

Universidad Rafael Landívar.

Facultad De Ingeniería.

Licenciatura En Ingeniería En Informática Y Sistemas.

Laboratorio de Pensamiento Computacional, Sección 19.

Docente: Inga. Cindy García.

PROYECTO 01 (INCISO B)

“SIMULADOR DE VIAJE ESPACIAL – REGRESO A CASA”

Estudiante: Cucul Tut, Erik Carlos Omar

Carné: 1292625

Estudiante: Ajin Orellana, Davi`s Mauricio

Carné: 1270325

9 de abril de 2025

I. ACCIONES DEL PROGRAMA

El jugador debe elegir una de las siguientes acciones para intentar sobrevivir hasta llegar a su destino:

1. Explorar un planeta cercano:
 - **Costo:** -15 unidades de combustible.
 - Si no tiene suficiente, no puede explorar
 - **Eventos posibles:**
 - **Encontrar oxígeno** (20-40 unidades) → 60% de probabilidad.
 - **Encontrar combustible** (10-30 unidades) → 25% de probabilidad.
 - **Encontrar suministros** (30-100 unidades) → 50% de probabilidad.
 - **Tormenta eléctrica** (daño a la nave: -10% a -20%) → 25% de probabilidad.
 - **Aterrizaje forzado** (daño a la nave: -10% a -20%) → 25% de probabilidad.
2. Reparar la nave:
 - **Costo:** Se gastan suministros.
 - **Beneficio:** Recupera la integridad de la nave.
 - **Detalles:**
 - Cada 1% de integridad reparada consume 10 unidades de suministros.
 - El jugador elige cuánto reparar.
 - La integridad de la nave no puede superar el 100%.
 - Si el jugador pide más del máximo, se ajusta automáticamente.
3. Enviar señales al espacio
 - **Riesgo:** Puede atraer ayuda o peligro.
 - **Eventos posibles (50/50 de probabilidad):**
 - **Ayuda:** +60 unidades de combustible.
 - **Piratas espaciales:** -15% de integridad de la nave y -20 suministros.
4. Fin del Día – Consumo de recursos y eventos aleatorios.
 - **Consumo automático:**
 - Oxígeno: -20 unidades.
 - Suministros: -30 unidades.
 - **Evento aleatorio nocturno** (15% de probabilidad de ocurrir):
 - **Tormenta cósmica:** -10 unidades de oxígeno.
 - **Encuentro alienígena:** Puede ser amistoso (+20 de combustible) o hostil (-10% de integridad).
 - **Lluvia de meteoritos:**
 - Si el jugador maniobra, gasta entre 10-30 de combustible.
 - Si el jugador no maniobra, la nave recibe -15% a -25% de daño.

5. Fin del Juego

El juego termina cuando ocurre alguna de estas condiciones:

- Victoria:
 - El jugador sobrevive 10 días y llega a su destino.
- Derrota:
 - Se queda sin oxígeno, sin combustible o sin suministros.
 - La integridad de la nave llega a 0%.
 - El jugador decide rendirse.

II. DATOS DE ENTRADA

1. Selección de acción diaria:

Cada día, el jugador debe elegir entre las opciones disponibles:

- (1) Explorar un planeta.
- (2) Reparar la nave.
- (3) Enviar una señal.
- (4) Rendirse.
- Tipo de dato: Número entero (Int / Integer).

2. Cantidad de integridad a reparar:

Si el jugador elige reparar la nave, se le pedirá que indique cuántos puntos de integridad desea restaurar.

- La cantidad ingresada no puede hacer que la integridad supere el 100%.
- Tipo de dato: Número entero (Int / Integer).

3. Decisiones en eventos:

En eventos como la lluvia de meteoritos, el jugador debe decidir:

- Maniobrar.
- No maniobrar.
- Tipo de dato: Número entero (Int / Integer).

III. VARIABLES

1. Recursos del jugador:

- **Combustible:** Cantidad de combustible disponible para avanzar.
 - Tipo de dato: Número entero (Int / Integer).
 - Valor inicial: 30 unidades.
- **Oxígeno:** Cantidad de oxígeno disponible para sobrevivir.
 - Tipo de dato: Número entero (Int / Integer).
 - Valor inicial: 50 unidades.
- **Suministros:** Cantidad de alimentos y herramientas para reparaciones.
 - Tipo de dato: Número entero (Int / Integer).
 - Valor inicial: 40 unidades.

2. Estado de la nave:

- **Integridad de la nave:** Porcentaje de integridad de la nave.
 - Tipo de dato: Número entero (Int / Integer).
 - Valor inicial: 100%.

3. Contadores y control del juego:

- **Días transcurridos:** Número de días que han pasado en el viaje.
 - Tipo de dato: Número entero (Int / Integer).
 - Valor inicial: 0.
- **Días totales:** Número máximo de días para llegar al destino.
 - Tipo de dato: Número entero (Int / Integer).
 - Valor inicial: 10.

4. Generación de eventos aleatorios:

- **Random:** Objeto para generar números aleatorios.
 - Tipo de dato: Random.
 - Valor inicial: Se inicializa una vez al inicio del programa.

5. Variables de Acciones y Eventos:

- **Accionvalida:** Controla si la acción elegida es válida.
 - Tipo de dato: bool.
 - Valor inicial: false.
- **Accion:** Almacena la opción del menú elegida.
 - Tipo de dato: string.
 - Valor inicial: null.
- **Entrada:** Guarda el input temporal.
 - Tipo de dato: string.
 - Valor inicial: null.
- **Porcentaje:** Porcentaje de reparación ingresado.
 - Tipo de dato: int.
 - Valor inicial: 0.
- **Maxreparable:** Máximo porcentaje reparable.
 - Tipo de dato: int.
 - Valor inicial: 0.
- **Costo:** Costo en suministros para reparar.
 - Tipo de dato: int.
 - Valor inicial: 0.
- **Oxi:** Recurso de oxígeno encontrado al explorar.
 - Tipo de dato: int.
 - Valor inicial: 0.
- **Comb:** Recurso de combustible encontrado al explorar.
 - Tipo de dato: int.
 - Valor inicial: 0.
- **Sum:** Recurso de suministro encontrado al explorar.
 - Tipo de dato: int.
 - Valor inicial: 0.
- **Daño:** Daño recibido en eventos.
 - Tipo de dato: int.
 - Valor inicial: 0.
- **Evento:** Tipo de evento nocturno.
 - Tipo de dato: int.
 - Valor inicial: 0.
- **Opción:** Elección durante meteoritos.
 - Tipo de dato: string.
 - Valor inicial: null.
- **Gasto:** Combustible gastado al esquivar meteoritos.
 - Tipo de dato: 0.
 - Valor inicial: false.

IV. CONDICIONES Y CALCULOS

1. Cantidades iniciales:

- Combustible: 30 unidades.
 - Fórmula: combustible = 30.
- Oxígeno: 50 unidades.
 - Fórmula: oxigeno = 50.
- Suministros: 40 unidades.
 - Fórmula: suministros = 40.
- Integridad de la nave: 100%.
 - Fórmula: integridadNave = 100.
- Días transcurridos: 0.
 - Fórmula: diasTranscurridos = 0.
- Días totales: 10.
 - Fórmula: diasTotales = 10.

2. Consumo diario de recursos:

- Oxígeno:
 - Formula: oxigeno -= 20.
- Suministros:
 - Formula: suministros -= 30.

3. Acciones diarias:

- **Explorar un planeta:**
 - Gasta 15 unidades de combustible.
 - Formula: combustible -= 15
 - Eventos aleatorios:
 - Encontrar oxígeno:
 - Formula: oxigeno += random.Next(20, 41).
 - Encontrar combustible:
 - Formula: combustible += random.Next(10, 31).
 - Encontrar suministros:
 - Formula: suministros += random.Next(30, 101).
 - Tormenta eléctrica:
 - Formula: integridadNave -= random.Next(10, 21).
 - Aterrizaje forzado:
 - Formula: integridadNave -= random.Next(10, 21).

- **Reparar la nave:**
 - Gasta suministros:
 - Formula: suministros -= (porcentaje * 10).
 - Recupera integridad:
 - Formula: integridadNave += porcentajeReparar.
 - La integridad no puede superar el 100%.
 - Formula: maxReparable = 100 - integridadNave
 - Formula: porcentajeReparar > maxReparable
 - Formula: porcentajeReparar = maxReparable
- **Enviar señales:**
 - Ayuda con combustible:
 - Formula: combustible += 60.
 - Piratas espaciales:
 - Formula: integridadNave -= 15.
 - Formula: suministros -= 20.
- **Rendirse**
Termina el juego inmediatamente.

4. Eventos nocturnos:

- **Tormenta cósmica:**
 - Formula: oxigeno -= 10.
- **Encuentro alienígena:**
 - Amistoso:
 - Formula: combustible += 20.
 - Hostil:
 - Formula: integridadNave -= 10.
- **Meteoritos:**
 - Maniobrar:
 - Formula: combustible -= random.Next(10, 31).
 - Recibir impacto:
 - Formula: integridadNave -= random.Next(15, 26).

5. Condiciones de fin del juego:

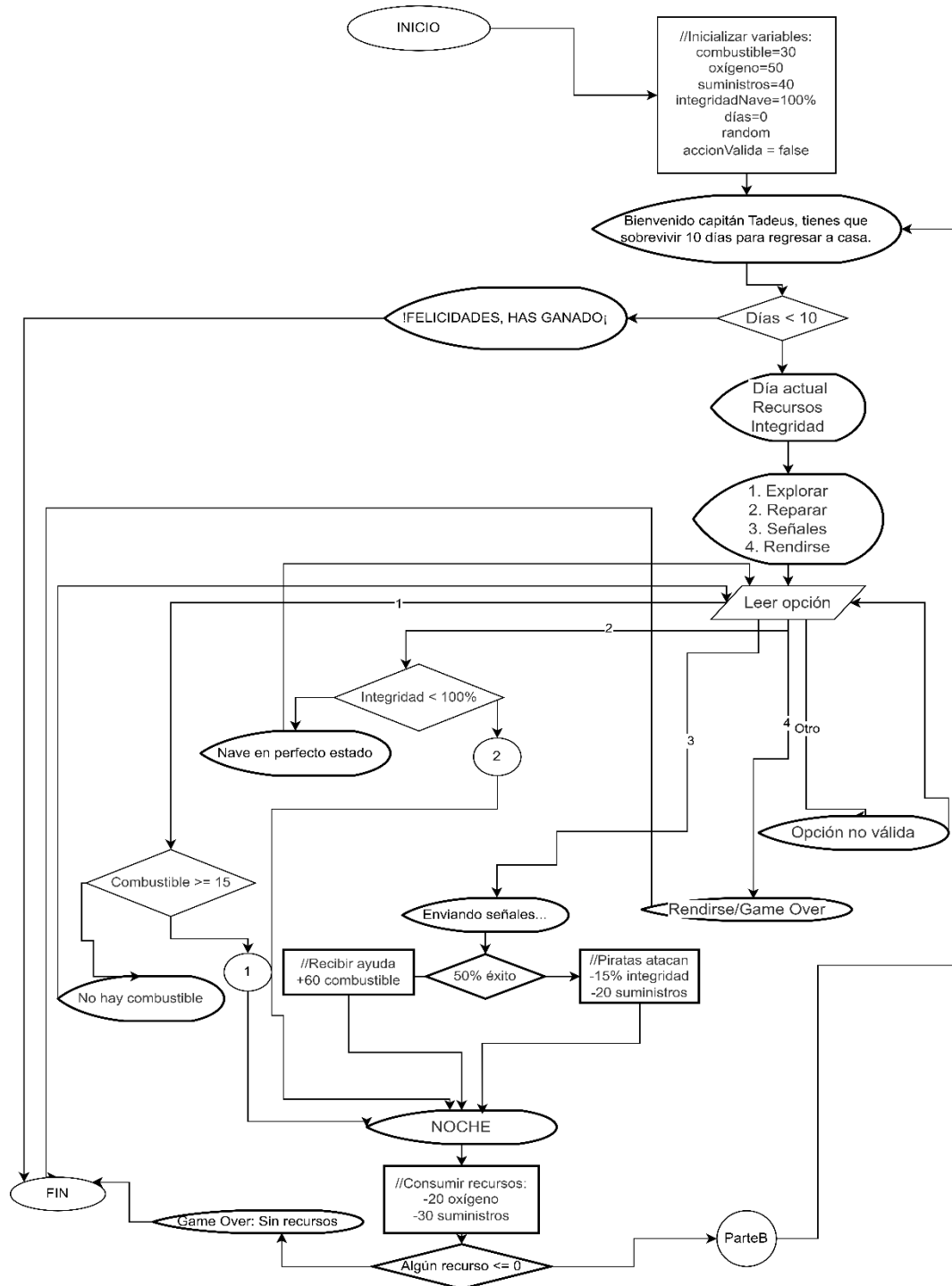
- Victoria: `diasTranscurridos == 10`.
- Derrota:
 - `oxigeno <= 0`.
 - `combustible <= 0`.
 - `suministros <= 0`.
 - `integridadNave <= 0`.
- Rendición: El jugador decide elegir la opción 4.

V. ALGORITMO

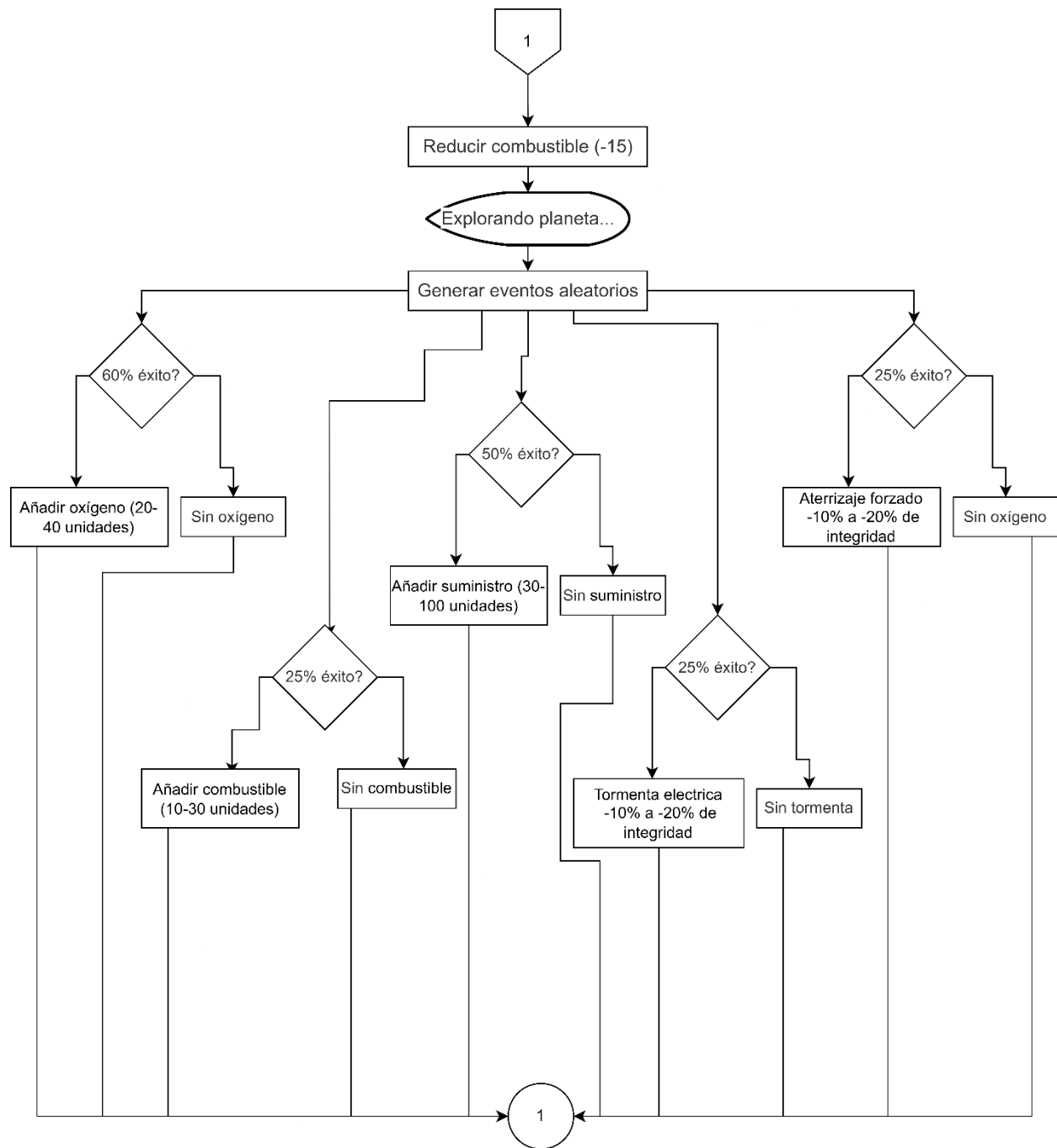
Enlace: <https://drive.google.com/file/d/1-HToq6qntjHyocE6UsC947nAbUXgQSVF/view?usp=sharing>

Diagrama de flujo:

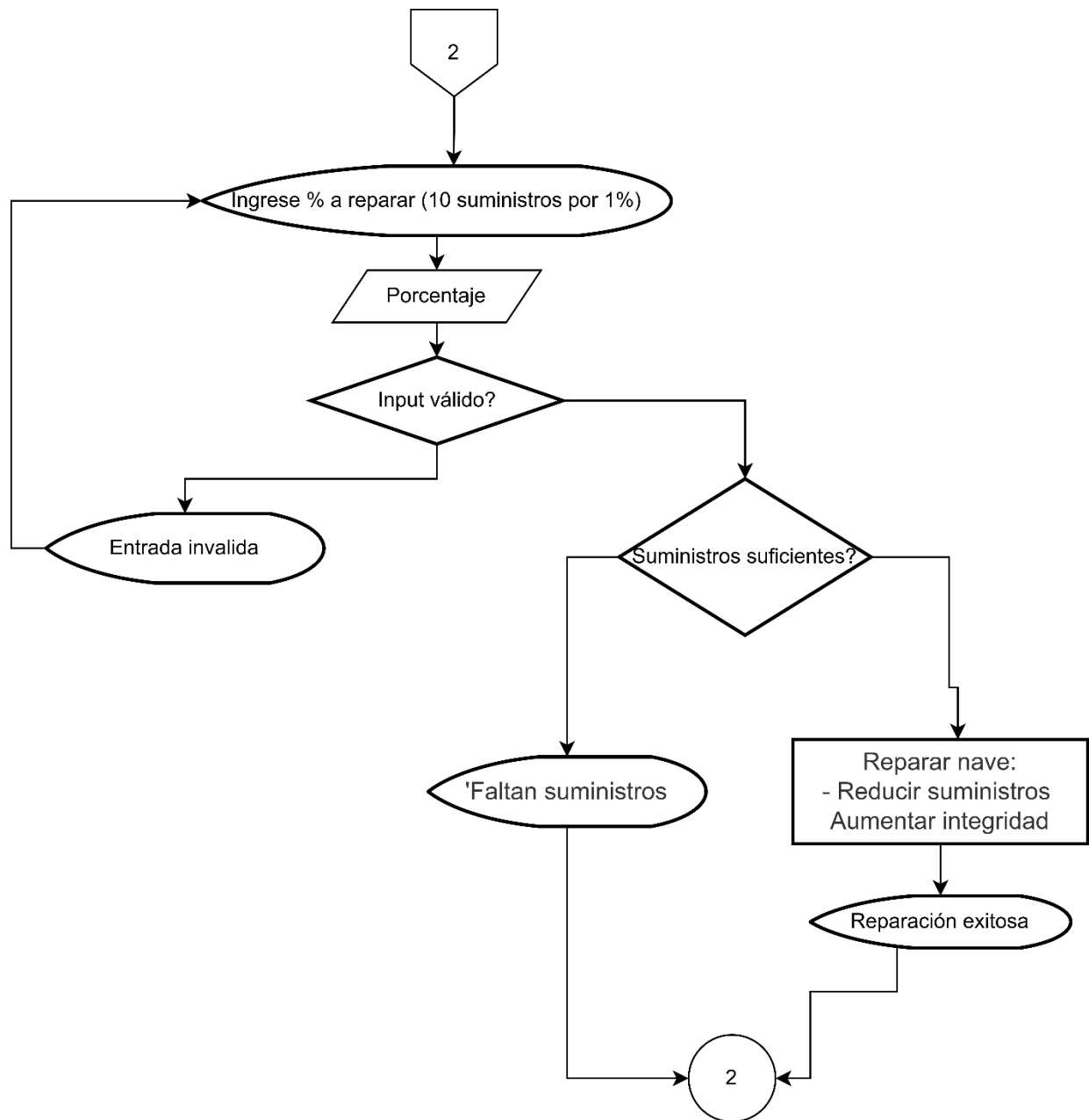
Parte A:



1:



2:



ParteB:

