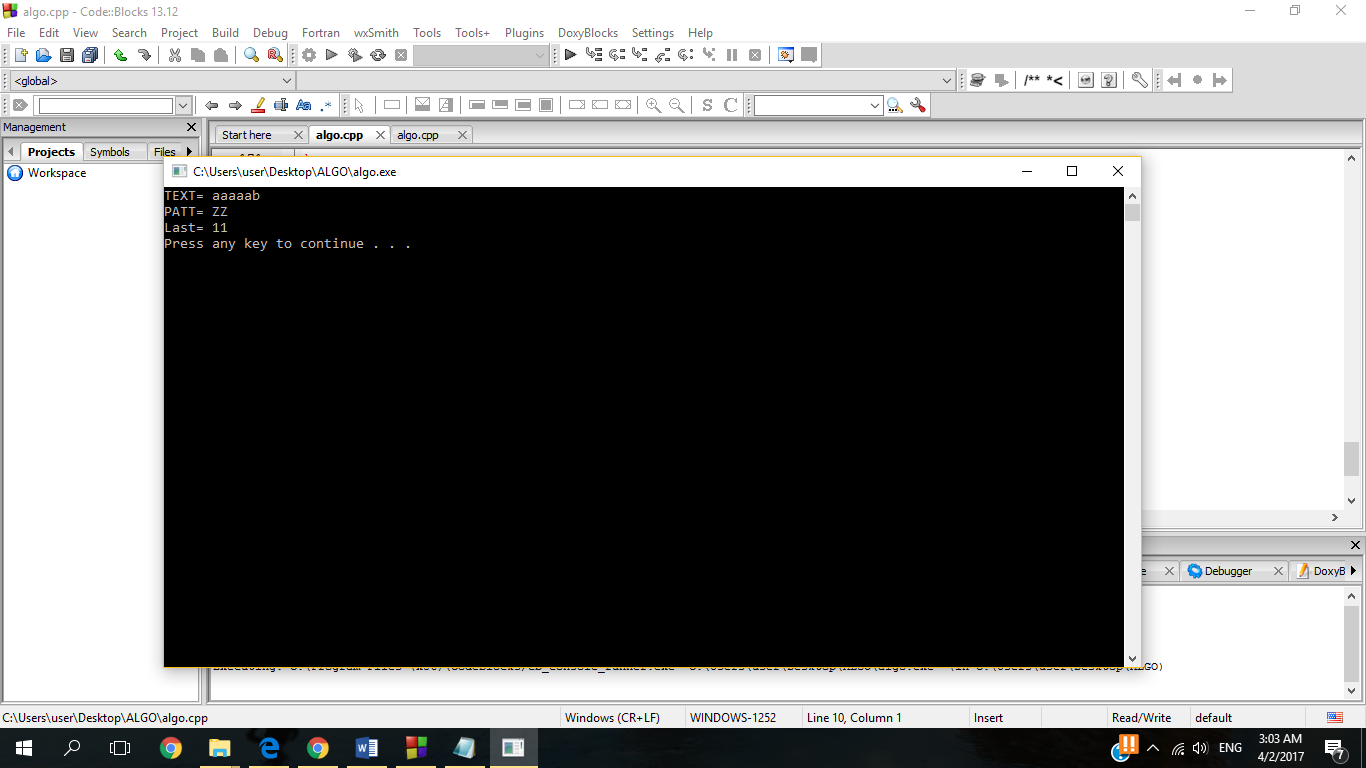


**Best Case Walkthrough 6(Final Result):**

Pattern occurs at shift = 0

This steps indicated best case scenario as it take a very quick attempt to find matching pattern and text.

**Pattern does not exist in the text sample :**

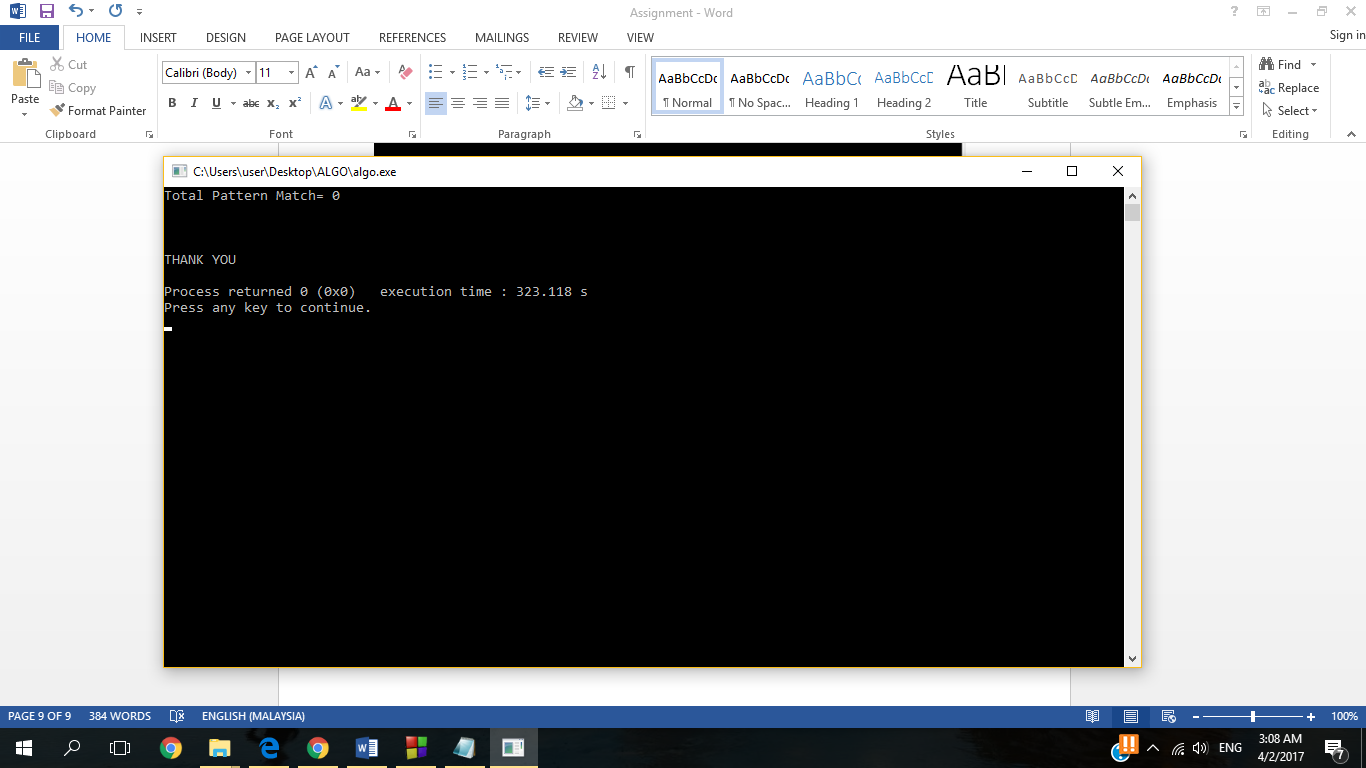


**Walkthrough (main page) :**

Text = aaaaab

Patt = ZZ

Last = 11



Final result : Total Pattern Match = 0 , no pattern match with the text .

Experimental Graph for running time VS size of input between Worst Case and Best Case .

Time (millisecond)

Length of character

Conclusion:

n= Size of text

m=Size of pattern

As the experiment resulted, the worst case when the length of the character hit 100000(100K) the graph slightly increased. The graph increase drastically when length of character 1000000(1M). When number of length 2000000(2M) you realise the graph will increase based on length of char and number of pattern. The result will be different drastically based on number of pattern & text. The number of comparison for worst case will be pattern times text

Worst Case Time complexity is T(n)=O(nm)

As for the average case, when the length of character hit 1000000(1M) the graph increase tremendously until the length of character 2000000(2M). For Average case number of comparison will be based on number of text.

Average : Case Time complexity is T(n)=O(n+m)

As for best case, happen when first “n” character of text immediately match with pattern. This only apply for Boyer Moore find 1 and exit. If length of pattern is 5 and all 5 match with first character from text. Even with text length 1000000(1M) number of comparison will take only 5 for the best case

Best : Case Time complexity is T(n)=O(m)

Reference :

1. <https://en.wikipedia.org/wiki/Boyer%E2%80%93Moore_string_search_algorithm>
2. <https://github.com/zhuozhongcao/Boyer_Moore/blob/master/Boyer_Moore.cpp>
3. <http://www.stoimen.com/blog/2012/04/17/computer-algorithms-boyer-moore-string-search-and-matching/>