

Práctica 2 de Programación 1

José Castro y Jesús Pérez

Octubre, 2022

Fechas válidas de entrega: desde el 10/10/2022 hasta el 17/10/2022. Su entrega consiste de un archivo llamado `jugador.c`. Por favor, colocar en comentario al inicio de ese archivo, su número de cédula, nombre y apellido.

1 Introducción

La práctica consiste en realizar un programa para determinar si un jugador es transferible de equipo y también comparar su valor con otros jugadores del mismo equipo. Para ello, debe implementar las funciones y procedimientos que se especifican en la sección 4.

Para conocer la información de cada jugador es necesario extraer los datos que están disponibles en [1].

2 Práctica

Para la realización de esta práctica, debe hacer uso de lo visto en clases, sin añadir ninguna otra biblioteca adicional a “`stdio.h`” y sin añadir contenido adicional a lo explicado en el curso hasta los momentos. Además, debe aplicar las reglas del estilo de código de PR1-ULA [2].

2.1 Casos de prueba:

Para los casos de prueba es necesario recibir los datos de dos jugadores del equipo, los cuales pueden ser ingresados al menos de dos maneras: la primera, de forma manual; la segunda, a través de un archivo llamado `datos.txt` que contenga la información extraída de [1] y ordenada como aparece en los inputs (ver los siguientes ejemplos), para luego ejecutar `./jugador.c < datos.txt`

Input:

Lebron
James
37
56
3.5
2.2
30.3
Kent
Bazemore
32
39
0.5
1.8
3.4

Output:

Jugador: Lebron James
Edad: 37
Estadísticas: 56 - 30.3 - 2.2 - 3.5
Valor: 99.0
Estado: no transferible

Jugador: Kent Bazemore
Edad: 32
Estadísticas: 39 - 0.5 - 1.8 - 3.4
Valor: 36.3
Estado: transferible

Tiene menor valor: Bazemore, Kent

Input:

Anthony
Carmelo
37
69
0.9
2.4
13.3
Russell
Westbrook
33
78
3.8
3.0
18.5

Output:

Jugador: Anthony Carmelo
Edad: 37
Estadísticas: 69 - 0.9 - 2.4 - 13.3
Valor: 69.0
Estado: no transferible

Jugador: Russell Westbrook
Edad: 33
Estadísticas: 78 - 3.8 - 3.0 - 18.5
Valor: 91.3
Estado: no transferible

Tiene menor valor: Carmelo, Anthony

Input:

Rajon
Rondo
35
18
1.9
1.4
3.1

Output:

Jugador: Rajon Rondo
Edad: 35
Estadísticas: 18 - 1.9 - 1.4 - 3.1
Valor: 34.6
Estado: transferible

DeAndre	Jugador: DeAndre Jordan
Jordan	Edad: 33
33	Estadísticas: 32 - 0.7 - 1.6 - 4.1
32	Valor: 38.8
0.7	Estado: transferible
1.6	
4.1	Tiene menor valor: Rondo, Rajon

Estos casos de prueba le servirán como una ayuda para verificar que su programa funcione bien, pero los evaluados serán distintos a los aquí expresados.

2.2 Registro y operaciones:

Esta práctica proporciona dos registros, uno de tipo NombreCompleto con dos atributos: nombre y apellido; y otro registro de tipo Jugador que cuenta con 6 atributos: nombre completo, edad, partidos jugados (PJ), promedio de faltas (PF), promedio de pérdidas (PPE) y promedio de puntos (PP).

Para realizar la práctica se requiere hacer dos operaciones muy importantes: calcular el valor del jugador y calcular la transferibilidad.

- Calcular el valor del jugador: se necesita calcular previamente los puntos totales (PT), faltas totales (FT) y pérdidas totales (PET). Para ello, simplemente se debe multiplicar cada uno de los promedios por PJ (por ejemplo, para obtener PT se multiplica PP x PJ). Luego para obtener el valor del jugador se aplica la fórmula: $\frac{PT - (FT + PET)}{10}$. Si el resultado es menor a 10 se le suma la edad del jugador, pero si el resultado es mayor a 100 entonces el valor queda definido en 99.

- Calcular la transferibilidad del jugador: para este caso se requiere usar el número de victorias (V) y las derrotas (D), del equipo respectivamente, estas son proporcionadas en el enunciado de la práctica y luego se aplica la fórmula: $\frac{(PJ * 100)}{(V + D)}$. Si el resultado es menor a 65 y el jugador tiene una edad mayor a 30 años, el jugador es transferible, pero si el resultado es menor a 45 y el jugador tiene entre 26 y 30 años, también es transferible. Para este caso se considera a todos los jugadores menores de 25 años como no transferibles porque son parte del futuro del equipo.

3 Enunciado

LOS ANGELES LAKERS son un equipo de baloncesto profesional que participa en la National Basketball Association (NBA) y cuenta con grandes estrellas como Lebron James y Russell Westbrook. Sin embargo en la temporada regular 2021-2022, el equipo no tuvo

su mejor participación porque no clasificó a los playoffs por su récord de 33 victorias y 49 derrotas. Por este motivo, la directiva considera necesario hacer algunos cambios en el equipo este verano para afrontar la próxima temporada y mejorar los resultados. Los principales cambios de los LAKERS serán enfocados en traer mejores jugadores por lo que se requiere transferir algunos de ellos para liberar masa salarial. Para tomar las decisiones, se calculará tanto el valor como la transferibilidad del jugador.

4 Rutinas

4.1 Calentamiento:

a) **void obtener_datos(Jugador*)**: recibe un registro Jugador de tipo apuntador y deberá permitir obtener todos los atributos del Jugador.

b) **float calcular_valor(Jugador)**: recibe un registro Jugador y deberá retornar el cálculo del valor del jugador.

4.2 Programa:

a) **int calcular_transferibilidad(Jugador, int, int)**: recibe un registro Jugador y dos enteros, el primero para las victorias del equipo y el segundo para las derrotas del equipo. Deberá retornar 1 si el Jugador es transferible o retornar 0 en caso contrario.

b) **NombreCompleto comparar_jugadores(Jugador, Jugador)**: recibe dos variables de tipo Jugador que representa a dos jugadores. Deberá devolver el nombre completo del jugador que menor valor tiene para el equipo.

c) **void mostrar_datos(Jugador, Jugador)**: recibe dos variables de tipo Jugador que representa a dos jugadores y deberá mostrar todos los datos ingresados y calculados.

5 Evaluación

La fecha de entrega de esta práctica es desde el 10/10/2022 hasta el 17/10/2022. Para la evaluación se debe enviar el archivo *jugador.c* a la dirección:

castroj@ula.ve

El “asunto del correo” debe ser exactamente el texto “PR1-LAB-02” sin las comillas. Si falla con el asunto del correo entonces su práctica no será evaluada, ya que el mecanismo automatizado de filtrado no podrá detectar su trabajo. NO comprima el archivo y no envíe

nada adicional a éste. El único contenido que debe aparecer en el correo es el archivo “jugador.c” en archivo adjunto, y en el mensaje, su primer apellido, y su primer nombre, como se muestra en el siguiente ejemplo:

Castro, José

Durante el periodo de entrega usted tiene dos intentos para la corrección de su práctica, de modo que, si en el primer intento no obtuvo la calificación máxima, se le responderá al correo con los fallos para que los acomode y la vuelva a enviar, y el segundo intento será el que tendrá la calificación final de la práctica. Tenga presente el rango de entrega para sus dos intentos, considerando que en el primer intento el envío de los fallos no será inmediato.

Por favor, no use otros medios para enviar la solución ni escriba otros comentarios adicionales en el correo. **Atención:** si su programa no compila, entonces el evaluador no compila. Si una de las rutinas cae en un bucle infinito entonces el evaluador cae en un bucle infinito. Por esa razón, en todos estos casos será imposible darle una nota, lo que simplemente se traduce en que usted tiene cero en el intento en el cual ocurra una de las circunstancias mencionadas. No envíe programas que no compilan.

6 Recomendaciones

- Asegúrese de leer y comprender bien esta práctica antes de proceder al diseño e implementación de cada rutina.
- Use el grupo de Telegram para plantear las dudas de comprensión. Pero de ninguna manera comparta código, pues es considerado plagio.
- No modifique la plantilla que se envía, ni tampoco agregue parámetros adicionales. Al añadir nuevas variables debe asignarles nombres que ayuden a comprender el funcionamiento del código, además de aplicar bien el estilo de código PR1-ULA en general.

7 Referencias

[1] Datos del equipo
https://docs.google.com/spreadsheets/d/15DxHPPQTdKn42QtAP84sjWzxbF3UQnfzRdmqNQIa_jQ/edit?usp=sharing

[2] Estilo de código
<https://pr1-ula.readthedocs.io/en/latest/#welcome-to-mkdocs>