

Bowling Rules



Rules:

Existen 10 turnos por jugador (**frames**)

En cada turno hay dos oportunidades para tumbar todos los pinos (**rolls**)

Existen 10 pinos (**pins**)

Puntaje:

Si tumbamos todos los pinos en el primer roll un **STRIKE** y se marca con una “**X**”. **El valor de ese FRAME 10 pts + los puntos de los siguientes dos rolls consecutivos.**

Si tumbamos todos los pinos en los dos rolls de un frame, es un **SPARE** que se marca con “**/**”. Cuando ocurre esto El valor de ese frame **se suma 10 puntos + los puntos del siguiente roll.**

Cuando no se tumban pinos se marca “**-**” y significa **0** puntos

Cuando en un roll el jugador comete una falta y pisa la línea acumula 0 pts en el roll y se marca en el tablero con “**F**”

Rules:

Si tumbamos menos de 10 puntos en un frame se suma la cantidad de puntos:

Ex

Frame:

4 5 9 pts

5 F 5 pts

9 - 9 pts

3 2 5 pts

En el roll 1 se tumbo 4 pinos y Segundo roll se tumbo 5 pinos

10th Frame:

Este es algo diferente ya que te permite jugar 2 Rolls como los demás frames. Sin embargo, se puede jugar hasta 3 Rolls consecutivos.

- If you roll a spare, you are awarded one extra roll.
- If you roll a strike, you are awarded two extra rolls

X X X

4 / 5

- / 3

X 4 6

Frame 10pts + 10 pts + 10 pts

Frame 4pts + 10 + 5

Frame 0pts + 10pts + 3pts

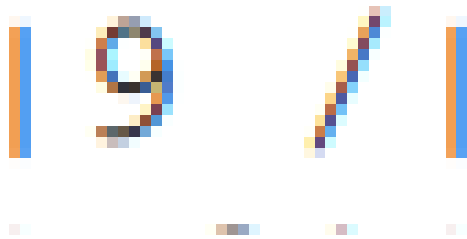
Let's make an example. The following game is provided and we need to compute the scores frame by frame:

#1	#2	#3	#4	#5	#6	#7	#8	#9	#10
+-----+									
5 4 7 - 9 / 5 / 8 - X 5 / X X 4 / X									
9 16 31 49 57 77 97 121 141 161									
+-----+									

Frame 1 (F1): 5 pts + 4 pts // total frame: 9 pts // cumulate game points: 9 pts
 (F2): 7 pts + 0 pts // total frame: 7 pts // cumulate game points: 16 pts
 (F3): 9 pts + 1pts + 5 pts // total frame: 15 pts // cumulate game points: 31 pts
 (F4): 5 pts + 5 pts + 8 pts // total frame: 18 pts // cumulate game points: 49 pts
 (F5): 8 pts + 0 pts // total frame: 8 pts // cumulate game points: 57 pts
 (F6): 10 pts + 5 pts + 5 pts // total frame: 20 pts // cumulate game points: 77 pts
 (F7): 5 pts + 5 pts + 10pts // total frame: 20 pts // cumulate game points: 97 pts
 (F8): 10 pts + 10 pts +4pts // total frame: 24 pts // cumulate game points: 121 pts
 (F9): 10 pts + 4pts + 6pts // total frame: 20 pts // cumulate game points: 141 pts
 (F10): 4 pts + 6 pts + 10pts // total frame: 20 pts // cumulate game points: 161 pts

Truco matematico Spare:

Frame n (Fn): 9 /



First roll = 9

Second roll = 10 – Previous Roll

$$\begin{aligned}\text{Total Frame Points} &= \text{First Roll} + (10 - \text{PrevisRll}) + \text{Next Roll} \\ &= 9 + (10 - 9) + 5 \\ &= 9 + 1 + 5 = 15\end{aligned}$$

Restricciones:

Additional information

You are free to choose whichever method you want to implement your program. You can add any helper functions you need and may even edit the structs in `tpb.c` (as you think more appropriate) with the following restrictions:

- you are not allowed to use global variables
 - all dynamically allocated memory should be freed
 - the function `bowling_score_parser` must return a linked list of `Frame` structs.
 - the function `print_scoreboard` must take a linked list of `Frame` structs as parameter.
- The last two constraints derive from the fact you are not allowed to change `tpb.h`.

As stated above, you can use any standard library functions (except the maths library).