

STUDENT PORTFOLIO

Insert Photo



Name: **Sambhram Ghosh**

Register Number: **RA2011031010036**

Mail ID: **sg9732@srmist.edu.in**

Department: **NWC CSE**

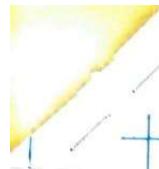
Specialization: **CSE-Information Technology**

Semester: **5th**

Subject Title: **18CSC301J Formal Language & Automata**

Handled By: **Dr.P.Balaji Srikaanth (102776)**

COMPLETION OF WORKSHEET



FLA - assignment

Set-1 Part-B

11.



NFA consists of 5 tuples $(Q, \Sigma, q_0, F, \delta)$

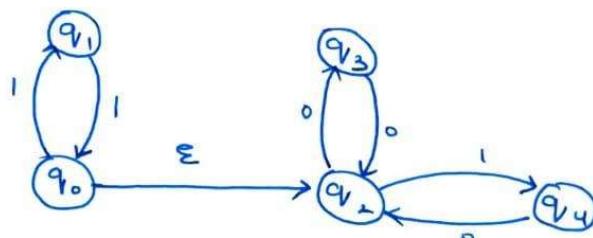
Q : a set of all states

Σ : a set of input symbols (Symbols which machine takes as an input)

q_0 : Initial State (Starting State of a machine)

F : a set of final states

12.



$$\epsilon\text{-closure}(q_v0) = \epsilon q_v0, q_v2, q_v3 \quad \text{--- A}$$

$$\delta(A, 0) = \epsilon\text{-closure}(\delta(\{\epsilon q_v0, q_v2\}, 0))$$

$$= \epsilon\text{-closure}(\emptyset \cup q_v3) = q_v3 \quad \text{--- B}$$

$$\delta(A, 1) = \epsilon\text{-closure}(\delta(\{\epsilon q_v0, q_v2\}, 1))$$

$$= \epsilon\text{-closure}(q_v1 \cup q_v2)$$

$$= \{q_v1, q_v2\} \quad \text{--- C}$$

$$\textcircled{B} \quad \delta(B, 0) = \epsilon\text{-closure}(q_v3, 0) = \epsilon\text{-closure}(q_v2)$$

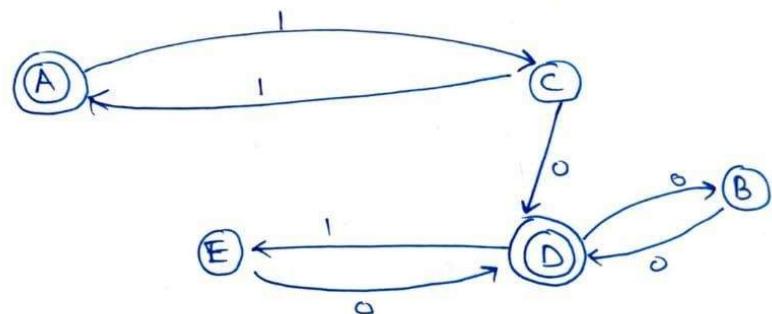
$$= q_v2 \quad \text{--- D}$$

$$\delta(B, 1) = \emptyset$$

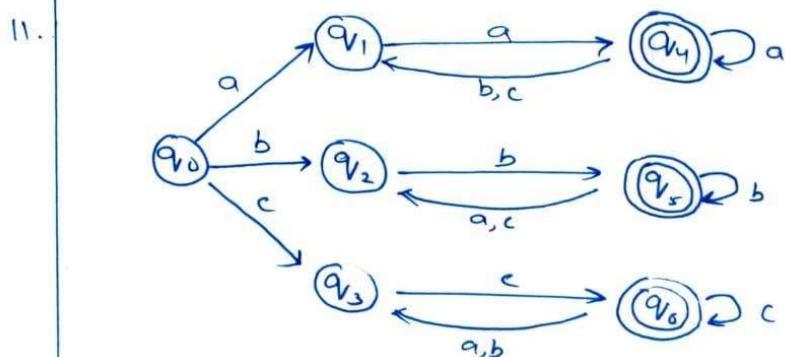
$$\begin{aligned}
 c) \quad \delta(c, 0) &= \varepsilon\text{-close}(\{q_1, q_4\}, 0) \\
 &= \varepsilon\text{-close}(\emptyset \cup q_2) = q_2 \longrightarrow \textcircled{D} \\
 \delta(c, 1) &= \varepsilon\text{-close}(\{q_1, q_4\}, 1)
 \end{aligned}$$

$$\begin{aligned}
 d) \quad \delta(d, 0) &= \varepsilon\text{-close}(\{q_2, 0\}) = \varepsilon\text{-close}(q_3) \\
 &= q_3 \longrightarrow \textcircled{B} \\
 \delta(d, 1) &= \varepsilon\text{-close}(q_2, 1) = \varepsilon\text{-close}(q_4) \\
 &= q_4 \longrightarrow \textcircled{E}
 \end{aligned}$$

$$\begin{aligned}
 e) \quad \delta(e, 0) &= \varepsilon\text{-close}(q_4, 0) = q_2 \longrightarrow \textcircled{D} \\
 \delta(e, 1) &= \varepsilon\text{-close}(q_4, 1) = \emptyset
 \end{aligned}$$



Set - B
Part - B



DFA consists of 5 tuples $\{Q, \Sigma, q_0, F, \delta\}$

Q : set of all states

Σ : a set of input symbols (symbols which machine takes as input)

q_0 : Initial state (starting state of a machine)

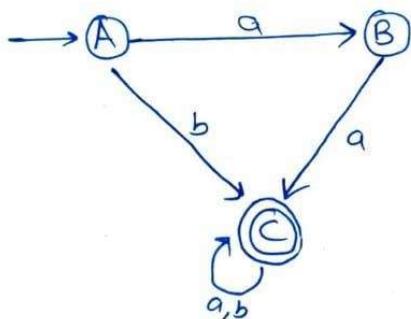
F : a set of final state

δ : Transition function, defined as

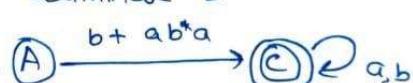
$$\delta: Q \times \Sigma \rightarrow Q.$$

Part-C

12.



eliminate - B



$$\begin{aligned} RE &= (\phi + (b + ab^*a)(a+b)^*d)^* \\ &\quad (b + ab^*a)(a+b)^* \\ &= (b + ab^*a)(a+b)^* \end{aligned}$$

Alternatively Rij method can be used

FLA - Worksheet-2

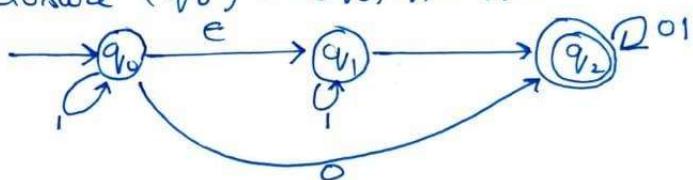
1. Set of binary strings that begin with zero or more instances of 0's and contains zero or more instances of substring 101 and ends with 11

2. A language consists of any combination of 0's & 1's beginning and endings with string '01'.

$$01 (0+1)^* 01$$

3. Yes $([a-z])^*$ substring $([a-z])^*$

4. ϵ -closure $(q_0) = \{q_0, q_1, q_2\}$



5. A deterministic finite automata (or DFA) is a quintuple $D = (Q, \Sigma, \delta, q_0, F)$ where
 Σ is a finite input alphabet
 Q is a finite set of states
 F is a subset of Q of final (or accepting) states

$q_0 \in Q$ is the start state (or initial state);

δ is the transition function, a function

$$\delta : Q \times \Sigma \rightarrow Q$$

A non-deterministic finite automata (NFA) is a quintuple $N = (Q, \Sigma, \delta, q_0, F)$ where
 Σ is a finite input alphabet;

Q is a finite set of states;

F is a subset of Q of final (or accepting) states;

$q_0 \in Q$ is the start state (or initial state);

δ is the transition function, a function

$$\delta : Q \times (\Sigma \cup \{\epsilon\}) \rightarrow 2^Q$$

MCQ →

1. i) set of all strings starting and ending with '10' and any number of 1's in b/w '10'.
2. ii) $L = (0+1)^*1001(0+1)^*$
3. c) (ii) and (iii)
4. i) Regular Languages

FLA Assignment - 1

Q- Show that $2^n - 1$ is divisible by 3 using the principle of mathematical induction

Sol- Basis \rightarrow Let the statement $P(n)$ given as

$P(n) : 2^n - 1$ is divisible by 3, for every natural number n .

We observe that $P(1)$ is true.

Since $2^1 - 1$ let $(n) = 1$

$$\rightarrow P(n) = 2^{2(1)} - 1 = 2^2 - 1 = 4 - 1 = 3$$

as RHS is divisible by 3

\therefore Basis Induction is true.

Induction Hypothesis -

Let $(P(n))$ is true for some natural number k

$P(k) : 2^k - 1$ is divisible by 3

$$2^k - 1 = 3q \text{ where } q \in N$$

Induction Proof \rightarrow

Now to prove that $P(k+1)$ is also true

$$\rightarrow P(k+1) : 2^{2(k+1)} - 1 = 2^{2k+2} - 1 = 2^{2k} \cdot 2^2 - 1$$

$$\rightarrow 2^{2k} \cdot 4 - 1 = 3 \cdot 2^{2k} + (2^{2k} - 1) = 3 \cdot 2^{2k} + 3q$$

$$\rightarrow 3(2^{2k} + q) = 3m, \text{ where } m \in N$$

Thus $P(k+1)$ is true when ever $P(k)$ is true

Hence, by Principle of Mathematical Induction
 $P(n)$ is true for all natural Number n .

Q- If a^2 is divisible by 3 then a is divisible by 3 proof by contradiction

Let $a^2 = 3k$ a^2 is div by 3

$a = (3k+1)$ or $(3k+2)$ a div by 3

Let a^2 is div by 3 $a^2 = 3k$

a is div by 3 $a = (3k+1)$ or $(3k+2)$

$a^2 = (3k+1)^2$ or $(3k+2)^2$

$a^2 = (9k^2 + 6k + 1)$ or $(9k^2 + 12k + 4)$

$a^2 = 3(3k^2 + 2k) + 1$ or $3(3k^2 + 4k + 1) + 1$

$k \in \mathbb{Z}$

$a^2 = 3c + 1$ or $3d + 1$

$\rightarrow \therefore a^2$ is not div by 3

$\therefore a$ is divisible by 3

Q- if $a+b$ is odd then either a or b is odd.

Ans Let a and b be integers. Assume that a and b are even (by contradiction)

Then $a = 2k$

$b = 2l$ for some integer
 k and l .

Now $a+b = 2k+2l = 2(k+l)$

Since $k+l$ is an integer, we can easily see that $a+b$ is even, completing the proof

Q- For any natural number $n \geq 5$, $n^2 < 2^n$

Sol- By induction on n , As a base case if $n=5$,
then we have that $5^2 = 25 < 32 = 2^5$,
so claim holds.

For inductive step, assume then for some
 $n \geq 5$ that $n^2 < 2^n$. Then we have that

$$(n+1)^2 = n^2 + 2n + 1$$

Since $n \geq 5$ we have -

$$(n+1)^2 = n^2 + 2n + 1$$

$$< n^2 + 2n + n \quad (\text{since } 1 < 5 \leq n)$$

$$= n^2 + 3n \quad (\text{since})$$

$$3n < 5n \quad (5n \leq n^2)$$

$$= 2n^2 + n$$

So $(n+1)^2 < 2n^2$. Now by our inductive hypothesis, we know that $n^2 < 2^n$. This means that,

$$(n+1)^2 < 2n^2 \quad (\text{from above})$$

$$< 2(2^n) \quad (\text{by inductive hypothesis})$$

$$= 2^{n+1}$$

Completing the induction

(Q) - For all $n \geq 1$, prove that

Sol - $1^2 + 2^2 + 3^2 + 4^2 + \dots + n^2 = \frac{n(n+1)(2n+1)}{6}$

Let $P(n) : 1^2 + 2^2 + 3^2 + \dots + n^2 = \frac{n(n+1)(2n+1)}{6}$

Base \rightarrow Let $n=1$

LHS = $1^2 = 1$ RHS = $\frac{1(1+1)(2 \times 1+1)}{6} = \frac{1 \times 2 \times 3}{6} = 1$

$\therefore \text{LHS} = \text{RHS}$

Inductive hypothesis \rightarrow

$\forall P(k)$ is true \rightarrow

$P(k) : 1^2 + 2^2 + 3^2 + \dots + k^2 = \frac{k(k+1)(2k+1)}{6} \quad (1)$

\therefore For $P(k+1) \rightarrow$

$P(k+1) = 1^2 + 2^2 + 3^2 + \dots + k^2 + (k+1)^2 = \frac{(k+1)((k+1)+1)(2(k+1)+1)}{6}$

$\Rightarrow 1^2 + 2^2 + 3^2 + \dots + k^2 + (k+1)^2 = \frac{(k+1)(k+2)(2k+3)}{6}$

$\Rightarrow \frac{k(k+1)(2k+1)}{6} + (k+1)^2$

$\Rightarrow \frac{(k+1)(k(2k+1) + 6(k+1)))}{6}$

$\Rightarrow \frac{(k+1)(2k^2+7k+6)}{6} = \frac{(k+1)(2k^2+4k+3k+6)}{6}$

$= \frac{(k+1)(2k+3)(k+2)}{6}$

$= \text{RHS}$

$\therefore P(k+1)$ is true when $P(k)$ is true

This by principle of mathematical induction.

$P(n)$ is true for n , where n is a natural number.

M.C.Q \rightarrow

1. b) 3. d)

UNIT-4 Worksheet - 1

1. True

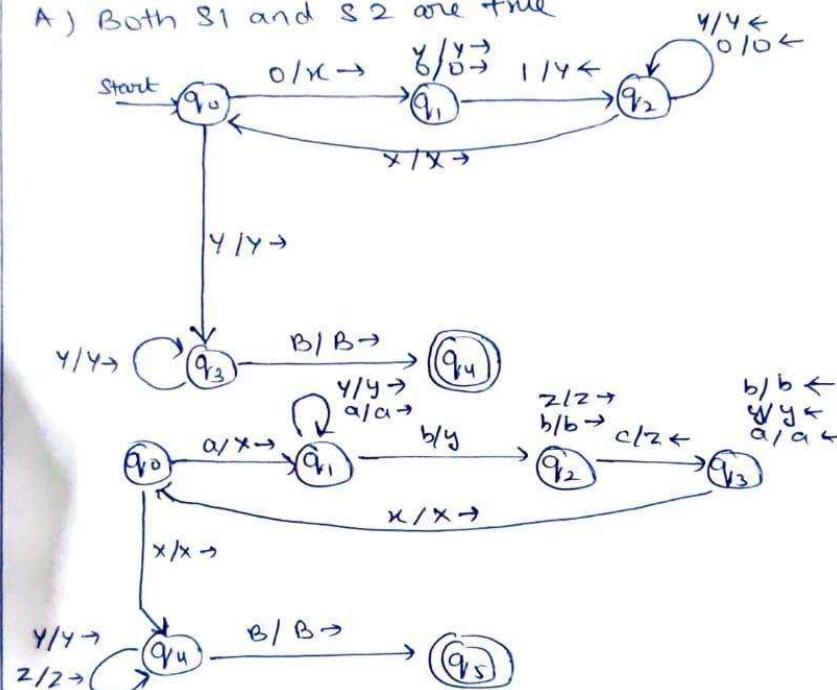
2. True

3. B) 7

4. A) Left

5. A) Both S1 and S2 are true

6.



2. Multi Tape (MultiTrack) Turing Machine
 Multi Dimensional Turing Machine
 Two Way infinite tape TM

State	Symbol				
	0	1	X	Y	B
q_{00}	(q_1, X, R)	\leftarrow	-	(q_3, Y, R)	-
q_1	$(q_1, 0, R)$	(q_2, Y, L)	\rightarrow	(q_1, Y, R)	-
q_2	$(q_1, 0, L)$	\rightarrow	(q_0, X, R)	(q_2, Y, L)	\rightarrow
q_3	-	-	\rightarrow	(q_3, Y, R)	(q_1, B, R)
q_4	-	-	-	\rightarrow	-

UNIT - 5 WORKSHEET - 1

1. Assume H = Head
T = Tail

List R = (T, H, HTH, TTH)

List S = (TH, TH, HT, T)

Now we have to find out a sequence that strings formed by R and S are identical such a sequence is 1, 2, 1, 3, 3, 4.

Hence from the R and S list

1 2 1 3 3 4 1 2 1 3 3 4 4
T H T HTH HTH TT TH TH TH HT HT +

OR

Take M = 5

Take the combination 1, 2, 1, 3, 3, 4

By concatenation strings in this series

$\omega_1 \omega_2 \omega_3 \omega_4 = n_1 n_2 n_3 n_4 n_5$

T H T HTH HTH T T = TH + H HHT HTT

Instance of PCP = 121334

4. Counting sort takes $O(n+k)$ time and $O(n+k)$ space, where n is the number of items were sorting and k is the number of possible values. We iterate through the input items twice - one to populate counts and once to fill in the output array. Both iterate are $O(n)$ time. Additionally we iterate through counts once to fill in next Index, which is $O(k)$ time. The algorithm allocates 3 additional array: one for count, one for next Index and one for the output. The first two are $O(k)$ space and the final one is $(O(n))$ space

HACKERRANK REGEX COMPLETION OF 20 QUESTIONS--

HackerRank Prepare > Regex > Applications > Find A Sub-Word

Problem

We define a word character to be any of the following:

- An English alphabetic letter (i.e., a~z and A~Z).
- A decimal digit (i.e., 0~9).
- An underscore (i.e., _ which corresponds to ASCII value 95).

We define a word to be a contiguous sequence of one or more word characters that is preceded and succeeded by one or more occurrences of non-word-characters or line terminators. For example, in the string I love-cheese_?, the words are I, love, and cheese..

We define a sub-word as follows:

- A sequence of word characters (i.e., English alphabetic letters, digits, and/or underscores) that occur in the same exact order (i.e., as a contiguous sequence) inside another word.
- It is preceded and succeeded by word characters only.

Given n sentences consisting of one or more words separated by non-word characters, process q queries where each query consists of a single string s . To process each query, count the number of occurrences of s as a sub-word in all n sentences, then print the number of occurrences on a new line.

Input Format

The first line contains an integer, n , denoting the number of sentences. Each of the n subsequent lines contains a sentence consisting of words separated by non-word characters. The next line contains an integer, q , denoting the number of queries. Each line i of the q subsequent lines contains a string, s_i , to check.

Constraints

- $1 \leq n \leq 100$
- $1 \leq q \leq 10$

Output Format

For each query string, s_i , print the total number of times it occurs as a sub-word within all words in all n sentences.

Sample Input

```
1
existing pessimist optimist this is
1
is
```

Sample Output

```
3
```

HackerRank Prepare > Regex > Applications > Detect the Email Addresses

Problem

You will be provided with a block of text, spanning not more than hundred lines. Your task is to find the unique e-mail addresses present in the text. You could use Regular Expressions to simplify your task. And remember that the "@" sign can be used for a variety of purposes! Requirements are simplified versus real world. There can be a number of strings separated by dots before and after the "@" symbol. Strings will be made up of characters in the ranges a-Z, 0-9, underscore.

Input Format

The first line contains an integer N ($N \leq 100$), which is the number of lines present in the text fragment which follows.

From the second line, begins the text fragment (of N lines) in which you need to search for e-mail addresses.

Output Format

All the unique e-mail addresses detected by you, in one line, in lexicographical order, with a semi-colon as the delimiter.

Sample Input

```
19
HackerRank is more than just a company
    We are a tight group of hackers, bootstrappers, entrepreneurial thinkers and innovators
Hypothesis: Every hacker loves a particular type of challenge presented in a certain set of
Available Positions
Product Hacker product@hackerrank.com
Challenge Hacker challenge@hackerrank.com
product Evangelist
Product Designer
Content Creator
ACM World Finals Hacker
Backend C++ Hacker
Backend C++ Hacker
Mail us at hackers@hackerrank.com to chat more. Or you can write to us at interviewstreet@HACKERRANK PERKS
Working for a startup is hard work, but there are plenty of benefits of working for a small
[Image] Park: Get tools for the jobAll the Right ToolsWe know that everyone's perfect works
CULTURE
The culture of a startup is reflective of the founders' DNA. Larry Page & Sergey Brin were i
The adjective to describe the environment/founders here is relentless hardworkers. It migh
You'll have to embrace randomness and chaos. There's some level of discipline (eg: daily sc
Sample Output
```

HackerRank Prepare > Regex > Applications > Detect the Email Addresses

Problem

You will be provided with a block of text, spanning not more than hundred lines. Your task is to find the unique e-mail addresses present in the text. You could use Regular Expressions to simplify your task. And remember that the "@" sign can be used for a variety of purposes! Requirements are simplified versus real world. There can be a number of strings separated by dots before and after the "@" symbol. Strings will be made up of characters in the ranges a-Z, 0-9, underscore.

Input Format

The first line contains an integer N ($N \leq 100$), which is the number of lines present in the text fragment which follows.

From the second line, begins the text fragment (of N lines) in which you need to search for e-mail addresses.

Output Format

All the unique e-mail addresses detected by you, in one line, in lexicographical order, with a semi-colon as the delimiter.

Sample Input

```
19
HackerRank is more than just a company
    We are a tight group of hackers, bootstrappers, entrepreneurial thinkers and innovators
Hypothesis: Every hacker loves a particular type of challenge presented in a certain set of
Available Positions
Product Hacker product@hackerrank.com
Challenge Hacker challenge@hackerrank.com
product Evangelist
Product Designer
Content Creator
ACM World Finals Hacker
Backend C++ Hacker
Backend C++ Hacker
Mail us at hackers@hackerrank.com to chat more. Or you can write to us at interviewstreet@HACKERRANK PERKS
Working for a startup is hard work, but there are plenty of benefits of working for a small
[Image] Park: Get tools for the jobAll the Right ToolsWe know that everyone's perfect works
CULTURE
The culture of a startup is reflective of the founders' DNA. Larry Page & Sergey Brin were i
The adjective to describe the environment/founders here is relentless hardworkers. It migh
You'll have to embrace randomness and chaos. There's some level of discipline (eg: daily sc
Sample Output
```

Congratulations

You solved this challenge. Would you like to challenge your friends? [Facebook](#) [Twitter](#) [LinkedIn](#)

Test case 0 Compiler Message Success

Test case 1 Input (stdin) Download

```
1
existing pessimist optimist this is
```

Test case 2 Input (stdin) Download

```
1
is
```

Test case 3 Input (stdin) Download

```
1
existing pessimist optimist this is
```

Test case 4 Input (stdin) Download

```
1
is
```

Test case 5 Input (stdin) Download

```
1
3
```

Test case 6 Input (stdin) Download

```
1
3
```

HackerRank Prepare > Regex > Applications > Detect the Email Addresses

Problem

You will be provided with a block of text, spanning not more than hundred lines. Your task is to find the unique e-mail addresses present in the text. You could use Regular Expressions to simplify your task. And remember that the "@" sign can be used for a variety of purposes! Requirements are simplified versus real world. There can be a number of strings separated by dots before and after the "@" symbol. Strings will be made up of characters in the ranges a-Z, 0-9, underscore.

Input Format

The first line contains an integer N ($N \leq 100$), which is the number of lines present in the text fragment which follows.

From the second line, begins the text fragment (of N lines) in which you need to search for e-mail addresses.

Output Format

All the unique e-mail addresses detected by you, in one line, in lexicographical order, with a semi-colon as the delimiter.

Sample Input

```
36
Finally this phone is testimony to our quest and ever open ears for hearing from our
customers since 1921. We look forward to hearing from you today.
All India National Toll Free Number: 180 0425 0426
Working Hours: 10:00 am to 6:00 pm (Monday ~ Friday),
10:00 am to 2:00 pm (Saturday). To report ATM Card Lost, Kindly contact: +91 (44) 2622 3106
/ 2622 3109.
THB Customer Care: +91 9842 461 461
For all your queries, on any of our services in any branch in India, you can now SMS
666Hello666 or call +91 9842 461 461. THB666 Customer Care team is at your service
```

HackerRank Prepare > Regex > Applications > Detect the Email Addresses

Problem

You will be provided with a block of text, spanning not more than hundred lines. Your task is to find the unique e-mail addresses present in the text. You could use Regular Expressions to simplify your task. And remember that the "@" sign can be used for a variety of purposes! Requirements are simplified versus real world. There can be a number of strings separated by dots before and after the "@" symbol. Strings will be made up of characters in the ranges a-Z, 0-9, underscore.

Input Format

The first line contains an integer N ($N \leq 100$), which is the number of lines present in the text fragment which follows.

From the second line, begins the text fragment (of N lines) in which you need to search for e-mail addresses.

Output Format

All the unique e-mail addresses detected by you, in one line, in lexicographical order, with a semi-colon as the delimiter.

Sample Input

```
36
Finally this phone is testimony to our quest and ever open ears for hearing from our
customers since 1921. We look forward to hearing from you today.
All India National Toll Free Number: 180 0425 0426
Working Hours: 10:00 am to 6:00 pm (Monday ~ Friday),
10:00 am to 2:00 pm (Saturday). To report ATM Card Lost, Kindly contact: +91 (44) 2622 3106
/ 2622 3109.
THB Customer Care: +91 9842 461 461
For all your queries, on any of our services in any branch in India, you can now SMS
666Hello666 or call +91 9842 461 461. THB666 Customer Care team is at your service
```

Congratulations

You solved this challenge. Would you like to challenge your friends? [Facebook](#) [Twitter](#) [LinkedIn](#)

Test case 0 Compiler Message Success

Test case 1 Input (stdin) Download

```
HACKERRANK
```

Test case 2 Input (stdin) Download

```
36
```

Test case 3 Input (stdin) Download

```
Finally this phone is testimony to our quest and ever open ears for hearing from our
customers since 1921. We look forward to hearing from you today.
```

Test case 4 Input (stdin) Download

```
All India National Toll Free Number: 180 0425 0426
```

HackerRank Prepare > Regex > Applications — The British and American Style of Spelling

Exit Full Screen View

Problem

American English and British English differ in several aspects which are reflected in their spelling. One difference frequently observed, is that words written in American English, which have a suffix **ze** often end in **se** in British English. Given the American-English spelling of a word which ends in **ze** your task is to find the total count of all its British and American variants in all the given sequences of words, i.e. you need to account for the cases where the word occurs as it is given to you (i.e. the version ending in -**ze**) and you also need to find the occurrences of its British-English counterparts (i.e. the version ending in -**se**).

Input Format

First line contains N, N lines follow each line contains a sequence of words (W) separated by a single space. Next line contains T, T testcases follow in a new line. Each line contains the **American English** spelling of a word (W)

Constraints

$1 \leq N \leq 10$
 Each line doesn't contain more than 10 words (W).
 Each character of W and W' is a lowercase alphabet.
 If C is the count of the number of characters of W or W', then
 $1 \leq C \leq 20$
 $1 \leq T \leq 10$
 W' ends with **ze** (US version of the word)

Output Format

Output T lines and in each line output the total number of American and British versions of (W) in all of N lines that contains a sequence of words.

Sample Input

```
2
hackerrank has such a good ui that it takes no time to familiarise its environment
to familiarize oneself with ui of hackerrank is easy
1
familiarize
```

Sample Output

```
2
```

Explanation

In the given 2 lines, we find familiarize and familiarise once each. So, the total count is 2.

Viewing Submissions

You can view others' submissions if you solve this challenge. Navigate to the challenge leaderboard.

Congratulations

You solved this challenge. Would you like to challenge your friends? [Facebook](#) [Twitter](#) [LinkedIn](#)

Test case 0 Compiler Message Success

Test case 1 Input (stdin) Download

```
1 2
```

Test case 2 Input (stdin) Download

```
2 hackerrank ui is easy to familiarise with
3 to familiarize oneself with ui of hackerrank is easy
```

Test case 3 Input (stdin) Download

```
4 1
```

Test case 4 Input (stdin) Download

```
5 familiarize
```

Test case 5 Expected Output Download

Test case 6 Expected Output Download

STATUS

- Solved
- Unsolved

DIFICULTY

- Easy
- Medium
- Hard

SUBDOMAINS

- Introduction
- Character Class
- Repetitions
- Grouping and Capturing
- Backreferences
- Assertions
- Applications

Utopian Identification Number Easy, Max Score: 15, Success Rate: 97.02%

Valid PAN format Easy, Max Score: 15, Success Rate: 93.41%

Find HackerRank Easy, Max Score: 15, Success Rate: 97.10%

Saying Hi Easy, Max Score: 15, Success Rate: 96.82%

HackerRank Language Easy, Max Score: 15, Success Rate: 92.66%

Building a Smart IDE: Programming Language Detection Medium, Max Score: 30, Success Rate: 81.56%

Split the Phone Numbers Easy, Max Score: 15, Success Rate: 98.05%

The British and American Style of Spelling Easy, Max Score: 15, Success Rate: 94.25%

[Blog](#) | [Scoring](#) | [Environment](#) | [FAQ](#) | [About Us](#) | [Support](#) | [Careers](#) | [Terms Of Service](#) | [Privacy Policy](#)

HackerRank Prepare > Regex > Applications – Detect the Domain Name

Problem

You will be provided with a chunk of HTML markup. Your task is to identify the unique domain names from the links or URLs which are present in the markup fragment.

For example, if the link <http://www.hackerrank.com/contest> is present in the markup, you should detect the domain: `hackerrank.com`. In case there are second level or higher level domains present in the markup, all of them need to be treated as unique. For instance if the links <http://www.xyz.com/news>, <https://abc.xyz.com/jobs>, <http://abcd.xyz.com/jobs> are present in the markup then `lyz.com`, `abc.xyz.com`, `abcd.xyz.com` should all be identified as unique domains present in the markup. Prefixes like "www." and "ww2.", if present, should be scrubbed out from the domain name.

Input Format

An Integer `N`. This is equal to the number of lines in the HTML Fragment which follows. A chunk of HTML Markup with embedded links, the length of which is `N` lines.

Output Format

One line, containing the list of detected domains, separated by semi-colons, in lexicographical order. Do not leave any leading or trailing spaces either at the ends of the line, or before and after the individual domain names.

Sample Input

```
<div class="refList" style="list-style-type: decimal;">

- Train \(noun\)
- Hydrogen trains
- Vehicle Projects Inc. Fuel cell
- Overview Of the existingumba

```

Sample Output

```
askoxford.com;bnsf.com;hydrogencarsnow.com;mrvc.indianrail.gov.in;web.archive.org
```

HackerRank Prepare > Regex > Applications – HackerRank Tweets

Problem

Increasing popularity of hackerrank can be seen in tweets like

- I love #hackerrank
- I just scored 27 points in the Picking Cards challenge on #HackerRank
- I just signed up for summer cup @hackerrank

Given a set of most popular tweets, your task is to find out how many of those tweets has the string `hackerrank` in it.

Input Format

First line is an Integer `N`. `N` lines follow. Each line is a valid tweet.

Constraints

$1 \leq N \leq 10$

Each character of the tweet is a valid ASCII character.

Output format

Print the total number of tweets that has `hackerrank` (case insensitive) in it.

Sample Input

```
4
I love #hackerrank
I just scored 27 points in the Picking Cards challenge on #HackerRank
I just signed up for summer cup @hackerrank
interesting talk by har1, co-founder of hackerrank
```

Sample Output

```
4
```

Explanation

All the 4 lines have the string `hackerrank` in them and thus 4.

Viewing Submissions

You can view others' submissions if you solve this challenge. Navigate to the challenge leaderboard.

Congratulations

You solved this challenge. Would you like to challenge your friends? [Facebook](#) [Twitter](#) [LinkedIn](#)

[Next Challenge](#)

Test case 0 Compiler Message Success

Test case 1

Test case 2

Test case 3

Test case 4

Input (stdin) Download

```
1027
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" lang="en">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8" />
<meta name="format-detection" content="telephone=no" />
<title>Rediff.com - India, Business, Stock, Sports, Cricket, Entertainment, Bollywood, Music, Video and Breaking news, Rediffmail NG, Shopping</title>
```

Test case 0 Compiler Message Success

Test case 1

Test case 2

Test case 3

Test case 4

Test case 5

Test case 6

Input (stdin) Download

```
1
4
```

Test case 0 Compiler Message Success

Test case 1

Test case 2

Test case 3

Test case 4

Test case 5

Test case 6

Expected Output Download

```
1
4
```

HackerRank Prepare > Regex > Applications - Detect HTML links.

Charlie has been given an assignment by his Professor to strip the links and the text name from the html pages. A html link is of the form,

```
<a href="http://www.hackerrank.com">HackerRank</a>
```

Where a is the tag and href is an attribute which holds the link charlie is interested in. The text name is HackerRank. Charlie notices that the text name can sometimes be hidden within multiple tags

```
<a href="http://www.hackerrank.com"><h1><b>HackerRank</b></h1></a>
```

Here, the text name is hidden inside the tags h1 and b.

Help Charlie in listing all the links and the text name of the links.

Input Format

The first line contains the number of lines in the fragment (N). This is followed by N lines from a valid HTML document or fragment.

Constraints

N < 100
Number of characters in the test fragments <= 10000 characters.
Characters will be restricted to ASCII. Fragments for the tests will be picked up from Wikipedia. Also, some tests might not have text or names on the links.

Output Format

If there are M links in the document, display each of them in a new line. The link and the text name must be separated by a "," (comma) with no spaces between them.
Strip out any extra spaces at the start and end position of both the link and the text name before printing.

```
link-1,text name-1
link-2,text name-2
link-3,text name-3
...
link-n,text name-M
```

Sample Input

```
Sample Input
```

Sample Output

```
Sample Output:1
```

Line: 11 Col: 49

Upload Code as File Test against custom input Run Code Submit Code

Test case 0	Compiler Message
Test case 1	Success
Test case 2	Input (stdin)
Test case 3	Download
Test case 4	Input (stdin)
Test case 5	Download
Test case 6	Input (stdin)

```
1 2
2 <p><a href="http://www.quackit.com/html/tutorial/html_links.cfm">Example Link</a></p>
3 <div class="more-info"><a href="http://www.quackit.com/html/examples
/html_links_examples.cfm">More Link Examples...</a></div>
```

Expected Output

```
1 http://www.quackit.com/html/tutorial/html_links.cfm,Example Link
2 http://www.quackit.com/html/examples/html_links_examples.cfm,More Link Examples...
```

HackerRank Prepare > Regex > Applications - Alien Username

In a galaxy far, far away, on a planet different from ours, each computer username uses the following format:

1. It must begin with either an underscore, _ (ASCII value 90), or a period, . (ASCII value 46).
2. The first character must be immediately followed by one or more digits in the range 0 through 9.
3. After some number of digits, there must be 0 or more English letters (uppercase and/or lowercase).
4. It may be terminated with an optional _ (ASCII value 95). Note that if it's not terminated with an underscore, then there should be no characters after the sequence of 0 or more English letters.

Given n strings, determine which ones are valid alien usernames. If a string is a valid alien username, print VALID on a new line; otherwise, print INVALID.

Input Format

The first line contains a single integer, n, denoting the number of usernames.

Each line i of the n subsequent lines contains a string denoting an alien username to validate.

Constraints

* 1 ≤ n ≤ 100

Output Format

Iterate through each of the n strings in order and determine whether or not each string is a valid alien username. If a username is a valid alien username, print VALID on a new line; otherwise, print INVALID.

Sample Input

```
3
_0898989811abcd_
.abce
_09e9e9e9abcD0
```

Sample Output

```
VALID
INVALID
INVALID
```

Explanation

We validate the following three usernames:

1. _0898989811abcd_ is valid as it satisfies the requirements specified above. Thus, we print VALID.
2. .abce is invalid as the beginning . is not followed by one or more digits. Thus, we print INVALID.
3. _09e9e9e9abcD0 is invalid as the sequence of English alphabetic letters is immediately followed by a number. _09e9e9e9abcD0

Line: 12 Col: 29

Upload Code as File Test against custom input Run Code Submit Code

Congratulations

You solved this challenge. Would you like to challenge your friends? [Facebook](#) [Twitter](#) [LinkedIn](#) [Next Challenge](#)

Test case 0	Compiler Message
Test case 1	Success
Test case 2	Input (stdin)
Test case 3	Download
Test case 4	Input (stdin)
Test case 5	Download
Test case 6	Input (stdin)

```
1 3
2 _0898989811abcd_
3 _abce
4 _09e9e9e9abcD0
```

Expected Output

```
1 VALID
2 INVALID
```

HackerRank Prepare > Regex > Applications - Detect HTML Tags.

In this challenge, we're using regular expressions to detect the various tags used in an HTML document.

- Tags come in pairs. Some tag name, `t`, will have an opening tag, `<t>`, followed by some intermediate text, followed by a closing tag, `</t>`. The forward slash in a closing tag will always come before the tag name.
- The exception to this is self-closing tags, which consist of a single tag (not a pair) with a forward slash after the tag name: `<p/>`

Here are a few examples of tags:

- The `p` tag is for paragraphs: `<p>This is a paragraph</p>`
- There may be 1 or more spaces before or after a tag name: `<p>This is also a paragraph</p>`
- A void or empty tag involves an opening and closing tag with no intermediate characters: `<p></p>`

Some tags can also have attributes, such as the `a` tag, which is used to add a hyperlink to another document: `Google`

In the above case, `a` is the tag name and `href` is an attribute having the value `http://www.google.com`.

Task

Given N lines of HTML, find the tag names (ignore any attributes) and print them as a single line of lexicographically ordered semicolon-separated values (e.g.: `tag1;tag2;tag3`).

Input Format

The first line contains an integer, N , the number of HTML fragments. Each of the N subsequent lines contains a fragment of an HTML document.

Constraints

- $1 \leq N \leq 100$
- Each fragment contains < 10000 ASCII characters.
- The fragments are chosen from Wikipedia, so analyzing and observing their markup structure may help.
- Leading and trailing spaces/indentation have been trimmed from the HTML fragments.

Output Format

Print a single line containing all of the unique tag names found in the input. Your output tags should be semicolon-separated and ordered lexicographically (i.e.: alphabetically). Do not print the same tag name more than once.

Sample Input

```
2
<p><a href="http://www.quackit.com/html/tutorial/html_links.cfm">Example Link</a></p>
<div class="more-info"><a href="http://www.quackit.com/html/examples/html_links_examples
/htm_links_examp...>More Link Examples...</a></div>
```

Sample Output

```
2;div;p;a
```

Exit Full Screen View

Line: 15 Col: 32

Upload Code as File Test against custom input Run Code Submit Code

Test case 0	Compiler Message
Success	
Test case 1	Input (stdin)
	2
Test case 2	Output
	1 2
Test case 3	Compiler Message
	Success
Test case 4	Input (stdin)
	<p>Example Link</p>
Test case 5	Output
	1 <p>Example Link</p>
Test case 6	Compiler Message
	Success
	Input (stdin)
	<div class="more-info">More Link Examples...</div>
	Output
	1 <div>

Download Download

Exit Full Screen View

Line: 12 Col: 39

Upload Code as File Test against custom input Run Code Submit Code

Congratulations

You solved this challenge. Would you like to challenge your friends? [Facebook](#) [Twitter](#) [LinkedIn](#)

Next Challenge

Test case 0	Compiler Message
Success	
Test case 1	Input (stdin)
	2
Test case 2	Output
	1 2
Test case 3	Compiler Message
	Success
Test case 4	Input (stdin)
	abc012333ABCDEEEEEE
Test case 5	Output
	1 VALID
Test case 6	Compiler Message
	Success
	Input (stdin)
	0123AB
	Output
	1 INVALID

Download Download

HackerRank Prepare > Regex > Introduction - Matching Specific String

Problem

Regular expression (or RegEx)
A regular expression is a sequence of characters that define a search pattern. It is mainly used for string pattern matching.

In the above image, a **Regex Pattern** is matched with the **Test String**.

Regular expressions are extremely useful in extracting information from text such as: code, log files, spreadsheets, documents, etc.

We can match a specific string X in a test string S by making our regex pattern X . This is one of the simplest patterns. However, in the coming challenges, we'll see how well we can match more complex patterns and learn about their syntax.

Task
You have a test string S . Your task is to match the string `hackerrank`. This is case sensitive.

Note
This is a regex only challenge. You are not required to write code.
You only have to fill in the regex pattern in the blank (_____).

Congratulations
You solved this challenge. Would you like to challenge your friends? [Facebook](#) [Twitter](#) [LinkedIn](#)

Test case 0 Compiler Message: Success
Input (stdin):
1 The hackerrank team is on a mission to flatten the world by restructuring the DNA of every company on the planet. We rank programmers based on their coding skills, helping companies source great programmers and reduce the time to hire. As a result, we are revolutionizing the way companies discover and evaluate talented engineers. The hackerrank platform is the destination for the best engineers and companies to find top engineers.

Expected Output: Download
1 Number of matches : 2

HackerRank Prepare > Regex > Applications - Build a Stack Exchange Scraper

Problem

Stack Exchange is an information power-house, which contains libraries of crowdsourced problems (with answers) across a large number of topics which are as diverse as electronics, cooking, programming, etc.

We are greatly interested in crawling and scraping as many questions, as we can, from stack-exchange. This is an example of a question library page from stack-exchange.

Your task will be, to scrape the questions from each library page, in the order in which they are listed. You will be provided with the markup of question listing pages, from which you need to detect:

(1) identifier (2) Question text (which is on the hyperlink to the question) (3) How long ago the question was asked.

The Markup in the Test Cases will be similar to the sample fragment shown below. Please note, that since this markup is real markup from the website, it is likely to contain some stray control and escape characters, unexpected whitespaces and newlines.

Sample Markup Fragment

```
<div class="question-summary" id="question-summary-80467">
  <div class="statscontainer">
    <div class="statsarrow"></div>
    <div class="stats">
      <div class="vote">
        <div class="votes">
          <span class="vote-count-post "><strong>2</strong></span>
          <div class="viewcount">votes</div>
        </div>
      </div>
      <div class="status answered">
        <strong>1</strong> answer
      </div>
    </div>
  </div>
  <div class="views " title="68 views">
    68 views
  </div>
  <div class="summary">
    <a href="/questions/80467/about-power-supply-of-operational-amplifier" class="excerpt">
      I am constructing an operational amplifier as shown in the following fi...
    </a>
    <div class="tags t-op-amp">
      <a href="/questions/tagged/op-amp" class="post-tag" title="show question...
    </div>
  </div>
</div>
```

build ^ v Highlight BL Match Case Match Diacritics Whole Words 2 of 3 matches

Congratulations
You solved this challenge. Would you like to challenge your friends? [Facebook](#) [Twitter](#) [LinkedIn](#)

Test case 0 Compiler Message: Success
Input (stdin):
1 <!DOCTYPE html>
2 <html>
3 <head>
4 <title>Newest Questions - Page 2 - Electrical Engineering Stack Exchange</title>
5 <link rel="shortcut icon" href="//cdn.sstatic.net/electronics/img/favicon.ico" type="image/x-icon" />
6 <link rel="apple-touch-icon" href="//cdn.sstatic.net/electronics/img/apple-touch-icon.png" type="image/x-icon" />
7 <link rel="search" type="application/opensearchdescription+xml" title="Electrical Engin...

HackerRank Prepare > Regex > Applications - Saying Hi

Given a sentence, s , write a RegEx to match the following criteria:

- The first character must be the letter H or h .
- The second character must be the letter I or i .
- The third character must be a single space (i.e., '\s').
- The fourth character must not be the letter D or d .

Given n lines of sentences as input, print each sentence matching your RegEx on a new line.

Input Format

The first line contains an integer, n , denoting the number of lines of sentences. Each of the n subsequent lines contains some sentence s you must match.

Constraints

- $1 \leq n \leq 10$
- Each sentence, s , contains 1 to 10 words.
- Each word/token in a sentence is comprised only of upper and lowercase English letters.

Output Format

Find each sentence, s , satisfying the RegEx criteria mentioned above, and print it on a new line.

Sample Input

```
5
Hi Alex how are you doing
hi Dave how are you doing
Good by Alex
Hidden agenda
Alex greeted Martha by saying Hi Martha
```

Sample Output

```
Hi Alex how are you doing
```

Explanation

The first sentence satisfies the RegEx criteria set forth in the Problem Statement (starts with the case-insensitive word *Hi*, followed by a space, followed by a letter that is not *d*), so we print the sentence on a new line.

The second sentence fails our RegEx criteria, as the second word/token starts with a *d* (so we print nothing).

The third sentence fails our RegEx criteria, as it doesn't start with an *h* (so we print nothing).

The fourth sentence fails our RegEx criteria, as the third character in the sentence is not a space (so we print nothing).

Line: 18 Col: 21

Upload Code as File Test against custom input Run Code Submit Code

Congratulations

You solved this challenge. Would you like to challenge your friends? [Facebook](#) [Twitter](#) [LinkedIn](#)

Test case 0 Success

Input (stdin)

```
5
Hi Alex how are you doing
hi Dave how are you doing
Good by Alex
Hidden agenda
Alex greeted Martha by saying Hi Martha
```

Download

Test case 1

Test case 2

Test case 3

Test case 4

Test case 5

Test case 6

Expected Output

```
Hi Alex how are you doing
```

Download

Next Challenge

HackerRank Prepare > Regex > Applications - Valid PAN Format

The equivalent of SSN in India is a PAN number, which is unique to each of its citizens. In any of the country's official documents, the PAN number is listed as follows

```
<char><char><char><char><char><digit><digit><digit><digit><char>
```

Your task is to figure out if the PAN number is valid or not. A valid PAN number will have all its letters in uppercase and digits in the same order as listed above.

Input Format

First line contains N . N lines follow, each having a PAN number.

Constraints

- $1 \leq N \leq 10$
- Each char is an uppercase letter, i.e., $char \in ['A' / 'Z']$.
- Each digit lies between 0 and 9, i.e., $digit \in [0, 9]$.
- The length of the PAN number is always 10, i.e., $length(PAN) = 10$.
- Every character in PAN is either char or digit satisfying the above constraints.

Output Format

For every PAN number listed, print YES if it is valid and NO if it isn't.

Sample Input

```
3
ABCD1234Y
ABAB12345Y
avCDS1234Y
```

Sample Output

```
YES
NO
NO
```

The first PAN number is valid since the first 5 characters are letters, the next 4 are digits and the last character is an alphabet. All letters in input is in uppercase.

The second PAN number is invalid as the fifth character is a digit as opposed to an letter.

The third PAN number contains lowercase characters. Hence invalid.

Viewing Submissions

You can view others' submissions if you solve this challenge. Navigate to the challenge leaderboard.

Line: 12 Col: 24

Upload Code as File Test against custom input Run Code Submit Code

Congratulations

You solved this challenge. Would you like to challenge your friends? [Facebook](#) [Twitter](#) [LinkedIn](#)

Test case 0 Compiler Message Success

Input (stdin)

```
3
ABCD1234Y
ABAB12345Y
avCDS1234Y
```

Download

Test case 1

Test case 2

Test case 3

Test case 4

Test case 5

Test case 6

Expected Output

```
YES
NO
```

Download

Next Challenge

HackerRank Prepare > Regex > Introduction - Matching Digits & Non-Digit Characters

Problem Submissions Leaderboard Discussions Editorial

\d
The expression \d matches any digit [0 - 9].

In the above image, a Regex Pattern is matched with the Test String.

\D
The expression \D matches any character that is not a digit.

HackerRank Prepare > Regex > Applications - Find a Word

Problem Submissions Leaderboard Discussions Editorial

We define a word as a non-empty maximum sequence of characters that can contain only lowercase letters, uppercase letters, digits and underscores ("ASCII value 95). Maximum sequence means that the word has to be immediately preceded by a character not allowed to occur in a word or by the left boundary of the sentence, and it has to be immediately followed by a character not allowed to occur in a word or by the right boundary of the sentence.

Given N sentences and T words, for each of these words, find the number of its occurrences in all the N sentences.

Input Format
In the first line there is a single integer N . Each of the next N lines contains a single sentence. After that, in the next line, there is a single integer T denoting the number of words. In the i -th of the next T lines, there is a single word denoting the i -th word for which, you have to find the number of its occurrences in the sentences.

Constraints
 $1 \leq N \leq 100$
 $1 \leq T \leq 10$

Output format
For every word, print the number of occurrences of the word in all the N sentences listed.

Sample Input

```
1
foo bar (foo) bar foo-bar foo_bar foo'bar bar-foo bar, foo,
1
foo
```

Sample Output

```
6
```

Explanation

- foo is the first word
- (foo) is preceded by "t" and followed by ")", so it's the second word.
- foo-bar is considered as two words and 'foo' is the first of them. It is preceded by a space and followed by a hyphen '-'
- bar-foo also contains foo for the same reason mentioned above
- foo_bar is a single simple word and hence foo in it is not counted
- foo'bar is considered as two words and 'foo' is the first of them. It is preceded by a space and followed by a apostrophe '''
- foo, as it is preceded by a space and followed by a dot'.'

Exit Full Screen View

Line: 1 Col: 65

Upload Code as File Test against custom input Run Code Submit Code

Test case 0 Compiler Message Success

Test case 1 Input (stdin) Download

```
1 06-11-2015
```

Test case 2 Expected Output Download

```
1 true
```

Test case 3

Test case 4

Test case 5

Line: 16 Col: 21

Upload Code as File Test against custom input Run Code Submit Code

Congratulations
You solved this challenge. Would you like to challenge your friends? [f](#) [t](#) [in](#)

Test case 0 Compiler Message Success

Test case 1 Input (stdin) Download

```
1
2 foo bar (foo) bar foo-bar foo_bar foo'bar bar-foo bar, foo,
3 1
4 foo
```

Test case 2

Test case 3

Test case 4

Test case 5

Test case 6

Expected Output Download

```
1 6
```

HackerRank Prepare > Regex > Applications - Split the Phone Numbers

There is a list of phone numbers which needs the attention of a text processing expert. As an expert in regular expressions, you are being roped in for the task. A phone number directory can reveal a lot such as country codes and local area codes. The only constraint is that one should know how to process it correctly.

A Phone number is of the following format

```
[Country code]-[Local Area Code]-[Number]
```

There might either be a '-' (ascii value 45), or a '' (space, ascii value 32) between the segments

Where the country and local area codes can have 1-3 numerals each and the number section can have 4-10 numerals each.

And so, if we tried to apply a regular expression with groups on this phone number: 1-425-9854706

We'd get:

```
Group 1 = 1
Group 2 = 425
Group 3 = 9854706
```

You will be provided a list of N phone numbers which conform to the pattern described above. Your task is to split it into the country code, local area code and the number.

Input Format

N, where N is the number of tests.

This will be followed by N lines containing N phone numbers in the format specified above.

Constraints

$1 \leq N \leq 20$

There might either be a hyphen, or a space between the segments

The country and local area codes can have 1-3 numerals each and the number section can have 4-10 numerals each.

Output Format

Your output will contain N lines.

CountryCode-[Country Code].LocalAreaCode-[Local Area Code].Number-[Number]

Recommended Technique

This problem can be solved in many ways, however, try to solve it using regular expressions and groups in order to gain a hands-on practice of the concepts involved.

Sample Input

```
2
1-425-9854706
1-425-9854706
```

Line: 18 Col: 94

Upload Code as File Test against custom input Run Code Submit Code

Congratulations

You solved this challenge. Would you like to challenge your friends? [Facebook](#) [Twitter](#) [LinkedIn](#)

Test case 0 Success

Input (stdin)

```
2
1-425-9854706
1-425-9854706
```

Download

Test case 1 Success

Input (stdin)

```
1-425-9854706
```

Download

Test case 2 Success

Input (stdin)

```
1-425-9854706
```

Download

Test case 3 Success

Input (stdin)

```
1-425-9854706
```

Download

Test case 4 Success

Input (stdin)

```
1-425-9854706
```

Download

Test case 5 Success

Input (stdin)

```
1-425-9854706
```

Download

Test case 6 Success

Input (stdin)

```
1-425-9854706
```

Download

HackerRank Prepare > Regex > Applications - HackerRank Language

Every submission at HackerRank has a field called language which indicates the programming language which a hacker used to code his solution.

CCCPPYTHONPERLPHPRUBYCSHARPJHASKELLCLOJUREBASHSCALAERLANGCLISPLUA BRAINFUCKJAVASCRIPTGODOCAMLRPASCALSBCLDARTGROOVYOBJECTIVEC

Sometimes, error-prone API requests can have an invalid language field. Could you find out if a given submission has a valid language field or not?

Input Format

First line contains N. N API requests follow, each in a newline. Each request has an integer api_id and a string language which are the request parameters placed by the an API request.

Constraints

```
1 <= N <= 100
10^4 <= api_id < 10^6
```

a valid language is any of the languages listed above (case sensitive):

Output Format

For every api request given in input, print "VALID" if it has a valid language string in it or print "INVALID" otherwise.

Sample Input

```
3
11011 LUA
11022 BRAINFUCK
11044 X
```

Sample Output

```
VALID
VALID
INVALID
```

Explanation

LUA and BRAINFUCK are valid languages as listed above. As X is doesn't appear in the list, it is an invalid request.

Line: 13 Col: 29

Upload Code as File Test against custom input Run Code Submit Code

Congratulations

You solved this challenge. Would you like to challenge your friends? [Facebook](#) [Twitter](#) [LinkedIn](#)

Test case 0 Success

Input (stdin)

```
3
11011 C
11022 CPP
11044 X
```

Download

Test case 1 Success

Input (stdin)

```
3
11011 C
11022 CPP
11044 X
```

Download

Test case 2 Success

Input (stdin)

```
3
11011 C
11022 CPP
11044 X
```

Download

Test case 3 Success

Input (stdin)

```
3
11011 C
11022 CPP
11044 X
```

Download

Test case 4 Success

Input (stdin)

```
3
11011 C
11022 CPP
11044 X
```

Download

Test case 5 Success

Input (stdin)

```
3
11011 C
11022 CPP
11044 X
```

Download

Test case 6 Success

Input (stdin)

```
3
11011 C
11022 CPP
11044 X
```

Download

Expected Output

```
1 VALID
2 VALID
3 INVALID
```

HackerRank NEW PREPARE CERTIFY COMPETE

Search | k6219

Prepare > Regex

Regex

Points: 270 Rank: 21409

Problem	Description	Status	Difficulty	Subdomains
Matching Specific String	Easy, Max Score: 5, Success Rate: 96.24%	Solved	Easy	Introduction
Matching Digits & Non-Digit Characters	Easy, Max Score: 5, Success Rate: 97.34%	Solved	Medium	Character Class
Detect HTML links	Medium, Max Score: 10, Success Rate: 72.89%	Solved	Hard	Repetitions
Detect HTML Tags	Easy, Max Score: 10, Success Rate: 93.17%	Solved		Grouping and Capturing
Find A Sub-Word	Easy, Max Score: 10, Success Rate: 90.28%	Solved		Backreferences
Alien Username	Easy, Max Score: 10, Success Rate: 95.29%	Solved		Assertions
IP Address Validation	Easy, Max Score: 10, Success Rate: 91.32%	Solved		Applications
Find a Word	Medium, Max Score: 15, Success Rate: 87.68%	Solved		

Problem

At HackerRank, we always want to find out how popular we are getting every day and have scraped conversations from popular sites. Each conversation fits in 1 line and there are N such conversations. Each conversation has at most 1 word that says `hackerrank` (all in lowercase). We would like you to help us figure out whether a conversation:

- Starts with `hackerrank`
- Ends with `hackerrank`
- Starts and ends with `hackerrank`

Input Format

First line of the input contains an integer, N. Then N lines follow.

From the second line onwards, each line contains a set of W words separated by a single space

Constraints

- $1 \leq N \leq 10$
- $1 \leq W \leq 100$
- All the characters in W are lowercase-alphabet characters.
- If C is the count of the characters in W, then $1 \leq C \leq 20$

Output Format

For every line,

- Print 1 if the conversation starts with `hackerrank`
- Print 2 if the conversation ends with `hackerrank`
- Print 0 if the conversation starts and ends with `hackerrank`
- Print -1 if none of the above.

Sample Input

```
4
i love hackerrank
hackerrank is an awesome place for programmers
hackerrank
i think hackerrank is a great place to hangout
```

Sample Output

```
2
1
0
-1
```

Explanation

The first conversation ends with `hackerrank` and hence 2

Code Editor

Upload Code as File Test against custom input Run Code Submit Code

Congratulations

You solved this challenge. Would you like to challenge your friends? [Facebook](#) [Twitter](#) [LinkedIn](#)

Test case 0

Input (stdin)

```
4
i love hackerrank
hackerrank is an awesome place for programmers
hackerrank
i think hackerrank is a great place to hangout
```

Expected Output

```
2
```

HackerRank Prepare > Regex > Applications - IP Address Validation

Problem

You will be provided with N lines of what are possibly IP addresses. You need to detect if the text contained in each of the lines represents an (a)IPv4 address (b)IPv6 address or (c)None of these.

IPv4 was the first publicly used Internet Protocol which used 4 byte addresses which permitted for 2^{32} addresses. The typical format of an IPv4 address is A.B.C.D where A, B, C and D are integers lying between 0 and 255 (both inclusive).

IPv6, with 128 bits was developed to permit the expansion of the address space. To quote from the linked article, The 128 bits of an IPv6 address are represented in 8 groups of 16 bits each. Each group is written as 4 hexadecimal digits and the groups are separated by colons (:). The address 2001:0db8:0000:0000:0000:ff00:0042:8329 is an example of this representation. Consecutive sections of zeros will be left as they are.

An IPv6 value such as "...0...." or "...5...." is address-wise identical to "...0000...." or "...0005....". Leading zeros may be omitted in writing the address.

Input Format
An Integer N such that $N \leq 50$. This is followed by N lines such that each the text in each line is either an IPv4 address or an IPv6 address, or a chunk of text which does not equal either of these. There will be no extra text or whitespace leading or trailing the IP address in a line (if it is an IP address). The number of characters in each line will not exceed 500.

Output Format
N lines.
The i^{th} output line should equal (a)IPv4 or (b)IPv6 or (c)Neither depending on what you detected the i^{th} input line to be.

Sample Input

```
3
This line has junk text.
1.21.18.19.20
2001:0db8:0000:0000:0000:ff00:0042:8329
```

Sample Output

```
Neither
IPv4
IPv6
```

HackerRank Prepare > Regex > Applications - Building a Smart IDE: Programming Language Detection

Problem

We are trying to hack together a smart programming IDE. Help us build a feature which auto-detects the programming language, given the source code. There are only three languages which we are interested in "auto-detecting": Java, C and Python.

We will provide you with links to a few short or medium size programs for Java, C and Python. In case you aren't familiar with some of these languages, these samples will help you make observations about the lexical structure and syntax of these programming languages. These sample programs are only for your manual inspection. You cannot read or access these sample programs from the code you submit.

After this, you will be provided with tests, where you are provided the source code for programs - or partial code snippets, but you do not know which language they are in. For each test, try to detect which language the source code is in.

You might benefit from using regular expressions in trying to detect the lexical structure and syntax of the programs provided.

Sample Programs to Understand the Lexical Structure of different Programming Languages

Sample Programs and Code Snippets in C
Sample Programs and Code Snippets in Java
Sample Programs and Code Snippets in Python

INPUT FORMAT
Source code of a program, or a code snippet, which might be in C, Java or Python.

OUTPUT FORMAT
Just one line containing the name of the Programming language which you have detected: This might be either C or Java or Python.

SAMPLE INPUT

```
import java.io.*;
public class SquareNum {
    public static void main(String args[]) throws IOException
    {
        System.out.println("This is a small Java Program!");
    }
}
```

SAMPLE OUTPUT

```
Java
```

HackerRank Prepare > Regex > Applications - Building a Smart IDE: Programming Language Detection

Problem

You solved this challenge. Would you like to challenge your friends? [Facebook](#) [Twitter](#) [LinkedIn](#)

Congratulations

Test case 0 Success

Compiler Message

```
Success
```

Test case 1 Success

Input (stdin)

```
# let us create a test string
```

Test case 2 Success

Input (stdin)

```
# let us create a test string
```

Test case 3 Success

Input (stdin)

```
# let us create a test string
```

Test case 4 Success

Input (stdin)

```
testString1 = "Hello World!"
```

Test case 5 Success

Input (stdin)

```
print "Original String: ", testString1
```

Test case 6 Success

Input (stdin)

```
# Print this string in lower case
```

Test case 7 Success

Input (stdin)

```
# Converting a string to lower case
```

Detect the Email Addresses Medium, Max Score: 15, Success Rate: 88.54%	★ Solved 	STATUS <input checked="" type="checkbox"/> Solved <input type="checkbox"/> Unsolved
Detect the Domain Name Medium, Max Score: 15, Success Rate: 89.62%	★ Solved 	DIFFICULTY <input type="checkbox"/> Easy <input type="checkbox"/> Medium <input type="checkbox"/> Hard
HackerRank Tweets Easy, Max Score: 15, Success Rate: 97.22%	★ Solved 	SUBDOMAINS <input type="checkbox"/> Introduction <input type="checkbox"/> Character Class <input type="checkbox"/> Repetitions <input type="checkbox"/> Grouping and Capturing <input type="checkbox"/> Backreferences <input type="checkbox"/> Assertions <input type="checkbox"/> Applications
Build a Stack Exchange Scraper Easy, Max Score: 15, Success Rate: 95.62%	★ Solved 	
Utopian Identification Number Easy, Max Score: 15, Success Rate: 97.02%	★ Solved 	
Valid PAN format Easy, Max Score: 15, Success Rate: 93.41%	★ Solved 	
Find HackerRank Easy, Max Score: 15, Success Rate: 97.10%	★ Solved 	
Saying Hi Easy, Max Score: 15, Success Rate: 96.82%	★ Solved 	
HackerRank Language Easy, Max Score: 15, Success Rate: 92.66%	★ Solved 	

