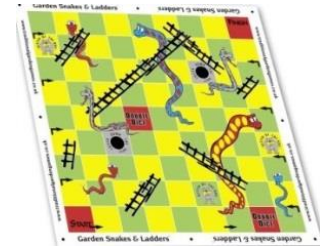


Parcial II Término 2005-2006. Diciembre 06, 2005

Tema 4 (25 puntos) Para una nueva versión del juego “Escaleras y Serpientes” se desea disponer del algoritmo para simulación en computador. El juego para dos jugadores consiste en llegar o pasar la meta en primer lugar en un tablero de 64 casillas cuyas especificaciones son las siguientes:



1. Al inicio los jugadores están en una misma posición y arrancan su trayectoria cuando lanzando una moneda (cara 1 o 2) el jugador que gane empieza.
2. Cada jugador realiza su recorrido alternadamente de acuerdo a los resultados de los lanzamientos de un dado (6 caras)
3. Al avanzar, el jugador puede “caer” en una “casilla de castigo”, por lo que retrocederá 3 pasos de la posición en la que se encuentra. Si cae en “casilla de premio”, el usuario avanzará 3 pasos de la posición en la que se encuentra.
4. Luego de un lanzamiento y determinación de la posición final, el jugador le pasa el turno al otro jugador.
5. Se repite el juego desde el paso 2 hasta que uno de los jugadores pase la meta.

Al final se deberá mostrar:

- Número de veces jugadas por cada jugador, y
- El jugador que ganó.

Nota: casillas de premio: 4, 9, 29, 34, 46 y de castigo: 8, 19, 38, 50, 60

Rubrica: Manejo de jugadores y aleatorios (10 puntos), asignación de premios o castigos (5 puntos), contador de turnos (5 puntos), determinar ganador (5 puntos)

Propuesta de Solución:

En un primer bosquejo, no se considera el lanzar la moneda para ver cuál jugador inicia, comparable al ajedrez en que las fichas blancas salen primero, sin considerar el nombre del jugador.

Para determinar las posiciones de las fichas en el tablero, usarán acumuladores para cada jugador A y B, usando como variables “fichaa” y “fichab”.

Serán necesarios otras variables como el contador para el turno, cada jugador dispondrá de una variable generada de forma aleatoria que simula un dado de 6 caras.

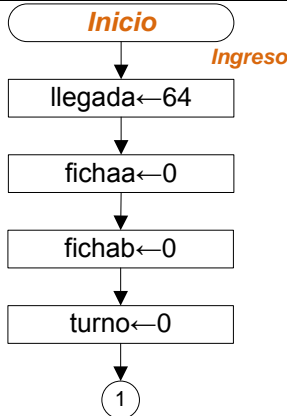
Se lanza el dado para el jugador A, se avanza acumulando conforme a los puntos del dado, se valida el premio o castigo para luego repetir el proceso para el jugador B, repitiendo el proceso hasta que alguna ficha se ubique o pase la casilla de “llegada”

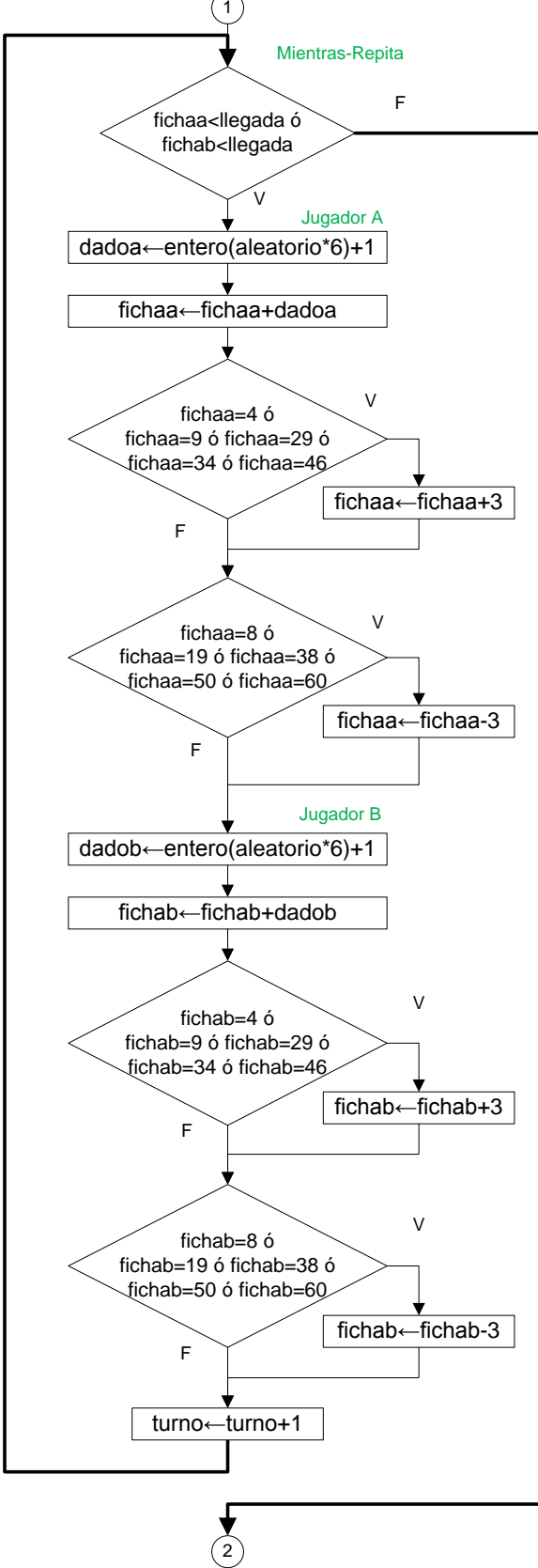
Terminado el proceso anterior, se determina cuál jugador ganó, observe que se supone que gana tiene inicialmente el valor de 0, para tener la opción de registrar un empate.

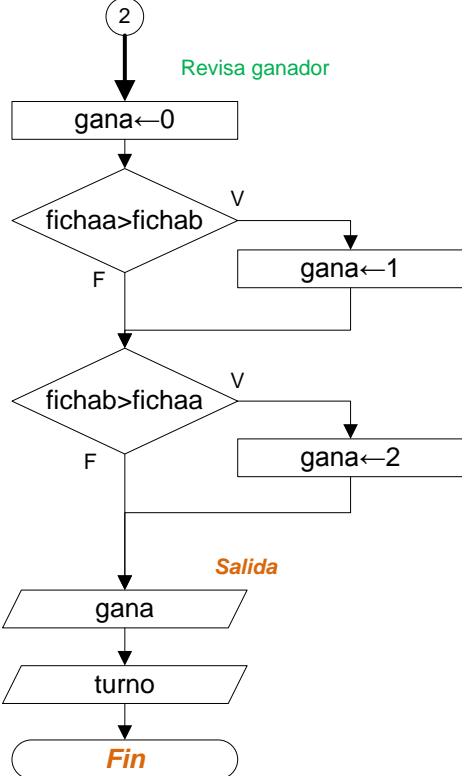
Al final se muestran los resultados del juego simulado.

Primero se presenta la solución usando lazo “Mientras-Repita” por compatibilidad con Matlab, luego se muestra la versión con el lazo “Repita-Hasta” que se puede escribir en los lenguajes de programación.

Tarea: Resolver con el lanzamiento de la moneda para ver cuál inicia.

Descripción	DIAGRAMA DE FLUJO	MATLAB /OCTAVE	
Inicio	 <pre> graph TD Inicio([Inicio]) -- Ingreso --> llegada[llegada ← 64] llegada --> fichaa[fichaa ← 0] fichaa --> fichab[fichab ← 0] fichab --> turno[turno ← 0] turno --> 1((1)) </pre>	<pre> % ICM00794-Fundamentos de Computación- FCNM-ESPOL % Propuesta de solución.edelros@espol.edu.ec. % Parcial II Termino 2005. Tema 4 % Juego de Escaleras y Serpientes llegada=64; fichaa=0; fichab=0; turno=0; </pre>	
Tamaño del tablero			
Posición jugador A			
Posición jugador B			
Turno inicial			

Descripción	DIAGRAMA DE FLUJO	MATLAB /OCTAVE
<p>Mientras jugador A no pasa casilla de llegada o jugador B no pasa casilla de llegada</p> <p>Lanza el dado</p> <p>Avanza jugador A</p> <p>Valida si hay Premio</p> <p>Valida si hay castigo</p> <p>Lanza el dado jugador B</p> <p>Avanza jugador B</p> <p>Valida premio</p> <p>Valida Castigo</p> <p>Cuenta turno</p> <p>Repita</p>		<pre> while (fichaa < llegada fichab < llegada) % jugador A dadoa = fix(rand*6)+1; fichaa = fichaa + dadoa; if (fichaa == 4 fichaa == 9 fichaa == 29 fichaa == 34 fichaa == 46) fichaa = fichaa + 3; end if (fichaa == 8 fichaa == 19 fichaa == 38 fichaa == 50 fichaa == 60) fichaa = fichaa - 3; end % jugador B dadob = fix(rand*6)+1; fichab = fichab + dadob; if (fichab == 4 fichab == 9 fichab == 29 fichab == 34 fichab == 46) fichab = fichab + 3; end if (fichab == 8 fichab == 19 fichab == 38 fichab == 50 fichab == 60) fichab = fichab - 3; end turno = turno + 1; end </pre>

Descripción	DIAGRAMA DE FLUJO	MATLAB /OCTAVE
<p>Considera Empate</p> <p>Caso en que gana A</p> <p>Caso en que gana B</p> <p>Muestra ganador</p> <p>Muestra turnos jugados</p> <p>Fin</p>	 <pre> graph TD Start((2)) --> Revisa[Revisa ganador] Revisa --> G0[gana ← 0] G0 --> D1{fichaa > fichab} D1 -- V --> G1[gana ← 1] D1 -- F --> D2{fichab > fichaa} D2 -- V --> G2[gana ← 2] G1 --> Out1[/gana/] G2 --> Out1 Out1 --> Out2[/turno/] Out2 --> End([Fin]) </pre>	<pre> gana=0; if fichaa>fichab gana=1; end if fichab>fichaa gana=2; end disp('gana el jugador: '); disp(gana); disp('turnos jugados:'); disp(turno); </pre>

Ejecución del algoritmo: escaleraserpiente.m

<pre> >> escaleraserpiente gana el jugador: 1 turnos jugados: 21 </pre>	<pre> >> escaleraserpiente gana el jugador: 2 turnos jugados: 20 </pre>
---	---

Otra Solución usando lazo “Repita-Hasta” (do-until), observe que solo cambia la forma del lazo para repetir. Ésta forma de lazo no es soportada por matlab, sin embargo en otros lenguajes se puede escribir en la lógica de “Repita – Hasta”.

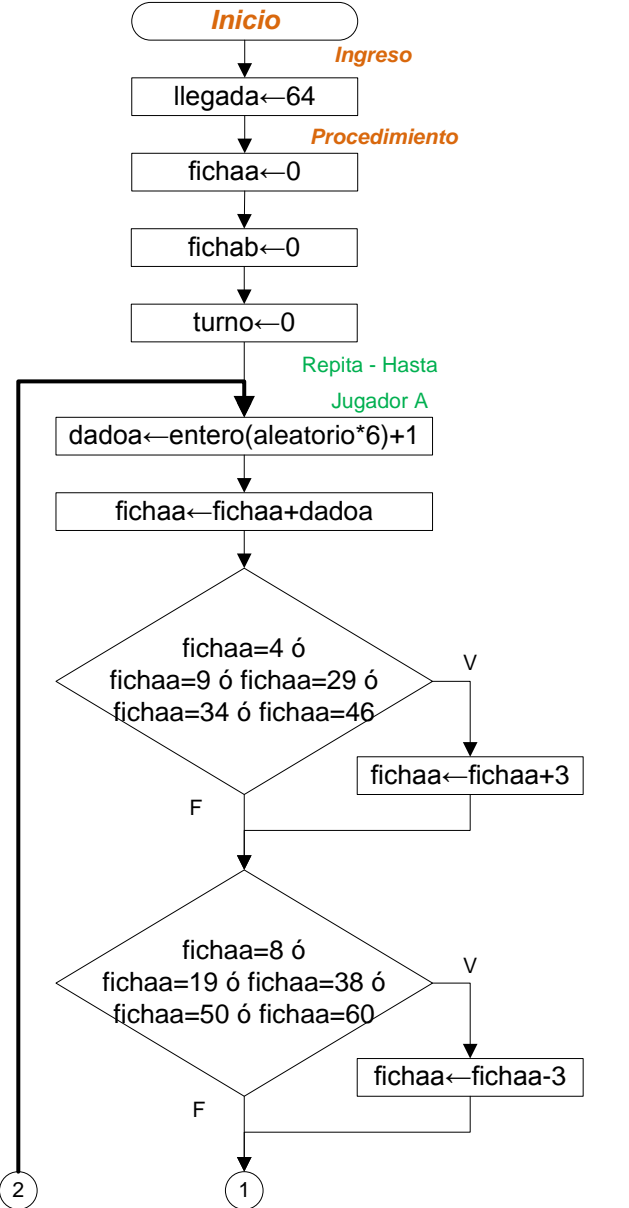
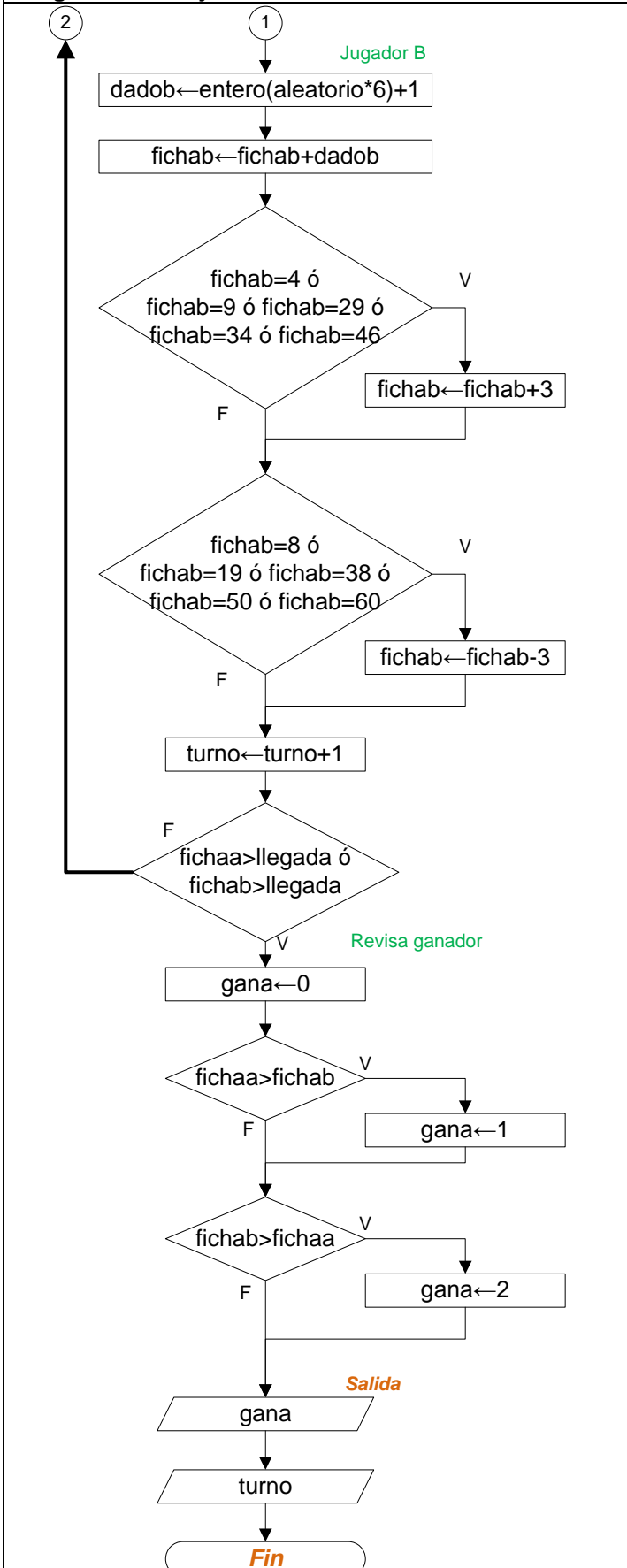
Diagrama de Flujo	OCTAVE
 <pre> graph TD Inicio([Inicio]) --> Ingreso[llegada ← 64] Ingreso --> Procedimiento[fichaa ← 0 fichab ← 0 turno ← 0] Procedimiento --> RepitaHasta[Repita - Hasta Jugador A] RepitaHasta --> Dadoa[dadoa ← entero(aleatorio*6)+1] Dadoa --> FichaaSum[fichaa ← fichaa+dadoa] FichaaSum --> Cond1{fichaa=4 ó fichaa=9 ó fichaa=29 ó fichaa=34 ó fichaa=46} Cond1 -- V --> FichaaPlus3[fichaa ← fichaa+3] Cond1 -- F --> Cond2{fichaa=8 ó fichaa=19 ó fichaa=38 ó fichaa=50 ó fichaa=60} Cond2 -- V --> FichaaMinus3[fichaa ← fichaa-3] Cond2 -- F --> RepitaHasta FichaaPlus3 --> Cond1 FichaaMinus3 --> Cond2 </pre>	<pre> % ICM00794-Fundamentos de Computación - FCNM-ESPOL % Propuesta de solución. edelros@espol.edu.ec. % Parcial II Termino 2005. Tema 4 % Juego de Escaleras y Serpientes llegada=64; fichaa=0; fichab=0; turno=0; do % jugador A dadoa=fix(rand*6)+1; fichaa=fichaa+dadoa; if (fichaa==4 fichaa==9 fichaa==29 fichaa==34 fichaa==46) fichaa=fichaa+3; end if (fichaa==8 fichaa==19 fichaa==38 fichaa==50 fichaa==60) fichaa=fichaa-3; end </pre>

Diagrama de Flujo	OCTAVE
 <pre> graph TD Start((1)) --> Roll[dadob ← entero(aleatorio*6)+1] Roll --> Add[fichab ← fichab+dadob] Add --> Win1{fichab=4 ó fichab=9 ó fichab=29 ó fichab=34 ó fichab=46} Win1 -- V --> Add3[fichab ← fichab+3] Win1 -- F --> Win2{fichab=8 ó fichab=19 ó fichab=38 ó fichab=50 ó fichab=60} Win2 -- V --> Sub3[fichab ← fichab-3] Win2 -- F --> Turno[turno ← turno+1] Add3 --> Turno Sub3 --> Turno Turno --> Reached{fichaa > llegada ó fichab > llegada} Reached -- F --> Start Reached -- V --> Revisa[Revisa ganador] Revisa --> Gana0[gana ← 0] Gana0 --> Win3{fichaa > fichab} Win3 -- V --> Gana1[gana ← 1] Win3 -- F --> Win4{fichab > fichaa} Win4 -- V --> Gana2[gana ← 2] Win4 -- F --> Output[/gana/] Output --> TurnoOut[/turno/] TurnoOut --> End([Fin]) </pre>	<pre> %jugador B dadob=fix(rand*6)+1; fichab=fichab+dadob; if (fichab==4 fichab==9 fichab==29 fichab==34 fichab==46) fichab=fichab+3; end if (fichab==8 fichab==19 fichab==38 fichab==50 fichab==60) fichab=fichab-3; end turno=turno+1; until(fichaa>=llegada fichab>=llegada) gana=0; if fichaa>fichab gana=1; end if fichab>fichaa gana=2; end disp('gana el jugador: '); disp(gana); disp('turnos jugados:'); disp(turno); </pre>