



**UNIVERSIDAD DISTRITAL
FRANCISCO JOSE DE CALDAS**

MARATON SEMANA TECNOLOGICA

CATEGORIA PRINCIPIANTES

UNIVERSIDAD DISTRITAL

5 de Noviembre de 2008

PROBLEMAS

**Elaborado por: Hector Florez
Basado de www.acis.org.co**

Problema 1

Fibonacci

Nombre del archivo fuente: fibonacci.c, fibonacci.cpp or fibonacci.java

Entrada: Entrada Estándar

Salida: Salida Estándar

La serie de fibonacci es una serie donde un número de la serie equivale a la suma de los dos anteriores. Esta serie contiene dos valores iniciales 0 y 1.

Entrada

La entrada consiste en varios casos. Cada caso tiene un número n que indica el limite de la serie. El ultimo caso antecede el valor 0.

Salida

Por cada caso, se debe imprimir la serie de fibonacci hasta el numero n. Los numero de la serie, deben estar separados por el carácter coma “,”.

Ejemplo de Entrada	Ejemplo de Salida
5	0,1,2,3,5
6	0,1,2,3,5,8
7	0,1,2,3,5,8,13
0	

Problema 2

Pirámide

Nombre del archivo fuente: piramide.c, piramide.cpp o piramide.java

Entrada: Entrada Estándar

Salida: Salida Estándar

Entrada

La entrada consiste en un conjunto de casos. Cada línea, describe un caso. En cada caso hay un número $1 \leq n \leq 20$ que indica la altura de la pirámide. El final de la entrada está indicado por una línea que contiene 0.

Salida

Por cada caso, se deben imprimir n filas. Cada fila tiene el número i que equivale al número de la fila repetido i veces, formando una pirámide.

Ejemplo de Entrada	Ejemplo de Salida
1	1
2	1
4	22
0	1
	22
	333
	4444

Problema 3

Decimal - Binario

Nombre del archivo fuente: decimal.c, decimal.cpp o decimal.java

Entrada: Entrada Estándar

Salida: Salida Estándar

Una conversión decimal – binario, es posible obtenerla mediante dos algoritmos:

1. Hacer divisiones sucesivas con divisor 2 capturando sus residuos del último al primero.

$$\begin{array}{r} 25 \quad | \quad 2 \\ 1 \quad | \quad 12 \quad | \quad 2 \\ \quad \quad | \quad 0 \quad | \quad 6 \quad | \quad 2 \\ \quad \quad \quad | \quad 0 \quad | \quad 3 \quad | \quad 2 \\ \quad \quad \quad \quad | \quad 1 \quad | \quad 1 \end{array}$$

Entonces el valor binario sería 11001_2

2. Calculando potencias base 2 de cada bit. Este método consiste en calcular el número binario equivalente al número decimal dado mediante la suma de los pesos binario que dan como resultado el número decimal. Es necesario tener en cuenta que los pesos binarios van de 2^0 hasta 2^n donde n es el número de bits. Esto es equivalente a decir que los pesos son: 1,2,4,8,16,32, etc.

$$\begin{array}{cccccc} 2^4 & 2^3 & 2^2 & 2^1 & 2^0 & \\ 1 & 1 & 0 & 0 & 1 & \\ 16 + 8 + 0 + 0 + 1 = 25 & & & & & \end{array}$$

Entrada

La entrada se compone de múltiples casos. Cada caso contiene un número decimal n entre 0 y 1000. La última línea antecede el valor 0

Salida

Por cada caso, se debe imprimir el número equivalente en binario de n

Ejemplo de Entrada	Ejemplo de Salida
10	1010
20	10100
0	

Problema 4

Factorial

Nombre del archivo fuente: factorial.c, factorial.cpp o factorial.java

Entrada: Entrada Estándar

Salida: Salida Estándar

Definición

El factorial de un número equivale al producto de 1 a n del número a evaluar n

Entrada

Hay múltiples casos. Cada caso esta contenido en una línea. Cada línea empieza con el número n, donde $1 \leq n \leq 10$. La ultima línea antecede el valor 0

Salida

Por cada caso se debe imprimir el factorial de n

Ejemplo de Entrada	Ejemplo de Salida
2	2
4	24
6	720
0	

Problema 5

Ordenamiento

Nombre del archivo fuente: ordenamiento.c, ordenamiento.cpp o ordenamiento.java

Entrada: Entrada Estándar

Salida: Salida Estándar

Dado un conjunto de cadenas de caracteres, se puede realizar ordenamiento alfabético de las mismas, en donde las letras a o A, serian las menores y las letras z o Z las mayores.

Entrada

Hay múltiples casos. La primer línea contiene un numero n que indica la cantidad de casos que se pretenden evaluar. La siguiente línea, contiene la cantidad de cadenas que se pretenden evaluar. Las siguientes líneas contienen las cadenas que se evaluaran.

Salida

Para cada uno de los n casos, se conoce cuantas cadenas se evaluaran. En el primer caso, en el cual se obtiene un número de cadenas m se hará el ordenamiento de las primeras m cadenas. Para el segundo caso, en el cual se obtiene un numero de cadenas p se hará el ordenamiento de las cadenas entre la numero m+1 hasta la cadena numero p. Esto se repite por el número de casos n. Por cada caso se deberá imprimir el texto Caso i, donde i es el numero de cada caso.

Ejemplo de Entrada	Ejemplo de Salida
3 3 hola Hector ABECEDARIO	Caso 1 ABECEDARIO Hector hola
4 PARCIAL cadena Maraton www.hectorflorez.com	Caso 2 cadena Maraton PARCIAL www.hectorflorez.com
3 C++ Programar es facil Habilidad	Caso 3 C++ Habilidad Programar es facil

Problema 6

Sumatoria

Nombre del archivo fuente: sumatoria.c, sumatoria.cpp or sumatoria.java

Entrada: sumatoria.in

Salida: Salida Estándar

Dado un arreglo de enteros, calcular la sumatoria dada la siguiente característica. La sumatoria será el resultado de la suma de todos los elementos del arreglo mas el numero mayor mas el numero menor.

Entrada

La entrada contiene múltiples casos. Cada caso esta dado por múltiples líneas. La primer línea contiene un numero n que indica la cantidad de números a sumar. Las siguientes n líneas contiene los números. El ultimo caso esta antecedido por el numero 0.

Output

Por cada caso de entrada la salida contiene el resultado de la sumatoria

Ejemplo de Entrada	Ejemplo de Salida
2	3
1	6
2	
3	
1	
2	
3	
0	