PONTIFICIA UNIVERSIDAD CATÓLICA DEL PERÚ
FACULTAD DE CIENCIAS E INGENIERÍA

ALGORITMIA
1ra práctica (tipo b)
(Primer semestre 2015)

Indicaciones generales:

- Duración: 2h 45 min.
- Materiales o equipos a utilizar: No se permite el uso de material de consulta.
- Al inicio de cada programa, el alumno deberá incluir, a modo de comentario, la estrategia que utilizará para resolver el problema. De no incluirse dicho comentario, el alumno perderá el derecho a reclamo en esa pregunta.
- Si la implementación es significativamente diferente a la estrategia indicada, la pregunta será corregida sobre el 50 % del puntaje asignado y sin derecho a reclamo.
- Un programa que no muestre resultados coherentes y/o útiles será corregido sobre el 60 % del puntaje asignado a dicha pregunta.
- Debe utilizar comentarios para explicar la lógica seguida en el programa elaborado.
- Cada programa debe ser guardado en un archivo con el nombre *preg#_<codigo_de_alumno>.c* y subido a PAIDEIA en el espacio indicado por los Jefes de Práctica.

Puntaje total: 20 puntos

Cuestionario:

Pregunta 1 (8 puntos) UVa 11827 - Maximum GCD

Given N integers, you have to find the maximum GCD (greatest common divisor) of every possible pair of these integers.

Input: The first line of input is an integer $N(1 < N < 100)$ that determines the number of test cases. The following N lines are the N test cases. Each test case contains a number $M(1 < M < 100)$ which indicates how many positive integers are in that line. Next to M appear the M positive integers of which you have to find the maximum of GCD.

Output: For each test case show the maximum GCD of every possible pair.

Sample Input:	Sample Output:

3
4 10 20 30 40
3 7 5 12
3 125 15 25
	20
	1
	25

To calcule the GCD of a pair of numbers you **must use recursion**.

Pregunta 2 (12 puntos) UVa 156 - Ananagrams

Most crossword puzzle fans are used to *anagrams* – groups of words with the same letters in different orders – for example OPTS, SPOT, STOP, POTS and POST. Some words however do not have this attribute, no matter how you rearrange their letters, you cannot form another word. Such words are called *ananagrams*, an example is QUIZ.

Obviously such definitions depend on the domain within which we are working; you might think that ATHENE is an ananagram, whereas any chemist would quickly produce ETHANE. One possible domain would be the entire English language, but this could lead to some problems. One could restrict the domain to, say, Music, in which case SCALE becomes a relative ananagram (LACES is not in the same domain) but NOTE is not since it can produce TONE.

Write a program that will read in the dictionary of a restricted domain and determine the **relative ananagrams**. Note that single letter words are, ipso facto, relative ananagrams since they cannot be "rearranged" at all. The dictionary will contain no more than 1000 words.

Input: Input will consist of a series of lines, each with one word. Words consist of up to 20 upper and/or lower case letters. Note that words that contain the same letters but of differing case are considered to be anagrams of each other, thus tIeD and EdiT are anagrams. The file will be terminated by a line consisting of a single #

Output: Output will consist of a series of lines. Each line will consist of a single word that is a relative ananagram in the input dictionary. Words must be output in lexicographic (case-sensitive) order. There will always be at least one relative ananagram.

Sample Input:	**Sample Output:**
ladder came tape soon leader acme RIDE lone Dreis peat ScAlE orb eye Rides dealer NotE derail LaCeS drIed noel dire Disk mace Rob dries #	Disk NotE derail drIed eye ladder soon

Profesores del curso: Fernando Alva
Robert Ormeño

San Miguel, 06 de abril de 2015