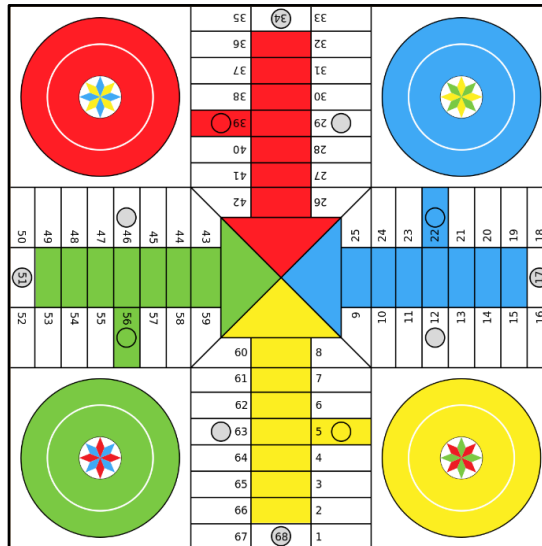


# Proyecto *Parchís*

## Versión 1

Fecha de entrega: **22 de noviembre**



El objetivo del proyecto es permitir a un usuario humano jugar al parchís contra otros jugadores controlados por la computadora. Para ello se desarrollará un programa en C++ en tres versiones sucesivas, en las que se irá añadiendo funcionalidad utilizando lo que ya se habrá visto en las clases. En esta primera versión simplemente queremos practicar un poco con la dinámica del juego con dos jugadores automáticos.

### 1. El juego: Parchís

Elementos y reglas del juego del parchís:

- Hay cuatro colores: rojo, verde, amarillo y azul.
- Cada jugador juega con cuatro fichas de su color asignado.
- Se utiliza un dado para determinar los movimientos para las fichas.
- *Casas*: zonas cuadradas con círculos en color que hay en las esquinas del tablero; las fichas de cada jugador comienzan encerradas en su casa.
- Alrededor de las casas hay 68 casillas, numeradas del 1 al 68. No puede haber más de dos fichas en una misma casilla.
- *Salidas*: casillas rectangulares en color que hay junto a cada casa (5, 22, 39 y 56).
- Los jugadores sacan una ficha a su salida cuando el dado marca un cinco.
- *Seguros*: casillas rectangulares con un círculo gris (12, 17, 29, 34, 46, 51, 63 y 68), además de las salidas.
- *Metas*: casillas triangulares en color que hay en el centro del tablero.
- *Zanatas*: casillas junto a las subidas a meta (17, 34, 51 y 68); cada ficha accede a la subida a meta desde la zanata de su jugador.

- *Puente*: dos fichas de igual color en una misma casilla (si están en un seguro es una barrera que no se puede franquear).
- Si un jugador, con un 5, saca una ficha de casa y en la salida ya hay dos fichas, alguna de ellas de otro jugador, se manda la última que haya llegado a casa.
- Las fichas se mueven siempre en el sentido creciente del número de casilla, siendo un circuito continuo (se pasa de nuevo a la casilla 1 desde la casilla 68).
- Una ficha que ya haya salido de casa y no esté bloqueada se podrá mover tantas casillas como indique el dado.
- Un jugador gana cuando todas sus fichas han llegado a la meta; las fichas sólo pueden llegar a la meta con el número de movimientos exacto.
- *Comer*: una ficha llega a una casilla, que no sea un seguro, en la que hay otra ficha de un contrario, enviando esta última a su casa y ganando una jugada extra de 20 movimientos.
- Cuando una ficha llega a la meta, el jugador gana una jugada extra de 10 movimientos.
- Cuando todas las fichas de un jugador están fuera de casa los seises se cuentan como siete.
- Cuando un jugador juega un seis, tiene derecho a una nueva jugada. Pero al tercer seis consecutivo se manda a casa a la última ficha movida por el jugador, excepto si ese último movimiento fue en la subida a meta.

Más información en: <https://es.wikipedia.org/wiki/Parchís>

UN PEQUEÑO CAMBIO: Para facilitar un poco los cálculos de los movimientos, en nuestro parchís **la casilla 68 pasa a ser la casilla 0**, de forma que las casillas estarán numeradas desde la 0 hasta la 67.

## 2. La primera versión

En esta primera versión jugaremos con una versión muy reducida del parchís, en el que sólo habrá dos jugadores:

- Cada uno de los dos jugadores dispone de una sola ficha.
- Cada jugador tiene un color asignado que el usuario elegirá al comenzar la ejecución.
- Las 68 casillas están dispuestas en una fila, numeradas de 0 a 67, y sólo cabe una ficha en cada casilla.
- El recorrido es circular, de forma que de la casilla 67 se pasa a la 0.
- Las casillas 0, 5, 12, 17, 22, 29, 34, 39, 46, 51, 56 y 63 son seguras (no se puede comer una ficha que esté en una casilla segura).
- La salida de la ficha amarilla es la 5, la de la azul es la 22, la de la roja es la 39 y la de la verde es la 56.
- La zanata (entrada a meta) de cada ficha es la casilla que está 5 antes de la salida. No hay subida a meta; basta con llegar a la zanata para ganar, aunque hay que llegar con una tirada exacta (si quedan 2 casillas y se saca un 3, no se mueve).
- El objetivo es que la ficha llegue la primera a la correspondiente zanata.
- Si una ficha sale de casa (con un 5) y hay otra ficha en su salida, ésta se va a casa.

- Con un 6 el jugador repite jugada, pero al tercer seis consecutivo su ficha se va a casa.
- Si un jugador come la ficha del otro, la de éste se va a casa y el primero cuenta 20 más.
- La ficha de cada jugador se representará con la inicial del color en mayúscula, excepto la amarilla, para la que se mostrará una M.

Las tiradas del dado se generarán aleatoriamente.

### 3. Representación del tablero

Podemos representar el tablero de forma lineal, por ejemplo así:

```

00000000001111111112222222223333333333444444444555555555666666666
01234567890123456789012345678901234567890123456789012345678901234567
CALLE 0---o-----o---o---o---o---o---o---o---o---o---o---o---o---o---o---
CASA  0---o-----o---o---o---o---o---o---o---o---o---o---o---o---o---o---
          M                                     R

```

Las casillas son las posiciones que hay entre las dos líneas de guiones y círculos (los círculos indican las casillas que son seguras). En el ejemplo la ficha roja está en la casilla 47 y la amarilla está en casa. Inicialmente las dos están en casa (por debajo de la línea inferior de guiones y círculos, a la altura de su casilla de salida):

```

00000000001111111112222222223333333333444444444555555555666666666
01234567890123456789012345678901234567890123456789012345678901234567
CALLE 0---o-----o---o---o---o---o---o---o---o---o---o---o---o---o---o---
CASA  0---o-----o---o---o---o---o---o---o---o---o---o---o---o---o---o---
          M                                     R

```

En cada tirada se indicará la ficha que se mueve (su color) y si la ficha sale de casa, llega a su zanata (gana), come otra ficha, llega a otra casilla... o simplemente no puede mover.

### 4. Detalles de implementación

Declara un tipo enumerado `tColor` cuyos valores sean los cuatro colores del parchís (Rojo, Verde, Amarillo y Azul).

Implementa las siguientes funciones de manejo del tipo `tColor`:

- ❖ `string colorACadena(tColor color)`: devuelve el nombre completo del color.
- ❖ `char letra(tColor color)`: devuelve la letra que representa el color (R=Rojo, V=Verde, M=Amarillo y A=Azul).

El usuario podrá elegir al comenzar qué dos colores van a competir. Necesitarás 2 variables con los números de los jugadores que juegan (1=Rojo, 2=Verde, 3=Amarillo y 4=Azul). También necesitarás otras dos variables para saber la casilla que ocupa en cada momento la ficha de cada jugador (-1 si la ficha está en casa).

Implementa otra función que dada una casilla diga si es segura o no, así como otras funciones que puedas considerar útiles.