



Trabajo individual

Identificación del trabajo

- a. **Módulo:** 2
- b. **Asignatura:** Lógica de Programación
- c. **RA:** Agregar condicionales al código.
- d. **Docente Online:** Richard Orellana Taibo.
- e. **Fecha de entrega:** 17/04/25

Identificación del estudiante

Nombre y apellido	Carrera
Christian Aránguiz	Técnico en Informática

Desarrollo

Dirección del código: [GitHub](#).

1. Código extraído desde PseInt:

```
Proceso Adivinanzas
    // Consignas:
    // 1.- Obtener (declarar) numero min, max y max de intentos
    Definir limInferior, limSuperior, numeroEscogido Como
Entero;
    Definir intento, maxIntentos, iteracion, intentosRestantes
Como Entero;

    //2.- Pedir límites
    Escribir "Ingrese número mínimo para adivinar";
    Leer limInferior;
    Escribir "Ingrese número máximo para adivinar";
    Leer limSuperior;

    // revisar que el limite inferior no sea mayor al superior
    si limSuperior < limInferior Entonces
        Escribir "ERROR ==> El número máximo no puede ser
menos al mínimo";
    SiNo
        // 3.- Pedir número máximo de intentos
        Escribir "Ingrese cantidad máxima de intentos";
        Leer maxIntentos;
        //4.- Generar número aleatoreo
        numeroEscogido <- obtenerNumero(limInferior,
limSuperior);
        // Generar iteracion para crear maximo de intentos
        // + contador de intentos (iteracion)
        Para iteracion <- 1 Hasta maxIntentos Con Paso 1 Hacer
            Escribir "Intento: ", iteracion, " de " ,
maxIntentos, ": Ingrese su número";
            Leer intento;
            // Revisar de que los intentos se encuentren
dentro de los parámetros.
            Si intento > limSuperior Entonces
                Escribir "Error ==> El número ingresado no
puede ser superior al numero máximo a adivinar";
            FinSi
            Si intento < limInferior Entonces
                Escribir "Error ==> El número ingresado no
puede ser inferior al numero mínimo a adivinar";
```

```

FinSi
intentosRestantes <- maxIntentos - iteracion;
// Comparar número generado contra intento del
usuario
// + revision de intentos restantes.
Si intento < numeroEscogido Entonces
    Escribir "El número es mayor. Te quedan ",
intentosRestantes, " intentos";
FinSi;
Si intento > numeroEscogido Entonces
    Escribir "El número es menor. Te quedan ",
intentosRestantes, " intentos";
FinSi
Si intento = numeroEscogido Entonces
    Escribir "Felicidades, el número era: ",
intento;
    iteracion <- maxIntentos + 1;
finsi;
FinPara
// Si no lo adivinó
Si intento <> numeroEscogido Entonces
    Escribir "Lo siento, no adivinaste. El número
era: ", numeroEscogido;
FinSi
FinSi
FinProceso

SubProceso resultado <- obtenerNumero ( min, max )
    Definir rango, num, resultado Como Entero;
    Definir random Como Real;

    rango <- max - min + 1;
    random <- Azar(rango) ;
    resultado <- random + min;
FinSubProceso

```

2. Explicación del código:

En el pseudocódigo desarrollado, si bien algunas estructuras condicionales ya se habían implementado en la actividad anterior, en esta segunda etapa se ampliaron y ajustaron conforme a los nuevos requerimientos. Se incorporó el cálculo de intentos restantes, proporcionando retroalimentación precisa al usuario después de cada intento, lo cual mejora la interacción y el control del flujo del juego. Asimismo, se fortaleció la validación de entradas, añadiendo una condición para asegurar que el número ingresado no sea inferior al límite mínimo definido.

Estos refinamientos contribuyen a una mayor solidez lógica y funcionalidad del algoritmo.

3. Resolución de consignas:

