

Ejercicio POO

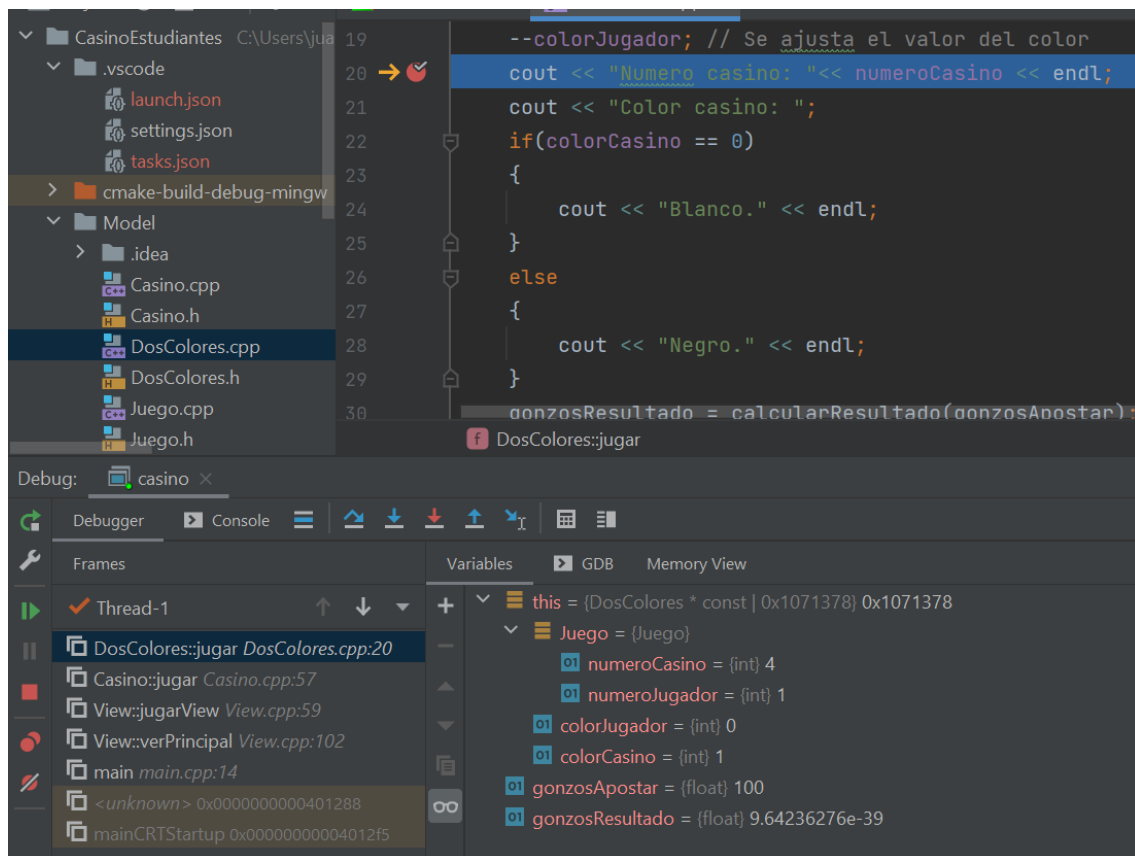
Practica de uso del debugger

Use el material disponible en esta página y las siguientes relacionadas con debug para completar el proceso aquí solicitado <https://www.jetbrains.com/help/clion/debugging-code.htm>

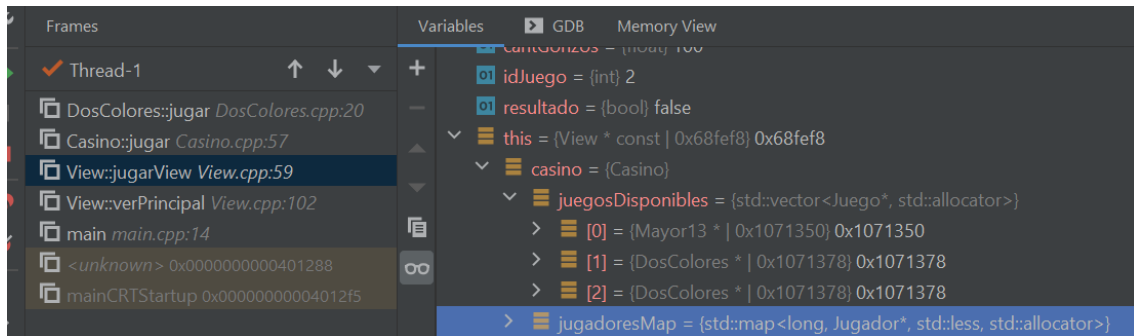
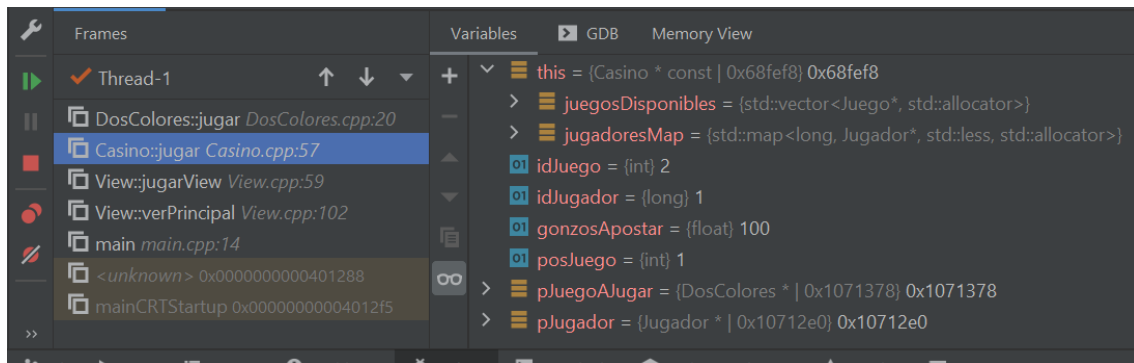
Como resultado de esta práctica deberá subir un documento a su repositorio (en formato PDF) con las fotos de evidencias para cada pregunta. Luego debe subir el link de este archivo en el repositorio a TEAMS.

1. Agregue un breakpoint al programa de casino en el método jugar de la clase DosColores. Prueba la ejecución hasta que llegue al punto del breakpoint y su programa se interrumpa. En ese punto tome una foto de los frames disponibles e indique:

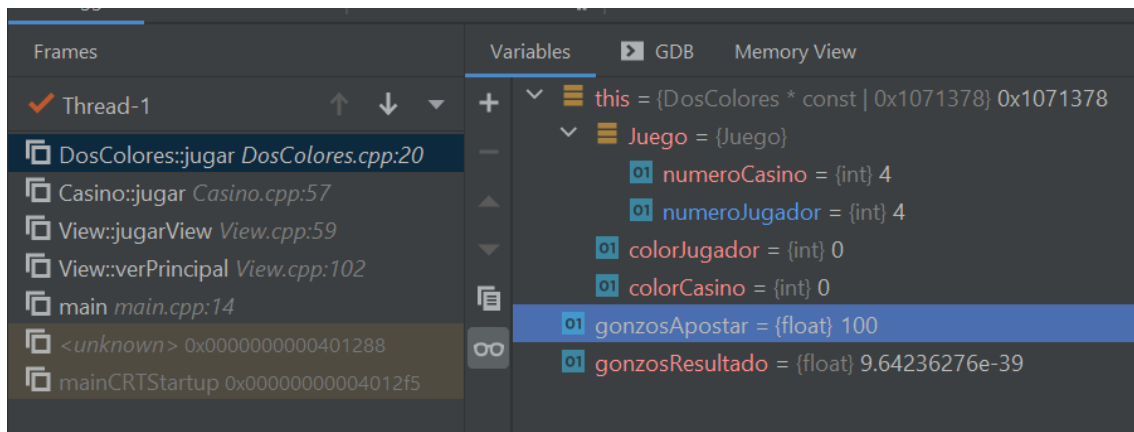
a. Cuál fue el método que inició toda la ejecución : mainCRTStartup



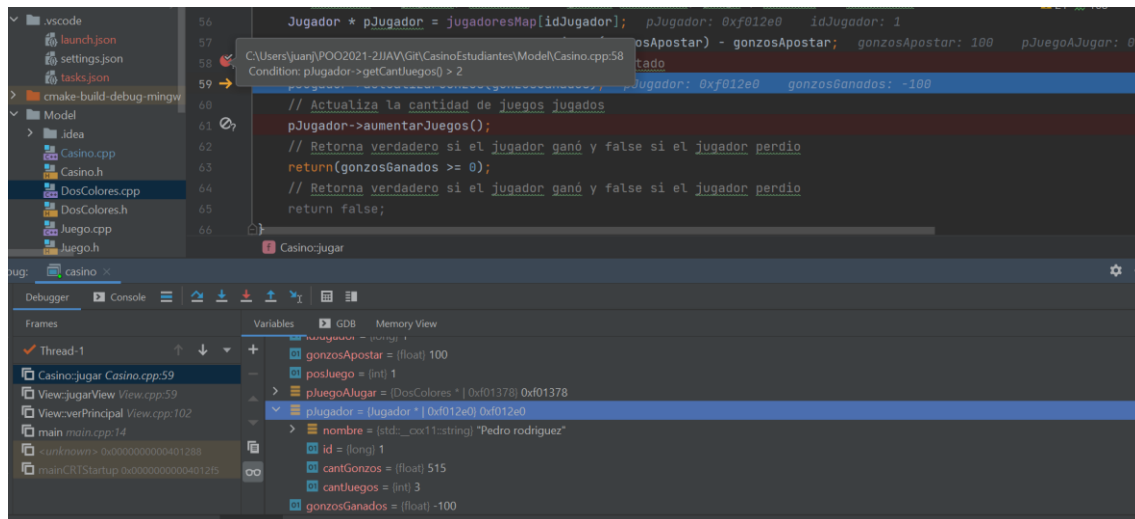
b. Explore que elementos tiene la variable `this` en el frame `DosColores`, en el frame `Casino` y en el frame `jugarView`. Tome una foto que evidencie los valores de las variables de algunos de estos frames.



2. Set variable values: en el mismo breakpoint, una vez se han generado las variables aleatorias para el casino y para el jugador ajústelas directamente en el debug para asegurarse de que el jugador gane.

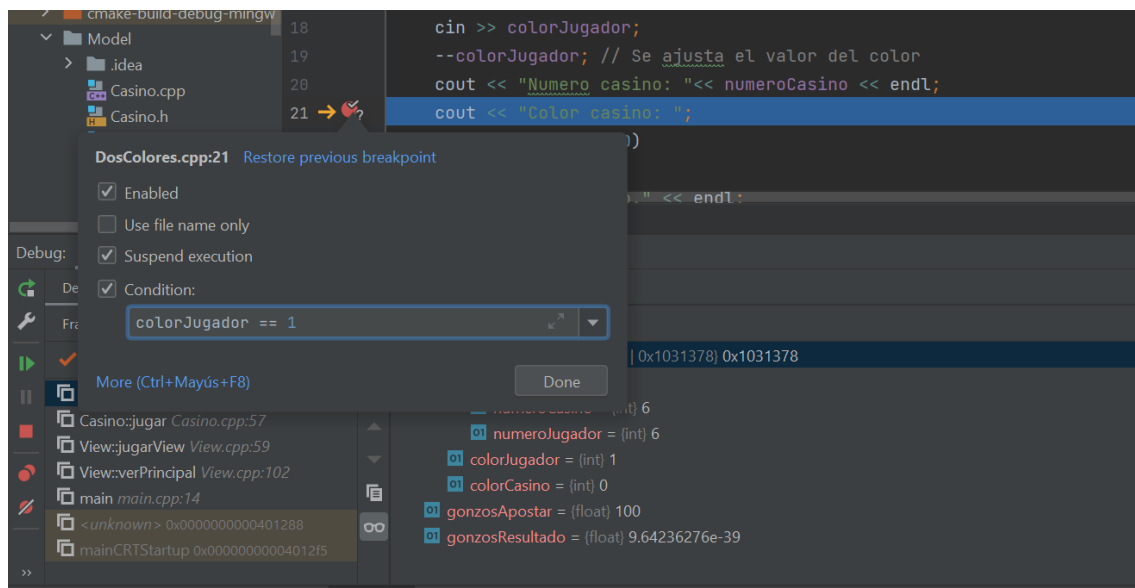


3. Cree un breakpoint que se lance después de que ocurran cierto número de repeticiones.



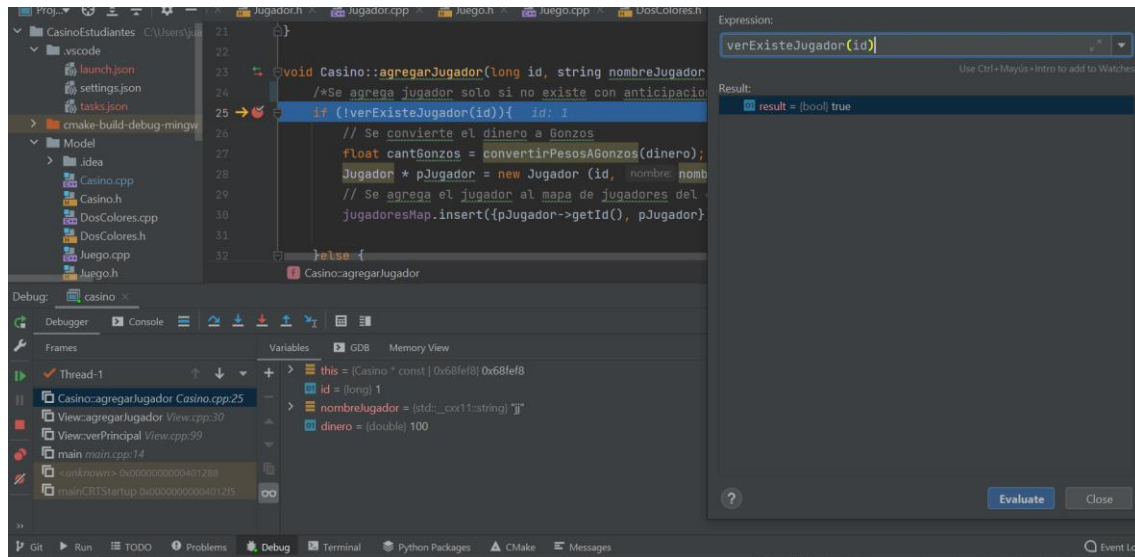
Para cuando el jugador haya jugado mas de 2 veces

4. BreakPoint condicional: incorpore un breakpoint condicional en el código fuente. Explique en qué funcionalidad lo incorporó. Tome una foto que muestre que el programa se interrumpió exitosamente y muestre el estado de las variables cuando se interrumpió la ejecución.

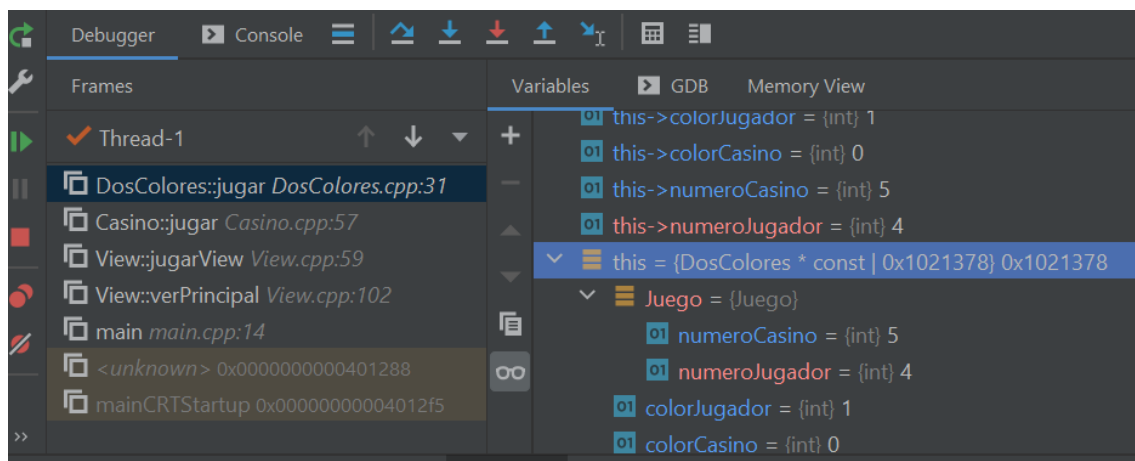


Si el jugador escogía negro, paraba el programa

5. Pruebe el evaluador de expresiones con alguna expresión. Tome una foto que muestre que lo probó



6. Agregue un watch para observar alguna de las variables definidas a lo largo del casino.



De las variables en watch, cambiaron todas menos el número del jugador.

7. Explique: qué diferencia encontró entre la funcionalidad step over (F8) y la funcionalidad step into (F7). Explique con un ejemplo de las funcionalidades del casino.

El step over se salta el break point y se queda como en la función y en el cpp, en cambio con el step into indaga dentro de las funciones internas y va a otros cpp.