



Sistema Universidad Abierta

UNIDAD 5 ACTIVIDAD 2

Materia:

PROGRAMACIÓN DE DISPOSITIVOS MÓVILES

Asesor:

Cristian Cardoso Arellano

Semestre:

**6°
Semestre**

Alumno:

Sealtiel Esteban Solano Flores

Unidad 5.

Actividad 2.

1. A partir del siguiente problema, resolver la solución en lenguaje Java en Android Studio, imprimir el resultado en el logcat.

Descripción del problema

--- Conteo de calorías ---

Los renos de Papá Noel suelen comer comida normal para renos, pero necesitan mucha energía mágica para entregar regalos en Navidad. Por eso, su bocadillo favorito es un tipo especial de carambola que solo crece en lo profundo de la jungla. Los Elfos te han traído en su expedición anual al bosque donde crece la fruta.

Para suministrar suficiente energía mágica, la expedición debe recuperar un mínimo de cincuenta estrellas antes del 25 de diciembre. Aunque los Elfos te aseguran que la arboleda tiene mucha fruta, decides agarrar cualquier fruta que veas por el camino, por si acaso.

Recoge estrellas resolviendo acertijos. Habrá dos rompecabezas disponibles cada día en el calendario de Adviento; el segundo rompecabezas se desbloquea cuando completas el primero. Cada rompecabezas otorga una estrella. ¡Buena suerte!

La jungla debe estar demasiado cubierta de maleza y ser difícil de transitar en vehículos o acceder desde el aire; La expedición de los Elfos tradicionalmente va a pie. A medida que sus barcos se acercan a tierra, los Elfos comienzan a hacer un inventario de sus suministros. Una consideración importante es la comida, en particular, la cantidad de calorías que lleva cada Elfo (su entrada de rompecabezas).

Los Elfos se turnan para anotar el número de Calorías que contienen las distintas comidas, meriendas, raciones, etc. que han traído consigo, un elemento por línea. Cada Elfo separa su propio inventario del inventario del Elfo anterior (si lo hay) con una línea en blanco.

Por ejemplo, supongamos que los Elfos terminan de escribir las Calorías de sus artículos y terminan con la siguiente lista:

1000
2000
3000
4000
5000
6000
7000
8000
9000
10000

Esta lista representa las Calorías de la comida que llevan cinco Elfos:

El primer Elfo lleva comida con 1000, 2000 y 3000 Calorías, un total de 6000 Calorías.

El segundo elfo lleva un alimento con 4000 calorías.

El tercer Elfo lleva comida con 5000 y 6000 Calorías, un total de 11000 Calorías.

El cuarto Elfo lleva comida con 7000, 8000 y 9000 Calorías, un total de 24000 Calorías.

El quinto Elfo lleva un alimento con 10000 Calorías.

En caso de que los Elfos tengan hambre y necesiten refrigerios adicionales, necesitan saber a qué Elfo preguntar: les gustaría saber cuántas Calorías lleva el Elfo que lleva la mayor cantidad de Calorías. En el ejemplo anterior, esto es 24000 (llevado por el cuarto Elfo).

```
package com.example.myapplication;

import androidx.appcompat.app.AppCompatActivity;
import android.os.Bundle;
import android.util.Log;

public class MainActivity extends AppCompatActivity {

    private static final String TAG = "ConteoCalorias";

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        encontrarMaximoCalorias();
    }

    /**
     * problema del conteo de calorías
     * imprime el resultado en Logcat.
     */
    private void encontrarMaximoCalorias() {

        // 1. Los datos de entrada del ejemplo como un solo String
        String input = "1000\n" +
            "2000\n" +
            "3000\n" +
            "\n" + // Línea en blanco
            "4000\n" +
            "\n" + // Línea en blanco
            "5000\n" +
            "6000\n" +
            "\n" + // Línea en blanco
            "7000\n" +
            "8000\n" +
            "9000\n" +
            "\n" + // Línea en blanco
            "10000";

        // 2. Variables para guardar el máximo y la suma actual
        int maxCalorias = 0;
        int caloriasElfoActual = 0;

        // 3. Dividir la entrada por líneas
        String[] lineas = input.split("\n");

        // 4. Recorrer cada línea
        for (String linea : lineas) {
```

```

        if (linea.trim().isEmpty()) {
            // Es una línea en blanco: hemos terminado con este elfo

            // Comparamos si la suma de este elfo es la nueva máxima
            if (caloriasElfoActual > maxCalorias) {
                maxCalorias = caloriasElfoActual;
            }

            // Reiniciamos el contador para el siguiente elfo
            caloriasElfoActual = 0;
        } else {
            // Es una línea con un número
            try {
                // Convertimos el String a entero y lo sumamos
                int calorias = Integer.parseInt(linea.trim());
                caloriasElfoActual += calorias;
            } catch (NumberFormatException e) {
                // Ignorar si la línea no es un número válido
                Log.w(TAG, "Línea ignorada, no es un número: " +
linea);
            }
        }
    }

    // 5. Verificación final (para el último elfo)

    if (caloriasElfoActual > maxCalorias) {
        maxCalorias = caloriasElfoActual;
    }

    // 6. Imprimir el resultado final en Logcat (nivel "Debug")
    Log.d(TAG, "--- Problema Resuelto ---");
    Log.d(TAG, "El elfo con más calorías lleva: " + maxCalorias);
}
}

```

