# Análisis del problema:

En el contexto del proyecto Blind.AR para la fabricación de metales recuperados de pilas gastadas, el Coordinador de Vinculación con el Medio de la UNDAV solicita el desarrollo de un prototipo de software (programado en Python 3) que produzca estadísticas mensuales a partir de datos de recolección de pilas usadas, provenientes de universidades bonaerenses del Bicentenario.

Un/a/e único/a/e usuario/a/e accederá al sistema en algún momento del mes, para volcar los datos de la recolección que le enviaron las universidades. Cuando algún dato ingresado que requiera validación resulte erróneo, el programa deberá pedir su reingreso hasta que cumpla los requisitos de validez especificados.

El programa deberá solicitar el ingreso del mes y el año del informe, considerando como criterios de validez que el mes informado sea un número de mes posible y el año no sea anterior a 2023.

Por cada universidad de la cual se haya recibido información, debe pedirle al usuario que ingrese los siguientes datos:

* El nombre abreviado de la universidad
* El día de la fecha de envío
* Las cantidades de gramos de pilas usadas enviadas ese mes, clasificadas según su composición (tipos de pila salinas o alcalinas o de litio) o según su tamaño (pilas de botón) sin considerar su composición.

El sistema debe considerar como criterios de validez para los datos ingresados de cada universidad:

* que el día de envío determine una fecha posible en el período informado
* que la cantidad de gramos de cada tipo de pilas enviadas no sea negativa ni supere el valor 99000
* que la cantidad total de pilas enviadas (suma de las cantidades de pilas salinas, alcalinas, de litio y de botón) sea mayor que cero, pero menor que 350 kilogramos (será necesario convertir la cantidad total, de gramos a kilogramos).

Los datos válidos de cada universidad informada deberán mostrarse en pantalla, simultáneamente, con su fecha de envío, su cantidad total de kilogramos de pilas enviadas y el respectivo porcentaje de pilas de litio (cociente entre la cantidad de pilas de litio, multiplicada por 100, y la cantidad total de pilas enviadas por la universidad).

Además, el programa deberá determinar y, finalmente, mostrar con descripciones expresivas la siguiente síntesis estadística: el período que se está informando; la cantidad total de universidades informadas; el total de kilogramos de pilas usadas enviadas; el total de kilogramos de pilas salinas, de pilas alcalinas, de pilas de litio y de pilas de botón; el porcentaje de pilas de litio relativo a la cantidad total de kilogramos de pilas enviadas; también deberá mostrar la mayor cantidad de kilogramos de pilas enviadas ese mes (por una o más universidades) y la cantidad de universidades que no enviaron pilas tipo botón en el periodo informado.

# Especificación de la solución:

El sistema deberá pedirle al usuario que ingrese datos para generar un informe estadístico mensual de la recolección de pilas usadas. Para algunos de esos datos debe verificar que sean válidos antes de continuar y en caso de que el dato ingresado no cumpla con los requisitos de validación, deberá solicitar su reingreso hasta asegurar su validez.

Debe solicitar al usuario el ingreso del Mes y el Año del informe, asegurando que ambos sean válidos (mes entre 1 y 12, año igual o posterior a 2023).

Por cada universidad informada debe solicitar el ingreso de:

* + El Nombre abreviado de la Universidad
  + El Día de la fecha de envío (debe controlar que sea válido dentro del mes del período informado)
  + Las Cantidades de gramos de pilas usadas enviadas ese mes, discriminadas según el tipo en: Pilas salinas, Pilas alcalinas, Pilas de litio y Pilas de botón (debe verificar que la cantidad de gramos de cada tipo de pilas sea mayor o igual que cero y menor o igual que 99000; a su vez, debe controlar que la suma de las cantidades expresada en kilogramos -cantidad total de kilogramos de pilas enviadas– sea mayor que cero, pero menor que 350)

Con los datos de cada universidad informada, debe calcular el Porcentaje de pilas de litio enviadas (como el cociente de: la cantidad de gramos de pilas de litio, multiplicado por 100, dividido entre la respectiva cantidad total de gramos de pilas enviadas por la universidad) y mostrar el resultado en pantalla, simultáneamente con los datos válidos de la universidad, incluidos la Fecha de envío y su Cantidad total de kilogramos de pilas enviadas en esa fecha.

Además, deberá determinar y, finalmente, mostrar con descripciones claras la siguiente síntesis estadística: el período que se está informando; la Cantidad total de universidades informadas; el Total de kilogramos de pilas usadas enviadas; el Total de kilogramos de Pilas salinas, de Pilas alcalinas, de Pilas de litio y de Pilas botón; el Porcentaje de pilas de litio relativo a la cantidad total de kilogramos de pilas enviadas; la Mayor cantidad de kilogramos de pilas enviadas ese mes (por una o más universidades) y la Cantidad de universidades que no enviaron pilas tipo botón en el período informado.

## Datos de entrada

Datos del informe:

* + Mes
  + Año

Datos de cada universidad informada:

* + Nombre abreviado de la universidad.
  + Día de la fecha de envío de las pilas.
  + Cantidad de gramos de pilas salinas enviadas por la universidad.
  + Cantidad de gramos de pilas alcalinas enviadas por la universidad.
  + Cantidad de gramos de pilas de litio enviadas por la universidad.
  + Cantidad de gramos de pilas botón enviadas por la universidad

## Datos de salida

Por cada universidad informada:

* + Nombre abreviado de la universidad
  + Fecha de envío de las pilas (Día, Mes, Año)
  + Cantidad de gramos de pilas salinas enviadas por la universidad
  + Cantidad de gramos de pilas alcalinas enviadas por la universidad
  + Cantidad de gramos de pilas de litio enviadas por la universidad
  + Cantidad de gramos de pilas botón enviadas por la universidad
  + Cantidad total de kilogramos de pilas enviadas por la universidad
  + Porcentaje de pilas de litio enviadas, relativo al total de la universidad

Período del informe (Mes, Año)

Cantidad total de universidades informadas

Cantidad total de kilogramos de pilas usadas enviadas Cantidad total de kilogramos enviados del tipo de pilas salinas Cantidad total de kilogramos enviados del tipo de pilas alcalinas Cantidad total de kilogramos enviados del tipo de pilas de litio Cantidad total de kilogramos enviados del tipo de pilas de botón

Porcentaje de pilas de litio relativo a la cantidad total de pilas enviadas Mayor cantidad de kilogramos de pilas usadas enviadas ese mes Cantidad de universidades que no enviaron pilas tipo botón en el período

# Diseño de la solución:

Se solicita el ingreso del año **(anio)**, mientras el año sea menor que 2023, pide el reingreso del dato.

Se ingresará también el valor del mes **(mes)** de ingreso, mientras no sea mayor o igual a uno y menor o igual a doce, pide el reingreso del dato. Se iniciarán variables acumuladoras: **cant\_univ, uni\_no\_boton, gramos total ,**

**total\_pilas\_salinas, total\_pilas\_alcalinas, total\_pilas\_litio, total\_pilas\_boton, máximo\_enviado, total\_kg\_unis.**

Se iniciará un ciclo interactivo para solicitar los datos de cada universidad informada y se preguntará

si desea seguir ingresando universidades. Se pedirá al usuario que ingrese el nombre abreviado (**abrev\_uni**) de la universidad que aumenta más uno el acumulador (**cant\_univ**). Se solicita el día de envió (**día**) de la universidad, mientras la invocación a la

función **fecha\_valida (día, mes, anio)** devuelva falso (False), se pedirá el reingreso del día. Se da valor inicial 0 a ala variable (máximo\_enviado).

Se da valor inicial 0 a la cantidad de pilas enviadas por la universidad (**gramos\_total**), mientras no sea mayor que 0 y menor que 350000, repite las siguientes acciones:

Se pide al usuario que ingrese la cantidad de gramos de pilas alcalinas **(cant\_pilas\_alcalinas)**,invocando a la

función **pide\_cant\_pilas (cant\_pilas\_alcalinas)** y se sumara también a la variable (**gramos\_total**) la cantidad valida.

Se sumara la cantidad validad ingresada a (total\_pilas\_alcalinas) diviendo por 1000

Se pide al usuario que ingrese la cantidad de gramos de pilas de litio (**cant\_pilas\_litio**),

invocando a la función **pide\_cant\_pilas (cant\_pilas\_litio)** y se sumara también a la variable (**gramos\_total**) la cantidad valida.

Se sumará la cantidad valida ingresada a **(total\_pilas\_litio)** dividiéndolo por 1000.

Se pide al usuario que ingrese la cantidad de gramos de pilas salinas (**cant\_pilas\_salinas**),

invocando a la función **pide\_cant\_pilas (cant\_pilas\_salinas**) y se sumara también a la variable (**gramos\_total**) la cantidad valida. 

Se sumará la cantidad valida ingresada a **(total\_pilas\_salinas)** dividiéndolo por 1000**.**

Se pide al usuario que ingrese la cantidad de gramos de pilas botón (**cant\_pilas\_boton**), invocando a la función **pide\_cant\_pilas (cant\_pilas\_boton)** y se sumara también a la variable (**gramos\_total**) la cantidad valida. Se sumará la cantidad

El programa calculara el porcentaje de pilas de litio (**porcentaje\_pilas\_litio)**, entre la cantidad de pilas de litio informadas (**cant\_pilas\_litio**), y la cantidad total de pilas enviadas por esa universidad (**gramos\_total**). Se mostrará en pantalla el mismo con el

nombre de la universidad que informo los datos (**abrev\_uni**) y su cantidad total en kg de pilas enviadas (**total\_de\_kg**) que divide los **gramos\_total** a 1000.

Se sumará la cantidad de kilogramos enviados a **(total\_kg\_unis).** Se evaluara si el máximo enviado (**máximo\_enviado)**, es menor que el total de kg (**total\_de\_kg**), si eso se cumple se guardara el valor (**total\_de\_kg**).

Al salir del ciclo las variables: **cant\_uni, total\_pilas\_salinas, total\_kg\_unis, total\_pilas\_alcalinas, total\_pilas\_litio, total\_pilas\_boton, porcentaje\_total, máximo\_enviado, uni\_no\_boton.**

Determinaran la cantidad de universidades informadas (**cant\_uni**), el total de kg de pilas enviadas entre todas las universidades (**total\_kg\_unis**), el total de cada tipo de pila (**total\_pilas\_salinas, total\_pilas\_alcalinas, total\_pilas\_litio, total\_pilas\_boton)**, el porcentaje total de pilas enviadas del tipo litio (**porcentaje\_total**), la mayor cantidad de kg de pilas enviadas por una universidad (**máximo\_enviado**) y la cantidad de universidades que no enviaron del tipo botón (**uni\_no\_boton)**.

***Funciones:***

**Fecha\_valida (día, mes, anio):** dado por parámetros una fecha en números (día, mes, anio), devuelve un resultado booleano que indica si es válida o no. Invocando a la función: **cant\_dias\_mes (mes, anio)**

Entradas: día, mes, año. Salidas: booleano.

**Es\_bisiesto (anio):** recibe por parámetros un número que representa un año y devuelve un resultado booleano que indique si es, o no es, bisiesto.

Entrada: año. Salida: booleano.

**Cant\_dias\_mes (mes, anio):** dados por parámetros dos números que representan el mes y el año, devuelve como resultado la cantidad de días correspondiente al mes. Invocando a la función: **es\_bisiesto (anio).**

Entradas: mes, año.

Salidas: días correspondientes al mes.

**pide\_cant\_pilas (gramos):** Pide el ingreso de la cantidad de gramos de pilas, mientras la cantidad no sea mayor o igual que 0 y menor o igual que 99000, pide el reingreso del dato. Devuelve una cantidad válida.

Entradas: cantidad de gramos. Salidas: cantidad de gramos.