A - Scanner Test

請注意不能用 #include<regex> 抓到 BA 0分!!! 請注意不能用 #include<regex> 抓到 BA 0分!!! 請注意不能用 #include<regex> 抓到 BA 0分!!!

Description

在編譯器中, Token是組成程式碼的最小單位,需要由 Scanner 先把 Input text 轉換Token, Parser才會進行後續的工作。請利用表一,撰寫一個 Scanner來取得Token並輸出。

_	
=	
1X	

Terminal	Regular Expression
NUM	[1-9][0-9]* 0
ID	[A-Za-z]+
ОР	[\+\ - */ =]
LPR	γ
RPR	\)

Input Format

輸入一行"運算式,每筆測試資料只會有一個運算式,但其中可能會夾雜多個空格。

部分輸入的程式碼將會有不符合 Regular Expression 的情況發生。

Output Format

請在切割後輸出其 Token 種類。

若為數字, 需附上其數值, 並以一個空白做為區隔。

若為運算子, 需附上其符號, 並以一個空白做為區隔。

例如 0 則需輸出 NUM 0. 以此類推

例如 + 則需輸出 OP +. 以此類推

若不符合Regular Expression, 即使只出現一次錯誤, 則只需印出"invalid output", 前面合法輸出則不印出。

每個 token 輸出後請以\n 分隔。

Sample input 1	Sample output 1
int a = 10	ID int
	ID a
	OP =
	NUM 10
Sample input 2	Sample output 2

A/40 - 44 - 42)	ID A
A(10 + 11 + 12)	ID A
	LPR
	NUM 10
	OP +
	NUM 11
	OP +
	NUM 12
	RPR
Sample input 3	Sample output 4
"int	invalid output
Sample input 4	Sample output 4
A(10+11+12)\$	invalid output
Sample input 5	Sample output 5
K (A(10 + 11 + 12) * Hello (999))	ID K
	LPR
	ID A
	LPR
	NUM 10
	OP +
	NUM 11
	OP+
	NUM 12
	RPR
	OP *
	ID Hello
	LPR
	NUM 999
	RPR
	RPR