



UNIVERSIDAD NACIONAL EXPERIMENTAL DEL TÁCHIRA
VICERRECTORADO ACADÉMICO
DECANATO DE DOCENCIA
DEPARTAMENTO DE INFORMÁTICA
ASIGNATURA: Programación II

DIFERIDO PRIMER EXAMEN LAPSO 2015-1

VALOR: 55 Puntos.

APELLIDOS Y NOMBRES: _____

CEDULA DE IDENTIDAD: _____

SECCIÓN:

ESPECIALIDAD: INGENIERÍA EN INFORMÁTICA

FECHA: 11/05/2016

1. Solución de encriptación (35 puntos)

Los miembros del comité del maratón de programación, el cual diseñan los problemas para la competencia, usan un algoritmo de encriptación diseñado por ellos para poder enviar los problemas de una manera segura a través de internet. Para encriptar un texto cada letra es cambiada por otra letra (posiblemente ella misma), de tal manera que dos letras no sean cambiadas por la misma letra. El texto encriptado y el texto original están en mayúscula y espacios en blanco. Los espacios en blanco no son encriptados y estarán en la misma posición tanto en el texto original y el encriptado. Por ejemplo, la cadena GSRH RH GSV URIHG HZNVKOV es la encriptación de THI S IS THE FIRST SAMPLE, de acuerdo con la tabla de encriptación ($A \rightarrow Z$, $B \rightarrow Y$, $C \rightarrow X$, ..., $Z \rightarrow A$). En este caso el problema radica en que la tabla de encriptación se perdió, pero se tienen todas las palabras disponibles del diccionario. Su trabajo será ayudar a los miembros del comité a establecer la tabla de encriptación. Dado el diccionario con las palabras originales usadas en el texto, un texto encriptado, se desea encontrar la correcta tabla de encriptación tal que después de desencriptar el texto encriptado, el texto retorne al original. Todas las palabras pueden ser encontradas en el diccionario.

Entrada (encriptación.in)

La primera parte del archivo de entrada será el diccionario de palabras en inglés común para todos los casos de prueba. La primera línea es d ($1 \leq d \leq 50000$); que corresponde al número de palabras en el diccionario, seguidamente se tienen d líneas donde cada una tendrá las palabras del diccionario ordenadas en orden alfabético. Cada palabra tendrá como máximo 20 caracteres y como máximo se tendrán 350.000 palabras. La siguiente línea contiene un entero t ($1 \leq t \leq 10$), que será el número de casos de prueba, seguidamente cada caso de prueba. Cada caso de prueba es precedido por una línea en blanco y consistirá de una o varias líneas que formará el texto encriptado. Máximo hay 80 casos de prueba.

Salida (Pantalla)

La salida consistirá de t líneas, cada línea estará compuesta por una cadena de 26 caracteres el cual es la encriptación de la cadena ABCDEFGHIJKLMNOPQRSTUVWXYZ de acuerdo a la tabla de encriptación utilizada en el caso de prueba. Las letras deberán estar en mayúscula. Es posible que algunas letras no aparezcan en el texto encriptado, en este caso coloque un * en los espacios donde estas letras no aparezcan. Si el caso de prueba no tiene solución escriba como salida #No solution#. Si hay más de una solución posible para un caso de prueba escribir en la salida #More than one solution#.

Ejemplo de Entrada

```
14
BE
CHANGE
FIRST
IN
IS
MUST
SAMPLE
SEE
THE
THIS
TO
WISH
WORLD
YOU
4

GSRH RH GSV URIHG HZKOV

IZM BMVU      SP UGP
RGTANP IZM KFVG UZ VPP
FA UGP KZWCQ

XYZ ABCDEFG

XZY ABD
```

Ejemplo de Salida

```
Z***VU*SR**ON**K*IHG*****
TSRQP*NGF**CBAZ**WVUM*K*I*
#No solution#
#More than one solution#
```

2. Nombres de Servicio (30 puntos)

Se desea una lista de los nombres de servicios que están asociados con cada EntradaDeseada (tales como gen, DNAFragment, genome, etc). Lo que está disponible de proveedor del servicio es una lista de cadenas, cada una contiene el nombre de un servicio seguido por todos las EntradaDeseada que requiere.

Dada una lista de servicios, mostrar una lista en la que cada elemento contenga una EntradaDeseada seguido por los nombres de todos los servicios que usan esa EntradaDeseada.

Entrada (servicio.in)

La entrada comienza con un solo entero positivo en una línea, indicando el número de los casos siguientes, cada uno de ellos como se describe a posteriormente. Esta línea es seguida por una línea en blanco, y también hay una línea en blanco entre dos entradas consecutivas.

La entrada consiste en 1 o más líneas. Cada línea tiene las siguientes fichas: un nombre de servicio, un número entero k ($k \geq 0$) y un conjunto de k EntradaDeseada. Cada

ficha está separada por un único espacio. Cada ficha constará de 1 o más letras (AZ o az). La primera ficha en cada línea de entrada será distinta. Entre cada línea de servicios, las EntradaDeseada serán distintos.

Salida (Pantalla)

Para cada caso de prueba, la salida debe seguir la descripción que se describe posteriormente. Las salidas de dos casos consecutivos se separaran por una línea en blanco.

La salida consiste de una o más líneas. Cada línea contiene una EntradaDeseada (solo una vez), 5 caracteres “ ==> “ (espacio igual igual mayor espacio) y el conjunto de nombres de servicio separados por los dos caracteres “, ” (coma espacio). Si no hay EntradaDeseada, se imprime una línea que diga “No h ay EntradaDeseada”.

Las EntradaDeseada y servicios deben estar en orden alfabético. Todas las palabras son sensible a mayúsculas y minúsculas y por orden alfabético se refiere al orden ASCII donde por ejemplo ‘Z’ precede a ‘a’.

Ejemplo:

Entrada de Ejemplo	Salida de Ejemplo
3	Annotation ==> BLAST, X
	Genome ==> BLAST
BLAST 3 Genome Annotation Sensitivity	Sensitivity ==> BLAST
PING 0	
X 1 Annotation	No hay EntradaDeseada
PING 0	Annotation ==> X
	Genome ==> BLAST
BLAST 3 Genome annotation Sensitivity	Sensitivity ==> BLAST
PING 0	X ==> Apple
X 1 Annotation	ample ==> Apple
Apple 2 X ample ghghjfgghjfgthj	annotation ==> BLAST

Problema 3: Reloj

Se tiene un reloj estándar de 12 horas con una pantalla análoga, con una manecilla para la hora y una para los minutos. Cuantas veces la manecilla de los minutos pasa por la manecilla de la hora en un intervalo de tiempo? Un ejemplo se muestra a continuación:

12	50	1	2
3	8	3	20
2	45	11	0
11	0	3	20
1	2	12	50
3	20	3	8

Program 3 by team X		
Initial time	Final time	Passes
12:50	01:02	0
03:08	03:20	1
02:45	11:00	8
11:00	03:20	4
01:02	12:50	11
03:20	03:08	10
End of program 3 by team X		

- El archivo de entrada tiene un número indefinido de líneas; cada línea contiene 4 números.
- El primer par de número representa el tiempo inicial; el segundo par representa el tiempo final.
- En cada par de números, el primer número representa las horas el segundo representa los minutos.
- Las horas están en el rango de 1..12, los minutos de 0..59.
- El tiempo inicial y el tiempo final no estarán en el instante en la cual la manecilla de los minutos acabe de pasar la manecilla de la hora. (ejm. 12 00 no estará como hora inicial ni final).
- El tiempo inicial no será igual al tiempo final.
- Entre el tiempo inicial y el correspondiente tiempo final, la manecilla de la hora habrá dado una vuelta completa (360 grados).
- Como la manecilla de la hora empieza desde la posición inicial hasta su posición final esta podría pasar o no pasar por el número 12 en el dial del reloj.
 - o Si lo hace la hora inicial será “AM” y la hora final será “PM” o viceversa.
 - o Si no lo hace, entonces los dos serán “AM” o “PM”.
- Cada línea de la entrada corresponderá una línea en la salida, que contendrá.
 - o La hora inicial y final.
 - o El número de veces que la manecilla de los minutos pasa por la manecilla de la hora entre el tiempo inicial y el tiempo final.
- En cada caso, las horas y los minutos son mostradas con un tamaño de 2 caracteres, separados por dos puntos.

Un ejemplo del formato:
Tiempo Inicial Tiempo Final Pasadas
12:50 01:02 0

Puntuación Total		55
		Puntos
Detalle por Problema		Valor
Caso de prueba del parcial		11.5
Caso de prueba adicional		16

NOTAS:

- **Guarde su examen en una carpeta con:
Primer-Apellido-4utilmos-Dígitos-de-su-Cedula**
- **Deposite el examen en la dirección:
\\nimrod\Evaluacion\Programacion II\Diferidos**
- **No se repetirán exámenes. Es su responsabilidad que su examen se deposite correctamente en la dirección dada.**
- **Exámenes de código compartido tienen 0 Puntos.**
- **Solo se puede utilizar material impreso.**

