

# PD2 Homework 1



Generated by DALL.E



## Announcement:

- Deadline - 2024/3/15 23:59
- Server作業繳交Guideline 請見:

作業繳交Guideline



#### 規則特別說明：

- 以String 'aaa'為例，包含'aa'字串2次；同理，'AAAA'包含'aa'3次
- 請注意 $str_1$ 跟 $str_2$ 不見得都是大寫或小寫，請注意要改為一致全大寫或全小寫。
- $str_1$ 跟 $str_2$ 不會包含 空白 或標點符號 , . 等
- 為確保資源有效使用，我們每一個process最長10分鐘就會kill掉（基本上一個testcase不應該超過1分鐘），這個testcase就會是fail case。另外我們採用java內定的記憶體heap size，如果出現out\_of\_memory exception，則這個testcase也是fail。
  - 我們最後批次執行時，會以256MB做為一個基礎，在java執行時以 -Xmx256m 參數設定，讓JVM的heap size最大為256MB，如：

```
java -Xmx256m RegExp tc1 abc b 3 >ansi
```

- 測資提供tc0-tc5及對應的答案ans0-ans5

[testcase.zip](#)

- 請在command line上執行你的程式，所有的 $str_1$ 跟 $str_2$ 跟次數 $n$ 的值在這幾組測試都是一樣的：

```
java RegExp tc0 abc b 3 > ans0
java RegExp tc1 abc b 3 > ans1
java RegExp tc2 abc b 3 > ans2
java RegExp tc3 abc b 3 > ans3
java RegExp tc4 abc b 3 > ans4
java RegExp tc5 abc b 3 > ans5
```



### Rule Explanation:

- For example, the string 'aaa' contains the substring 'aa' twice; similarly, 'AAAA' contains 'aa' 3 times.
- Please note that  $str_1$  跟  $str_2$  may not be all uppercase or lowercase. Please make sure to change it to all uppercase or lowercase.
- $str_1$  跟  $str_2$  will not contain `space` or punctuation marks like `,` `.` etc.
- To ensure efficient use of resources, we kill each process after a maximum of 10 minutes (a testcase should not take more than 1 minute), and this testcase would be a fail case. Also, we use the default memory heap size in Java, if an `out_of_memory` exception occurs, then this testcase is also a fail.
  - When we execute in batches at the end, we set 256MB as a base, when executing in Java, we set the parameter `-Xmx256m`, so that the JVM's heap size is at most 256MB, like:

```
java -Xmx256m RegExp tc1 abc b 3 >ansi
```

- Test data provides tc0-tc5 and corresponding answers ans0-ans5  
[testcase.zip](#)
- Please run your program on the command line. The values of  $\$str\_1$ ,  $\$str\_2$ , and count  $\$n$  are the same in all these test groups:

```
java RegExp tc0 abc b 3 > ans0
java RegExp tc1 abc b 3 > ans1
java RegExp tc2 abc b 3 > ans2
java RegExp tc3 abc b 3 > ans3
java RegExp tc4 abc b 3 > ans4
java RegExp tc5 abc b 3 > ans5
```

# Regular Expression

正則表達式（Regular Expression）是一種用來進行字串Pattern Matching的強大工具，廣泛使用的應用包含：

1. 確認字串是否符合特定的模式。例如，我們可以用正則表達式來確認一個字串是否為電子郵件地址的格式。
2. 在一個string中尋找符合特定模式的substring。如果找到，正則表達式可以提取出這些substring，或者提供這些substring的位置。
3. 替換符合特定模式的子字串。例如，我們可以用正則表達式來將一個文本中的所有中式日期格式（如112/12/31）替換為美式日期格式（如2023/12/31）。

雖然各種程式語言均有很強的regular expression功能，例如在java中，透過

`java.util.regex` package可以實作複雜的regular expression應用，但在本次作業，我們將要求：



不能使用 `java.util.regex`，實現以下的功能：

1. 確認是否是迴文，是的話回答Y，不是的話回答N
2. 確認是否包含特定字串 $str_1$ ，是的話回答Y，不是的話回答N
3. 確認是否包含特定字串 $str_2$ 超過或等於n次，是的話回答Y，不是的話回答N
4. 確認是否包含字串 $a^m X b^{2m}$ , where  $m \geq 1$ , and  $X$  is any string (empty is ok).

其中

$a^m$ 指的是a或A連續重覆m次，而 $b^{2m}$ 指的是b或B連續重覆2m次。

如果符合包含字串

$a^m X b^{2m}$ 的話，回答Y, 不包含的話回答N



Implement the following functions without using `java.util.regex`:

1. Check if it is a palindrome, if yes, answer Y, if not, answer N.
2. Check if it contains a specific string  $str_1$ , if yes, answer Y, if not, answer N.
3. Check if it contains a specific string  $str_2$  equal to or more than  $n$  times, if yes, answer Y, if not, answer N.
4. Check if it contains the string  $a^m X b^{2m}$ , where  $m \geq 1$ , and  $X$  is any string (empty is ok).

Where

$a^m$  refers to a or A repeated  $m$  times sequentially, and  $b^{2m}$  refers to b or B repeated  $2m$  times sequentially.

If it contains the string

$a^m X b^{2m}$ , answer Y, if not, answer N.

## Input issues

- 每一行視為一個string，以斷行符號區別不同的string
- 每一行的text包含a-zA-z、空白符號、英文句點及分號等
- 不區分英文大小寫，所以A=a
- palindrome issue:
  - 如果string的長度是奇數，如 ABC\_to\_ot\_CBA (
- Each line is considered as one string, distinguished by line breaks.
- Each line of text contains a-zA-z, blank symbols, English periods, and semicolons, etc.
- English does not distinguish between upper and lower case, so A=a
- Palindrome issue:
  - If the length of the string is odd, like ABC\_to\_ot\_CBA

␣ 符號代表空白)

則答案為Y

- 如果string的長度是偶數，如  
ABCcba  
則答案為Y
- test file最多會包含1000行，每一行  
最多80個character
- An example input file

```
This is a bug, but is fixed.  
ABC to ot CBA  
ABCBA  
ABCCBA  
AA  
aA  
Aaa aBbBbbbbb  
Abmxabcbbc
```

('\_' symbol represents a space)

Then the answer is Y

- If the length of the string is  
even, like  
ABCcba  
Then the answer is Y
- The test file will contain up to 1000  
lines, with a maximum of 80  
characters per line

## Input Arguments

將有4個參數輸入，分別對應args[0]~args[4]。舉例：

```
java RegExp tc1 abc b 3
```

- args[0]: test case file name
- args[1]: 字串 $str_1$ ，在本例為abc
- args[2]: 字串 $str_2$ ，在本例為b
- args[1]: String  $str_1$ , in this case  
abc
- args[2]: String  $str_2$ , in this case b

- args[3]: 字串  $str_2$  超過或等於  $n$  次，在本例  $n$  為 3
- args[3]: String  $str_2$  exceeds or equals  $n$  times, in this case  $n$  is 3

## Output issues

- 請直接將結果 output 在 screen，你可以直接使用 `System.out.print()` 或 `System.out.println()` 將結果輸出
- Please output the results directly on the screen. You can use `System.out.print()` or `System.out.println()` to output the results.
- Output 中的每一行，包含 4 個 Y 或 N (均為大寫，以英文 , 分隔)，分別對應 (1) 是否是迴文、(2) 是否包含特定字串  $str_1$ 、(3) 是否包含特定字串  $str_2$  超過或等於  $n$  次、(4) 是否包含字串  $a^m X b^{2m}$ 。如下格式：
- Each line of the output contains 4 Y or N (all in uppercase, separated by English ,), corresponding to (1) whether it is a palindrome; (2) whether it contains a specific string  $str_1$ ; (3) whether it contains a specific string  $str_2$  more than or equal to  $n$  times; (4) and whether it contains the string  $a^m X b^{2m}$ . The format is as follows:

```
Y, N, N, N
N, Y, Y, N
```

- 我們在 linux server 將會直接 capture 你的螢幕輸出到 result 檔，做為跟 golden answer 比較：
- We will directly capture your screen output to the result file in our Linux server, as a comparison with the golden answer:

```
java RegExp tc1 abc b 3 > result_tc1
```



- 對應上面的input，你應該會得到以下的output：

- In response to the above input, you should get the following output:

```
N, N, N, N
Y, Y, N, N
Y, Y, N, N
Y, Y, N, N
Y, N, N, N
Y, N, N, N
N, N, Y, Y
N, Y, Y, Y
```



### Homework Ranking

- 在deadline前，一共會提供5個test cases給大家, named as tc1~tc5
  - 同時也將提供對應的答案ans1~ans5
- 在繳交deadline之後，我們一共會用10個test cases, named as tc1~tc10, 測驗你們的程式，其中tc6~tc10不會給大家，會在HW檢測完提供。
- 你可以使用 `String` 類別所有的method來實現這個Homework的要求，在10個test cases都pass之後，你可以獲得10 pt。
- 如果只有1個test cases fails, 9 pts.
- 如果只有2個test cases fails, 8 pts.
- 如果只有3個test cases fails, 7 pts.
- 如果超過3個test cases fails, 0 pts.





### Homework Ranking

- Before the deadline, a total of 5 test cases will be provided to everyone, named as tc1~tc5
  - At the same time, the corresponding answers ans1~ans5 will also be provided
- After the submission deadline, we will use a total of 10 test cases, named as tc1~tc10, to test your program, of which tc6~tc10 will not be given to everyone, and will be provided after the HW is finally checked.
- You can use all the methods in the `String` class to implement the requirements of this Homework. After all 10 test cases have passed, you can get 10 pt.
- If only 1 test case fails, 9 pts.
- If only 2 test cases fail, 8 pts.
- If only 3 test cases fail, 7 pts.
- If **more than 3 test cases fail**, 0 pts.



### Homework Challenge (1 pt)

除了不使用

`java.util.regex` 的要求之外，你也沒有使用到任何有實作到regex或index的Method，如String類別中的split()或matches()、contains()、indexOf()、lastIndexOf()等，則可以獲得Homework Challenge的1 pt



### Homework Challenge (1 pt)

In addition to the requirement not to use

`java.util.regex`, if you also do not use any method that use regex or index, such as split(), matches(), contains(), indexOf(), or lastIndexOf() in the String class, you can get 1 pt of the Homework Challenge.

## Homework Validation

- 於測資的地方也會提供 `validate.py` (`/home/share/hw{n}/validate.py`) 程式協助同學檢查自己程式於測試資料上執行的正確性
- 同學請在server將`validate.py`複製一份到`/home/{你的學號}/hw1`，也就是與你的`RegExp.java`相同資料夾
- `python3 validate.py` 指令是用來運行所有的 testcase，裡面將您的 `RegExp.java` 加上測資得到輸出後，使用 `diff` 與正確答案進行比對。
- 如果您想要分開執行testcase，可以在 `python3 validate.py` 後面加上參數 0~5。例如，執行 `python3 validate.py 0` 將會運行 `testcase0`。
- 對於此程式有發現任何問題，歡迎寄信到助教信箱，謝謝!

## Exception Consideration

- 請不用考慮任何輸入上的錯誤問題，例如給錯檔名或`args[3]`不是整數數字
- Please do not consider any input errors, such as giving the wrong file name, or `args[3]` is not an integer value.
- 不會有input argument個數不等於4的情況，但你思考這種情況是很好的習慣
- There won't be a situation where the number of input arguments is not equal to 4, but it's a good habit to consider this kind of situation.
- 你必需考慮到如果一個file是empty content
- You have to consider if a file is empty content
- 你可以直接assume以下條件：
- You can directly assume the following conditions:

1. `length(args[1]) ≥ 1` and `length(args[1]) ≤ 10`
2. `length(args[2]) ≥ 1` and `length(args[2]) ≤ 10`
3. `args[3] ≥ 1` and `args[3] ≤ 10`

## Tip

以下是一個Java範例，示範如何從一個文字檔中讀取每一行，並將每一行視為一個String處理。你可以用此範例做為出發。

Here is a Java example, demonstrating how to read each line from a text file, and treating each line as a String. You can use this example as a starting point.

```
import java.io.BufferedReader;
import java.io.FileReader;
import java.io.IOException;

public class RegExp {

    public static void main(String[] args) {
        String str1 = args[1];
        String str2 = args[2];
        int s2Count = Integer.parseInt(args[3]);

        //For your testing of input correctness
        System.out.println("The input file:"+args[0]);
        System.out.println("str1="+str1);
        System.out.println("str2="+str2);
        System.out.println("num of repeated requests of str2
= "+s2Count);

        try {
            BufferedReader reader = new BufferedReader(new Fi
leReader(args[0]));
            String line;
            while ((line = reader.readLine()) != null) {
                //You main code should be invoked here
                System.out.println(line);
            }
            reader.close();

        } catch (IOException e) {
```

```
        e.printStackTrace();  
    }  
}  
}
```