



UNIVERSIDAD AUTÓNOMA “TOMÁS FRÍAS” INGENIERÍA DE SISTEMAS

ESTUDIANTE:		C.I.:	
DOCENTE:	Ing. Ditmar Castro Angulo	MATERIA:	SIS-211 G2
AUXILIAR:	Univ. Gabriel Alejandro Garvizu Salas	FECHA:	18/09/2024
PRÁCTICA N2 AUX			

1. Realizar un programa que lea como entrada una cadena que está construida principalmente por 3 partes, un día cualquiera, un mes y un año. Mostrar por pantalla dicha cadena, pero con un formato solamente numérico, por ejemplo:

EJEMPLOS	
ENTRADA	SALIDA
4 de junio, 2004	4 / 06 / 2004
25 de diciembre, 2002	25 / 12 / 2002
28 de octubre, 2022	28 / 10 / 2022

Pd: Se debe considerar la cantidad de días que tienen los meses, es decir, si se da como entrada “30 de febrero, 2023” debe mostrar algún mensaje indicando el error.

2. Crear una función que tome una lista de palabras como argumento y cuente la cantidad de palabras que comienzan con una determinada letra.
3. En el juego Borderlands, los jugadores pueden encontrar y recolectar un variado conjunto de armas y equipo. Diseña un programa en JAVA que simule un sistema de botín para un jugador. Considera las siguientes condiciones:
 - Cada vez que el jugador derrota a un enemigo, tiene la oportunidad de encontrar botín.
 - El botín puede ser un arma, un escudo o un modificador de habilidad. Cada tipo de botín tiene un nivel de rareza: Común, Poco Común, Raro, Épico y Legendario.
 - Las armas tienen un nivel de daño, los escudos tienen una capacidad de absorción y los modificadores de habilidad mejoran alguna estadística específica del personaje.
 - Crea un método llamado “generarBotin” que simule el proceso de obtener botín. La función debe devolver un diccionario con la siguiente estructura:

```
{
'tipo': 'Arma',          # Puede ser 'Arma', 'Escudo' o 'Modificador'
'rareza': 'Épico',       # Puede ser 'Común', 'Poco Común', 'Raro', 'Épico' o 'Legendario'
'atributo_1': valor_1,   # Atributo específico del botín (por ejemplo, daño para armas)
'atributo_2': valor_2,   # Atributo adicional (puede ser capacidad de absorción para escudos)
'atributo_3': valor_3    # Otro atributo adicional (por ejemplo, mejora de habilidad para
                          # modificadores)
}
```

- Después de obtener el botín, el programa debe imprimir un mensaje indicando el tipo, rareza y atributos del botín encontrado (Deje volar su imaginación todo lo que pueda!!!).
- Preguntar al jugador si desea seguir enfrentándose a enemigos para obtener más botín. Si decide no continuar, muestra un resumen del botín obtenido durante la sesión.

Consejo: Para poder generar de mejor forma un botín aleatorio utilizar números pseudo aleatorios, Ejemplo:

```
import java.util.Random;
import java.util.Scanner;

public class Test1 {
    public static void main(String[] args) {
        Scanner read = new Scanner(System.in);
        Random random = new Random();

        for (int i = 0 ; i < 10 ; i++) {
            System.out.println(random.nextFloat());
        }
    }
}
```

4. Vinicius está en una reunión de la junta directiva del "Instituto de Consultoría de Palestras e Comentarios" (ICPC) pensando que sería bueno que los miembros de la junta fueran más concisos y mantuvieran sus discursos dentro del tiempo asignado para cada director, de modo que la reunión pudiera terminar antes del almuerzo. Desafortunadamente, tal vez debido a la naturaleza de la institución, a todos les encanta hablar.

Sabiendo que:

- Hay N directores que hablarán en la reunión.
- Cada director hablará durante la misma cantidad de tiempo.
- Y que entre dos discursos consecutivos hay un intervalo de 1 minuto.

Determine la duración máxima de cada discurso, en minutos, de modo que la reunión no dure más de K minutos.

Entrada

La primera línea contiene un entero N ($1 \leq N \leq 100$), el número de directores. La segunda línea contiene un entero K ($1 \leq K \leq 1000$ y $K \leq N$), la duración máxima de la reunión en minutos. En todos los casos de entrada, el discurso de cada director dura al menos 1 minuto.

Salida

Su programa debe generar una sola línea, que contenga un solo número entero, que indique la duración del discurso de cada miembro de la junta, en minutos.

Input example 1 7 120	Output example 1 16
Explicación del ejemplo 1: Hay 7 directores y la duración máxima de la reunión es de 120 minutos. Si cada director habla durante 16 minutos, tenemos $16 \times 7 = 112$ minutos. Como hay seis pausas entre los discursos y cada pausa dura un minuto, tenemos 118 minutos en total. Nótese que, en este caso, no se utilizan dos minutos del tiempo de la reunión y que, si los discursos fueran más largos de 16 minutos, el tiempo total superaría el límite de 120 minutos.	
Input example 2 1 10	Output example 2 10
Explicación del ejemplo 2: Hay un solo director y la reunión dura 10 minutos. Por lo tanto, el tiempo máximo de intervención del director es de 10 minutos.	
Input example 3 100 1000	Output example 3 9

5. La princesa de Nlogonia guarda su colección de perlas en un joyero cuadrado formado por N columnas, cada columna contiene V cajitas. Coloca un número diferente de perlas en cada cajita y ordena la cajita de manera que, en cada columna, de arriba a abajo, las cajitas contengan un número creciente de perlas y en cada fila, de izquierda a derecha, los agujeros también contengan un número creciente de perlas. La princesa sospecha que su hermana pequeña, que es muy traviesa, está trasteando con sus cosas en sus juegos. En particular, la princesa sospecha que su joyero ha sido girado 90 grados en el sentido de las agujas del reloj, posiblemente varias veces.

La figura (a) muestra un ejemplo de la disposición original de una caja de 4×4 . La figura (b) muestra la caja girada en el sentido de las agujas del reloj, 90 grados.

3	4	6	12
7	10	11	19
9	14	17	22
15	16	20	25

(a)

90°

15	9	7	3
16	14	10	4
20	17	11	6
25	22	19	12

(b)

Dado el número de perlas en cada caja, escriba un programa para determinar el número más pequeño de rotaciones de 90 grados en sentido antihorario que son necesarias para que el joyero vuelva a su estado original.

Entrada

La primera línea de la entrada contiene un entero N , el número de filas y columnas en la caja ($2 \leq N \leq 50$). Cada una de las siguientes N líneas contiene N enteros $K_{i,j}$ el número de perlas en la caja en la fila i y la columna j ($0 \leq K_{i,j} \leq 10^5$ para $1 \leq i \leq N$ y $1 \leq j \leq N$). En la entrada, las filas se dan de arriba hacia abajo y las columnas se dan de izquierda a derecha.

Salida

Su programa debe generar una sola línea que contenga solo un entero R (que puede ser 0, 1, 2 o 3), el número más pequeño de veces que se debe girar el joyero en sentido antihorario para volver a su estado original.

Input example 1 4 15 9 7 3 16 14 10 4 20 17 11 6 25 22 19 12	Output example 1 1
Explicación del ejemplo 1: Este ejemplo corresponde al del enunciado. Es necesario girar la caja en sentido antihorario una vez.	
Input example 2 3 300 250 150 280 200 140 240 190 130	Output example 2 2
Explicación del ejemplo 2: Es necesario girar la caja en sentido antihorario dos veces.	
Input example 3 2 2 4 1 3	Output example 3 3
Explicación del ejemplo 3: Es necesario girar la caja en sentido antihorario tres veces.	

6. El pequeño Charles era uno de los mejores programadores competitivos del mundo. Sin embargo, nunca le gustó mucho programar. Ahora que está jubilado, puede dedicar sus estudios a lo que realmente ama: continuar fracciones. Para prepararse para la próxima Imensa Competição de Phrações Contínuas (ICPC), necesita resolver el siguiente problema:

Defina $P_0 = 1$ como la fracción de nivel 0. Luego defina:

$$P_1 = \frac{1}{1 + 1}$$

Como la fracción de nivel 1 a P_1 y también,

$$P_2 = \frac{1}{1 + \frac{1}{1 + 1}}$$

Como la fracción de nivel 2 a P_2 y así sucesivamente.

Dado un valor entero N , ayude a Charles a determinar el valor del numerador de la fracción P_N .

Entrada

La primera y única línea contiene un entero N ($1 \leq N \leq 40$).

Salida

El valor P_N se puede escribir como una fracción de la forma $\frac{a}{b}$, donde a y b son coprimos. Imprima una línea que contenga el valor de a .

Input example 1 2	Output example 1 2
Input example 2 10	Output example 2 89

7. En la memoria de un dispositivo electrónico se almacena una lista de N números enteros a_1, \dots, a_N . Este dispositivo dispone de una operación muy peculiar: el intercambio de bits entre números. Más precisamente, dados los números enteros i, j y k , esta operación intercambia el k -ésimo bit del entero a_i con el k -ésimo bit del entero a_j (y viceversa). Al realizar esta operación una o más veces pueden ocurrir fenómenos muy interesantes, como por ejemplo obtener números que ni siquiera pertenecían a la lista original, o incluso números mayores o menores que todos los elementos originales.

Para este problema, nos interesa utilizar la operación tantas veces como sea necesario para cambiar la lista de números de modo que la lista resultante sea la máxima lexicográficamente, es decir, que a_1 sea el mayor posible, que a_2 sea el mayor posible entre las posibles soluciones que maximizan a_1 , y así sucesivamente.

Entrada

La primera línea de entrada contiene un entero N ($1 \leq N \leq 10^5$) y la segunda línea contiene N enteros, separados por espacios, correspondientes a la lista a_1, \dots, a_N ($0 \leq a_i \leq 10^9$).

Salida

Su programa debe imprimir una sola línea que contenga N enteros separados por espacios correspondientes a la secuencia máxima lexicográficamente obtenible.

Input example 1 4 8 4 2 1	Output example 1 15 0 0 0
Input example 2 4 12 15 1 20	Output example 2 31 3 4 0

8. Una vez completados los 7 ejercicios anteriores, es hora de realizar el ejercicio que vale más puntaje, hasta se podría decir que este ejercicio vale 100% de la práctica, puesto que si la carpeta con los distintos archivos no es subida a un repo de GitHub no podrá calificarse. Tal vez ya te hayas percatado de que va, en caso de que no, ahí va.

Deberá de subirse los distintos archivos que se desarrollaron a lo largo de la práctica, a su repo en GitHub para su revisión, caso contrario no será válida la entrega por ningún otro medio.

Buena suerte y nos vemos en la siguiente práctica.

Fecha de entrega limite: 18/09/2024