



## Objetivos de la Guía

- Recordar el uso diferentes tipos de variables
- Recordar respecto al uso de strings
- Recordar respecto al uso de condicionales

## >A - Javier guarda diferentes cosas

Javier va a comenzar su aventura en el taller avanzado. Entusiasmado, él empieza a programar para recordar los conceptos que necesita. Por lo que decide trabajar en un problema que le conto su amiga Paula.

### \* Parte 1: Leer información y mostrarlos por pantalla

La primera parte del problema consiste en leer 5 elementos por pantalla:

- un entero
- un flotante
- un entero grande
- un string
- un caracter

Y luego mostrar por pantalla esos elementos al revés de que fueron leídos.

#### \* Input:

```
2 12.32 10000000000000 hola c
```

#### \* Output:

```
c hola 10000000000000 12.32 2
```

### Tip 1

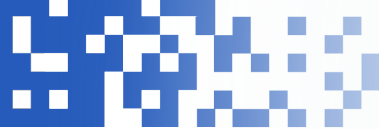
Recordar que para leer y mostrar por pantalla, se debe incluir la librería `iostream`. Si quieren leer con C++, recuerde el uso de `cin` y `cout`.

```
#include <iostream>

int main() {
    // tu codigo
    return 0;
}
```

### Tip 2

Recuerden que un entero grande se llama `long long int`.



## \* Parte 2: Jugar con los diferentes tipos

La primera parte del problema consiste en leer los mismos 5 elementos por pantalla de la parte anterior.

Pero, Paula quiere ahora aplicar operaciones a los elementos. Se deben realizar las 5 operaciones siguiente:

- Al entero se le debe sumar el elemento flotante
- Al flotante se le debe dividir por 1000
- Al entero grande dividirlo por el entero
- Al string se le debe sumar si mismo
- Al caracter se le debe sumar 1

Luego mostrar el resultado de cada una de las operaciones realizadas.

### \* Input:

```
3 123.2123 1000000000 poro d
```

### \* Output:

```
126 0.1232123 333333333 poroporo e
```

Explicación: El entero vale 3, el flotante vale 123.2123, el entero grande 1000000000, el string poro y el caracter d.

Entonces, los resultados son:

- $3 + 123.2123 = 126.2123$ , pero como es una variable entero el resultado final es 126
- $123.2123 / 1000 = 0.1232123$
- $1000000000 / 3 = 333333333$ , nunca fue un flotante porque son dos enteros
- `poro + poro = poroporo`
- $c + 1 = d$ , recuerden que los caracteres siguen los valores de la tabla ASCII, c corresponde a 97, si le sumo 1 da 98 que corresponde al caracter d

Ahora, Paula le gusta probar cosas, entonces Javier termino su código y funciona correctamente. Y Paula le ofrece los siguientes inputs que provocarán efectos inesperados:

```
2147482648 10000.1233 333333333 buenaloscabrosdelyoutube d
```

```
2147482648 10000.1233 333333333 buenaloscabrosdelyoutube z
```

Javier sin entender, te dice que pruebes estos inputs veas que sucede e intentes obtener una explicación.

### Tip

Recuerde el rango de los enteros y revise el valor de z en la tabla ASCII



## >B - Paula y las preguntas

Paula es una estudiante que se hace muchas preguntas, pero esta muy cansada para realizar las preguntas mentalmente. Por lo que te pide ayuda para hacer programa que resuelva sus preguntas.

### \* Parte 1: Numeros en desorden

Paula quiere leer 3 numeros por pantalla y hacer un código que muestre por pantalla esos mismos números en orden.

#### \* Input:

#### \* Output:

#### Tip

Recuerde que para saber si un número  $a$  es mayor a  $b$  se debe utilizar hacer  $a > b$

### \* Parte 2: Palabras en desorden

Paula quiere leer 3 palabras por pantalla y hacer un código que los muestre por pantalla en orden alfabetico.

#### \* Input:

#### \* Output:

#### Tip

Al igual que los numeros, los strings se pueden comparar utilizando el operador  $>$