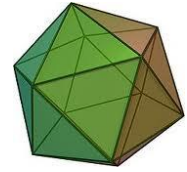


Parcial II Término 2003-2004. Diciembre 09, 2003

Tema 4 (30 puntos). Se requiere implementar un juego por computadora que consiste en generar aleatoriamente el lanzamiento de 2 **icosaedros** (poliedro regular de 20 caras triangulares).

Las caras están identificadas por un **color** (azul, blanco, rojo o negro) y un **número** entero (1, 2, 3, 4 o 5).



Una vez que se han detenidos los 2 icosaedros (luego de la simulación de los lanzamientos), considere las siguientes Reglas para el juego:

Se observan las caras de la base:

- Si coinciden los colores de las bases, el jugador gana 10 centavos.
- Si coinciden los números de las bases, el jugador gana 10 centavos.
- Si coinciden los colores y los números de las bases, el jugador gana 50 centavos.
- Si la suma de los números de las bases es impar, el jugador gana 5 centavos más.

Para iniciar el juego, se debe presionar el número **1**. Para seguir jugando se debe presionar el número **2** y para terminar el juego se debe presionar el número **3**. Al final del juego se deberá mostrar el **Total Pagado** al Jugador y la **Cantidad de Lanzamientos**.

A continuación se muestra una ejecución en pantalla del Algoritmo que Ud. debe construir:

Presione 1 para iniciar el juego:1 Icosaedro 1: 2 de color rojo Icosaedro 2: 4 de color rojo Jugador GANO 10 centavos Presione 2 para lanzar, 3 para salir: 2 Icosaedro 1: 3 de color azul Icosaedro 2: 3 de color negro Jugador GANO 10 centavos	Presione 2 para lanzar, 3 para salir: 2 Icosaedro 1: 4 de color blanco Icosaedro 2: 4 de color blanco Jugador GANO 50 centavos Presione 2 para lanzar, 3 para salir: 2 Icosaedro 1: 3 de color negro Icosaedro 2: 4 de color negro Jugador GANO 15 centavos	Presione 2 para lanzar, 3 para salir: 3 El jugador GANO 85 centavos en 4 Lanzamientos
--	--	--

Rubrica: manejo de icosaedros (10 puntos), implementar reglas del juego (10 puntos), respuestas (5 puntos, control del juego (5 puntos)

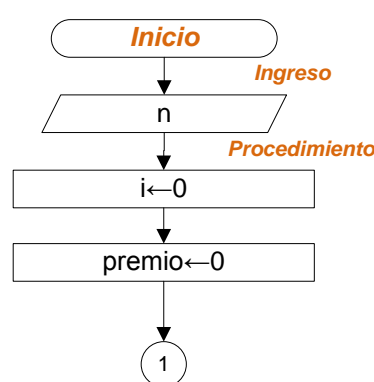
Propuesta de Solución:

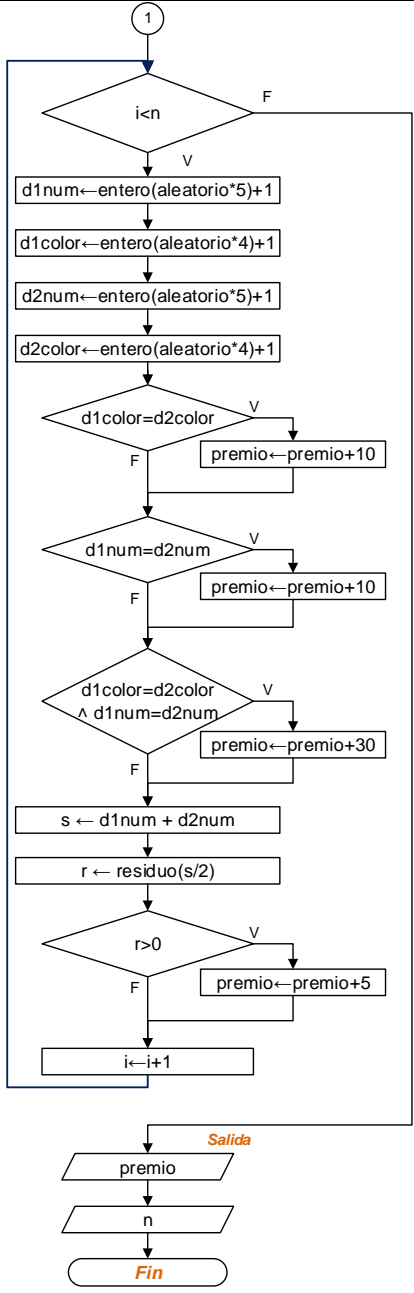
Se propone desarrollar el tema conociendo el número **n** de lanzamientos que desea participar el jugador, **dejando el control del juego como tarea**.

Se inicia un contador de lanzamientos y un acumulador de premios. Un icosaedro se simula con un aleatorio para el color y otro para el número para luego comparar éstos valores con lo simulado para el segundo icosaedro.

Las reglas descritas se implementan con condicionales, considerando que de cumplirse la tercera, ya se ha pagado premio por igualdad de números y color.

Se repite el juego hasta cumplir el número de lanzamientos.

Descripción	DIAGRAMA DE FLUJO	Python
Inicio Ingresa la cantidad de lanzamientos Contador de lanzamientos ejecutados Total de premios ganados	 <pre> graph TD Inicio([Inicio]) -- Ingreso --> n[/n/] n -- Procedimiento --> i["i ← 0"] i --> premio["premio ← 0"] premio --> End((1)) </pre>	<pre> # ICM00794-Fund.Computación - FCNM-ESPOL # Parcial II Término 2003. Tema 4. Juego Icosaedros # Propuesta solucion: edelros@espol.edu.ec # Tarea. Completar el control del juego import random n=int(input('Numero de lanzamientos: ')) #Procedimiento i=0 premio=0 </pre>

<p>Mientras queden lanzamientos</p> <p>Número del icosaedro 1</p> <p>Color del icosaedro 1</p> <p>Número del icosaedro 2</p> <p>Color del icosaedro 2</p> <p>Si los colores son iguales, se gana 10 centavos</p> <p>Si los números son iguales, se gana 10 centavos</p> <p>Si los colores y números son iguales, se completan los 50 centavos</p> <p>Se suman los números</p> <p>Para verificar si son impares</p> <p>Si son impares, se gana 5 centavos adicionales</p> <p>Cuenta un lanzamiento de icosaedros Repita</p> <p>Muestra el premio ganado</p> <p>Muestra el total de lanzamientos</p> <p>Fin</p>	 <pre> graph TD Start((1)) --> Cond1{i < n} Cond1 -- V --> D1num[d1num ← entero(aleatorio*5)+1] D1num --> D1color[d1color ← entero(aleatorio*4)+1] D1color --> D2num[d2num ← entero(aleatorio*5)+1] D2num --> D2color[d2color ← entero(aleatorio*4)+1] D2color --> Cond2{d1color = d2color} Cond2 -- V --> P10_1[premio ← premio+10] P10_1 --> Cond3{d1num = d2num} Cond3 -- V --> P10_2[premio ← premio+10] P10_2 --> Cond4{d1color = d2color & d1num = d2num} Cond4 -- V --> P30[premio ← premio+30] P30 --> Sum[s ← d1num + d2num] Sum --> Resid[r ← residuo(s/2)] Resid --> Cond5{r > 0} Cond5 -- V --> P5[premio ← premio+5] P5 --> IncI[i ← i+1] IncI --> Cond1 Cond1 -- F --> Salida[Salida] Salida --> P1[/premio/] P1 --> P2[/n/] P2 --> Fin([Fin]) </pre>	<pre> while (i<n): d1num=int(random.random()*5)+1 d1color=int(random.random()*4)+1 d2num=int(random.random()*5)+1 d2color=int(random.random()*4)+1 if d1color==d2color: premio=premio+10 if d1num==d2num: premio=premio+10 if ((d1color==d2color)and(d1num==d2num)): premio=premio+30 s=d1num+d2num r=s%2 if r>0: premio=premio+5 i=i+1 #Salida print('total ganado: ') print(premio) print('lanzamientos: ') print(n) </pre>
---	--	--

Ejecución del algoritmo: icosaedros.py

<pre> >>> Numero de lanzamientos: 5 total ganado: 40 lanzamientos: 5 </pre>	<pre> >>> Numero de lanzamientos: 15 total ganado: 160 lanzamientos: 15 </pre>
--	---