

**Universidad Mariano Gálvez de Guatemala**

**Facultad de Ingeniería**

**Centro Universitario Antigua Guatemala**



**CARRERA**

**Ingeniería En Sistemas**

# **Proyecto Curso**

**Curso: Algoritmos**

**Catedrático: “Ing Luis Ángel Cashaj”**

**Oscar Alejandro Flores Yllescas - 0910-25-5678**

**Daniela Karla Maria Figueroa Galvez-0910-19-19980**

**Oscar Emmanuel Cay Joj-0910-25-1484**

**Diego Alejandro Cordova Alvarez-0910-25-3363**

**Emilio José Ubeda Agosto-0910-24-14263**

**Antigua Guatemala, 08 de noviembre de 2025**

## **Introducción**

Este documento corresponde al inciso (a) del Proyecto No. 1, que solicita el análisis y diseño del programa 'Simulador de Viaje Espacial'. El propósito es estructurar, con un enfoque académico, la solución propuesta en C++ para simular un viaje intergaláctico de diez días, en el cual el jugador administra recursos críticos (combustible, oxígeno, suministros e integridad de la nave) y decide acciones diarias bajo condiciones de incertidumbre controlada.

La simulación promueve la toma de decisiones basada en información parcial y en el análisis de riesgos, ofreciendo un contexto significativo para la programación estructurada.

En este inciso se documentan las acciones del programa, los datos de entrada, el inventario de variables con su rol y dominio, las condiciones y cálculos que rigen la dinámica del sistema y la descripción del algoritmo que orquesta la ejecución. Asimismo, se incorpora un espacio reservado para el diagrama de flujo, conforme a lo solicitado en la guía.

## Índice

<b>Acciones del Programa .....</b>	<b>4</b>
<b>Datos de Entrada .....</b>	<b>5</b>
<b>Variables.....</b>	<b>6</b>
<b>Condiciones y Cálculos .....</b>	<b>9</b>
<b>Modelo de aleatoriedad y plan de pruebas.....</b>	<b>10</b>

## Acciones del Programa

El sistema ofrece un conjunto de acciones diarias que el jugador puede seleccionar, cada una con costos, beneficios y riesgos definidos. Todas las acciones actualizan el estado de los recursos y pueden desencadenar eventos aleatorios.

- Explorar un planeta cercano: consume 15 unidades de combustible y habilita la posibilidad de encontrar recursos. Resultados potenciales y probabilidades: oxígeno (20–40 u., 60%), combustible (10–30 u., 25%), suministros (30–100 u., 50%), tormenta eléctrica (–10% a –20% de integridad, 25%), aterrizaje forzado (–10% a –20% de integridad, 25%).
- Reparar la nave: requiere suministros en razón de 10 unidades por cada punto porcentual de integridad a restaurar. El jugador indica el porcentaje deseado (sin exceder 100%).
- Enviar señales: evento binario con igual probabilidad. Caso A (ayuda): +60 de combustible. Caso B (piratas): –15% de integridad y –20 suministros.
- Noche (automática): consumo de oxígeno (–20 u.) y suministros (–30 u.). Con 15% de probabilidad ocurre un evento nocturno: tormenta cósmica (–10 oxígeno), encuentro alienígena amistoso (+20 combustible) u hostil (–10% integridad), o meteoritos (maniobrar: –10 a –30 combustible; no maniobrar: –15% a –25% integridad).
- Modo hacker: Permite alterar los valores iniciales del sistema, modificando combustible, oxígeno, suministros e integridad, además se potencia al capitán.

## Datos de Entrada

Los datos de entrada permiten personalizar la experiencia de juego y controlan las decisiones del jugador. Se requiere validar tipo y rango para prevenir estados inválidos.

Dato de entrada	Tipo	Uso / Validación
Nombre del capitán	String	Personaliza mensajes; validar longitud > 0.
Acción diaria (menú)	int	Valores válidos: 1–4 y 99; rechazar otros y volver a pedir.
% de reparación	int	Solo si acción=Reparar; 1–100; verificar suministros suficientes.
Confirmación (continuar/rendirse)	char/string	Opciones válidas: S/N; normalizar a mayúsculas.

Criterios de validación: evitar números negativos, impedir integridad > 100%, y rechazar acciones cuando los recursos sean insuficientes para ejecutarlas (por ejemplo, explorar sin combustible disponible).

## Variables

Variable	Tipo	Inicialización	Descripción / Dominio
combustible	int	30	0–n; energía para desplazamiento y maniobras.
oxigeno	int	50	0–n; consumo diario de 20 unidades.
suministros	int	40	0–n; consumo diario de 30 y para reparar.
integridad	float	100.0	0–100; salud estructural de la nave.
dias	int	0	1–10; termina al llegar a 10.
accion	int	0	Selección de menú (1–3).
evento	int	0	Identificador de suceso aleatorio.

opcion	int	0	Almacena la opción del menú seleccionada por el jugador (1–4, y 99 para modo hacker).
estasJugando	bool	true	Controla la ejecución principal del juego; se vuelve false al rendirse o finalizar.

nombreCapitan	string	—	Guarda el nombre ingresado por el jugador, usado en mensajes personalizados.
hackerActivado	bool	false	Bandera para identificar si se activó el modo hacker durante el turno actual.
porcentajeReparar	int	—	Porcentaje que el jugador desea reparar de la nave (0–100).
suministrosNecesarios	int	—	Suministros requeridos para la reparación (porcentajeReparar * 10).
probabilidad	int	rand() % 100	Determina si ocurre un evento nocturno (15% de probabilidad).
tipoevento	int	rand() % 3	Tipo de evento nocturno (tormenta cósmica, encuentro alienígena o meteoritos).
tipoalien	int	rand() % 2	Distingue entre alienígenas amistosos u hostiles.
decision	int	—	Elección del jugador en evento de meteoritos (maniobrar o resistir el impacto).

combustibleGastado	int	—	Unidades de combustible perdidas al maniobrar (10–30).
dano	int	—	Porcentaje de daño sufrido en eventos negativos.
valor	int	—	En modo hacker, almacena el valor nuevo ingresado para una variable.
oxigenoEncontrado	int	—	Cantidad (20–40) de oxígeno hallada al explorar.
combustibleEncontrado	int	—	Cantidad (10–30) de combustible hallado al explorar.
suministrosEncontrados	int	—	Cantidad (30–100) de suministros hallada al explorar.

Las variables se imprimen al final de cada día para retroalimentar al jugador sobre el estado actual y soportar decisiones informadas.



## Condiciones y Cálculos

Reglas de del juego y fórmulas clave:

- Consumo nocturno:  $\text{oxigeno} = \text{oxigeno} - 20$ ;  $\text{suministros} = \text{suministros} - 30$ .
- Reparación:  $\text{costo} = \text{porcentaje} * 10$ ;  $\text{integridad} = \min(100, \text{integridad} + \text{porcentaje})$ .
- Exploración:  $\text{combustible} = \text{combustible} - 15$ ; ganancias aleatorias según probabilidad.
- Meteoritos: decidir maniobra ( $-10$  a  $-30$  combustible) o impacto ( $-15\%$  a  $-25\%$  integridad).
- Fin del juego: si  $\text{combustible} \leq 0$ ,  $\text{oxigeno} \leq 0$  o  $\text{integridad} \leq 0$ ; o si  $\text{dias} = 10$  (victoria); o si el jugador se rinde.

Restricciones y salvaguardas: no permitir valores negativos; verificar precondiciones antes de ejecutar acciones; encapsular la aleatoriedad con límites superiores e inferiores explícitos; usar funciones para normalizar y validar entradas.

## Modelo de aleatoriedad y plan de pruebas

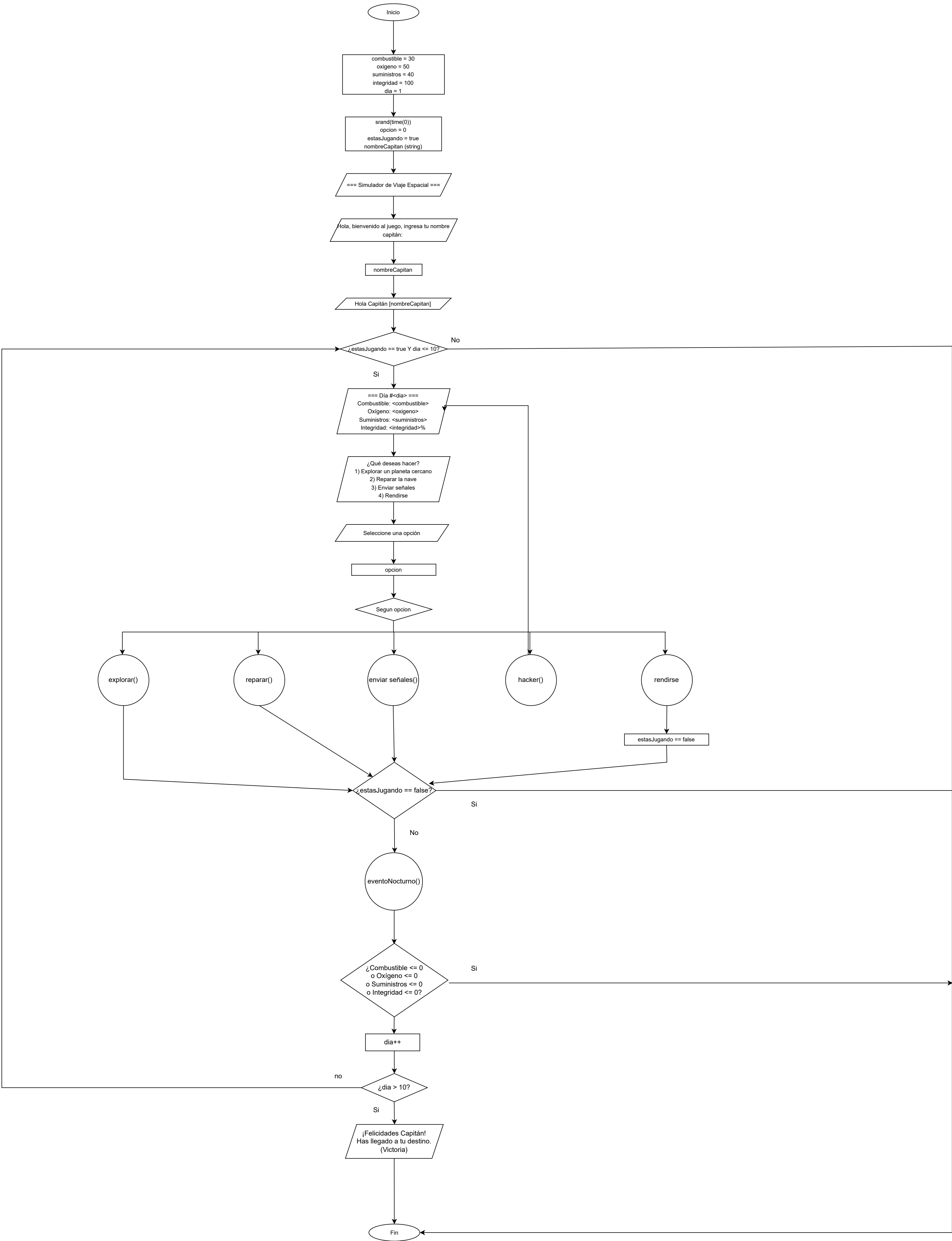
Para la generación de eventos se emplean números aleatorios uniformes acotados que se comparan con umbrales de probabilidad. Se recomienda inicializar la semilla con 'srand(time(NULL))' para evitar secuencias repetidas entre ejecuciones.

Evento	Probabilidad	Efecto
Tormenta eléctrica (explorar)	25%	−10% a −20% integridad
Aterrizaje forzado (explorar)	25%	−10% a −20% integridad
Hallazgo de oxígeno (explorar)	60%	+20 a +40 oxígeno
Hallazgo de combustible (explorar)	25%	+10 a +30 combustible
Hallazgo de suministros (explorar)	50%	+30 a +100 suministros
Ayuda al enviar señales	50%	+60 combustible
Piratas al enviar señales	50%	−15% integridad y −20 suministros
Evento nocturno	15%	Tormenta cósmica / Alienígenas / Meteoritos

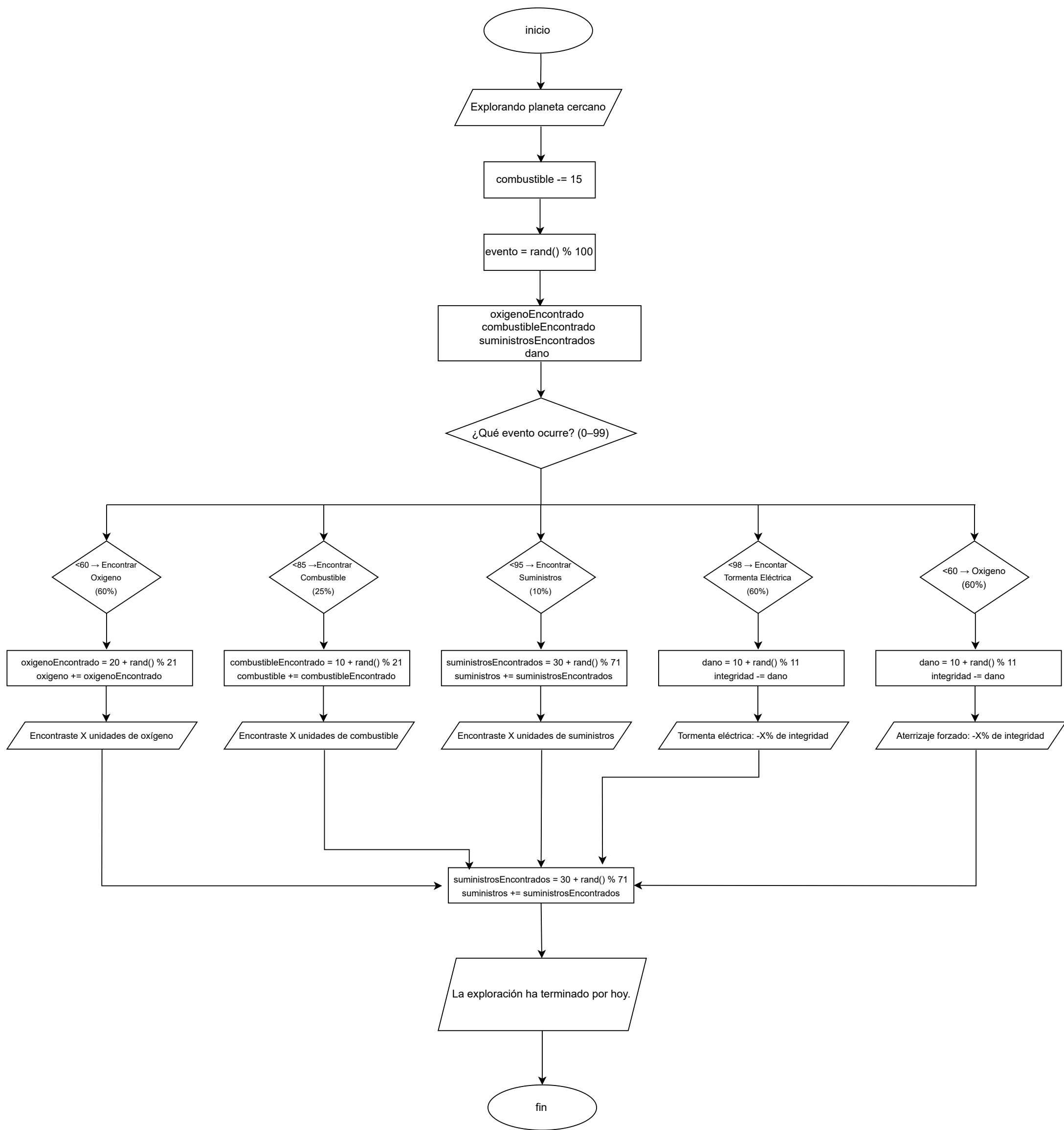
Plan de pruebas (escenarios sugeridos):

- Escenario A (camino ganador): decisiones conservadoras, reparaciones oportunas, exploración moderada.
- Escenario B (riesgo alto): exploración frecuente y pocas reparaciones; verificar supervivencia mínima.
- Escenario C (estrés de recursos): iniciar con valores cercanos al umbral (pruebas de borde).
- Escenario D (validación): entradas fuera de rango para confirmar manejo de errores.

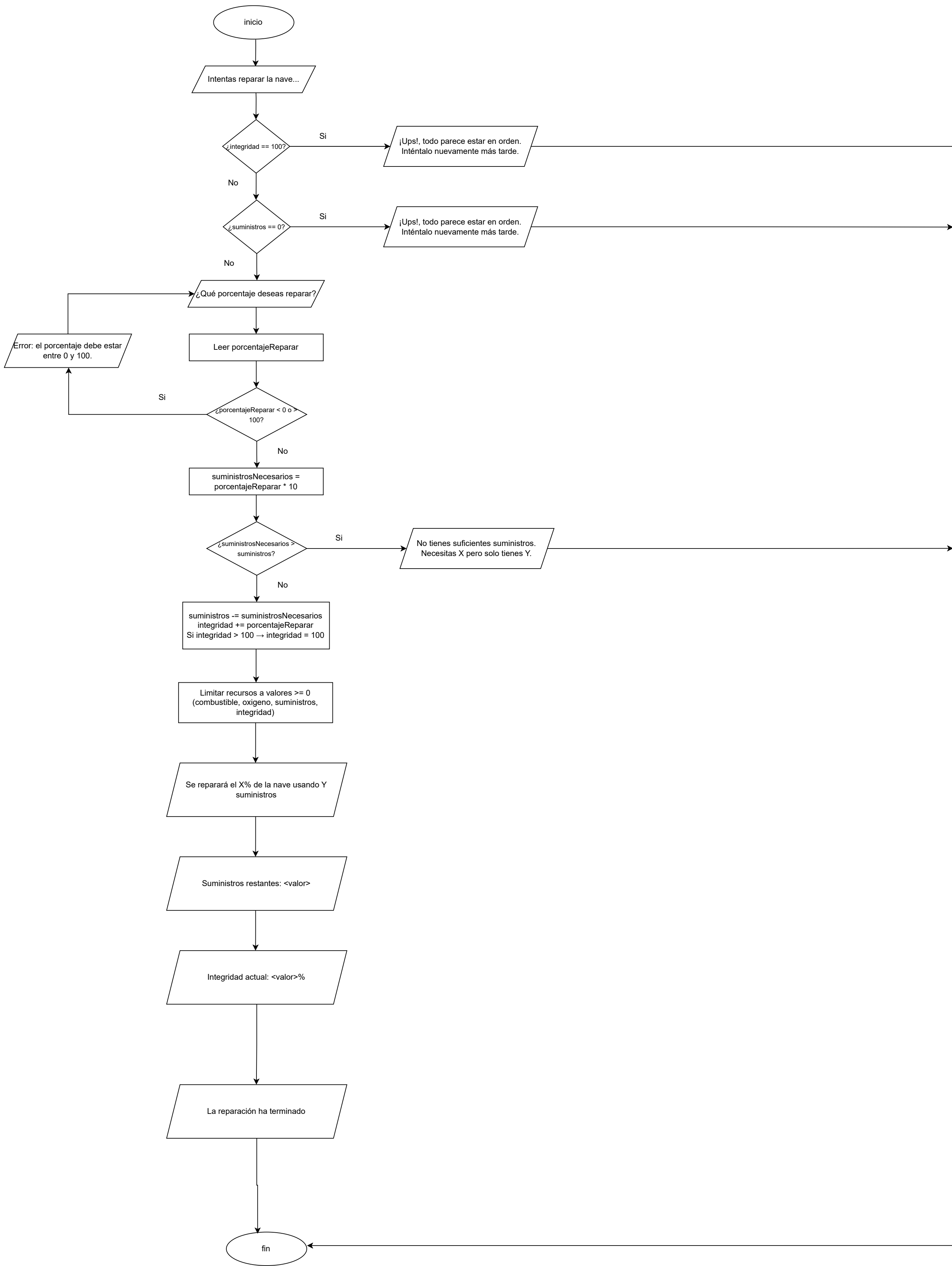
# MAIN



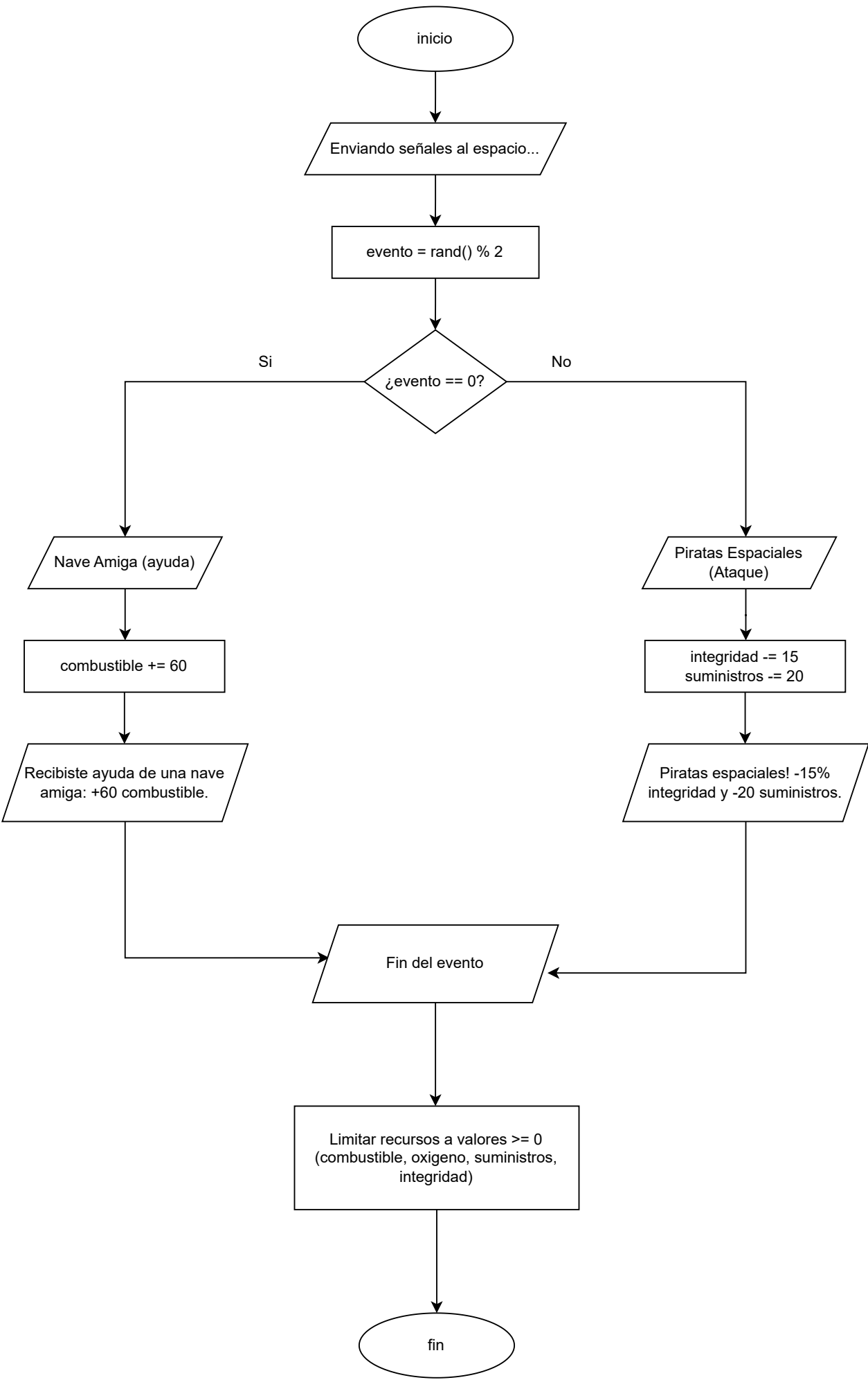
# MÉTODO EXPLORAR PLANETA



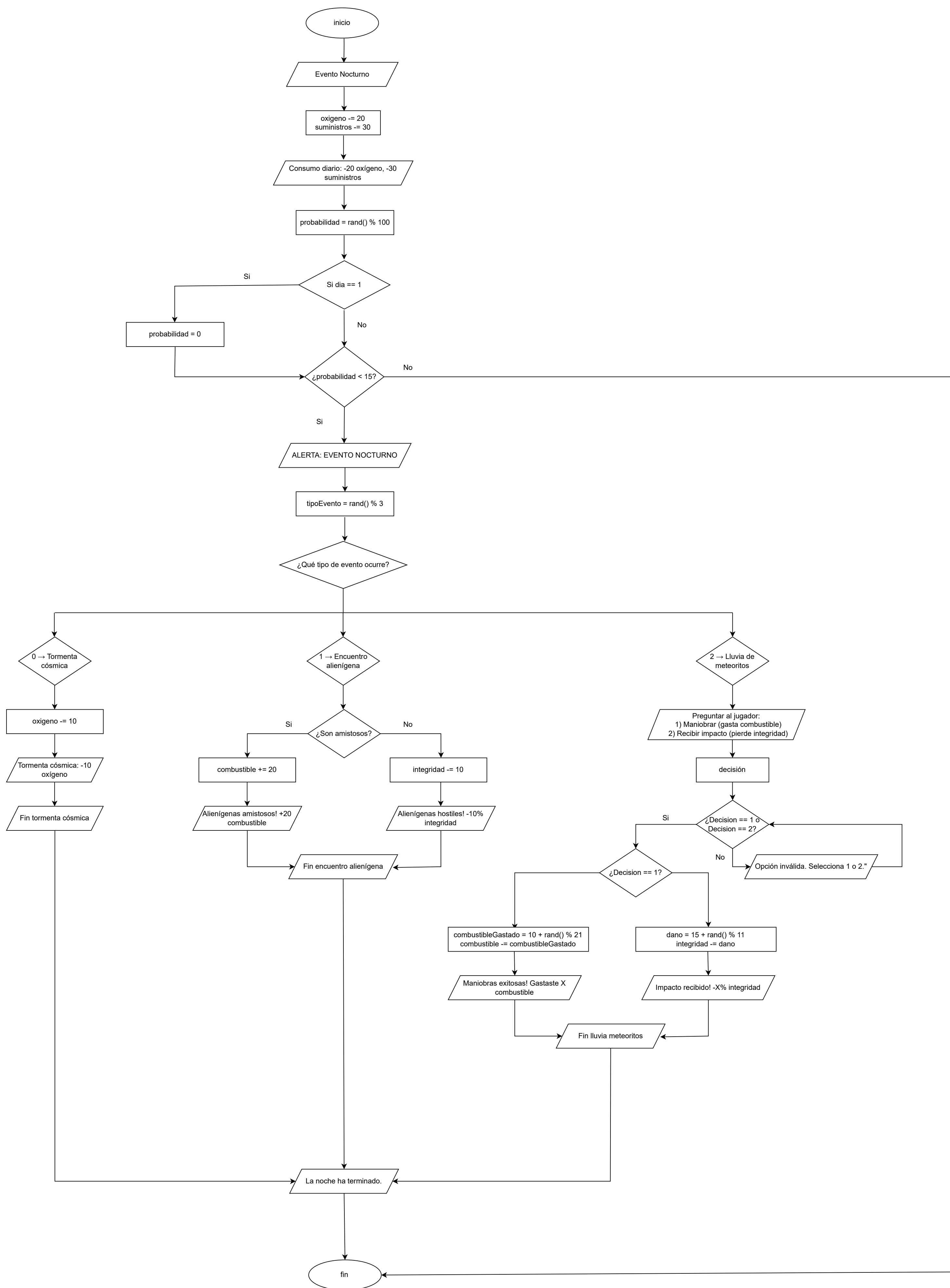
# MÉTODO REPARAR NAVE



# MÉTODO ENVIAR SEÑALES



# MÉTODO EVENTO NOCTURNO



# MÉTODO HACKER

