

Paradigmas de Programación

Práctica I – Curso 2023/24

Blackjack

Descargo de Responsabilidad: El presente documento es de uso interno para alumnos de la asignatura Paradigmas de Programación del Grado en Ingeniería Informática y del Grado en Estadística de la Universidad de Valladolid. En su contenido se utilizan elementos que pueden estar sujetos a derechos de propiedad intelectual.

Por tanto, se prohíbe la copia, tratamiento y difusión por cualquier medio de este documento fuera del ámbito anteriormente expuesto.

1. Objetivo y Descripción General

El objetivo de la práctica es el desarrollo de una aplicación en lenguaje Python que permita **jugar y evaluar estrategias** para una **versión especial** del juego **Blackjack** en un entorno de texto (no gráfico).

El Blackjack (o *veintiuna*) es un juego de origen español (su primera mención aparece en la Novela Ejemplar “Rinconete y Cortadillo” de Cervantes) en el que un único jugador¹ compite contra otro (el **croupier** o *la banca*) con el objetivo de conseguir manos (conjunto de cartas) con valor más cercano a 21 sin superarlo. En lo que sigue se describen las reglas de la **versión especial** de este juego que se va a implementar en esta práctica. Esta versión **no tiene por qué corresponderse con ninguna otra** de las que se utilizan en el mundo real:

1. El juego de cartas que se utiliza es el de la baraja francesa: Cuatro palos (tréboles ♣, picas ♠, diamantes ♦ y corazones ♥) con 13 cartas cada uno (10 cartas numeradas [1..10] y 3 figuras [J, Q, K]). En total cada baraja tiene 52 cartas. La única característica relevante de las cartas en este juego es su **valor**, que es **10** para las figuras y su número para el resto de las cartas, **salvo los ases** (cartas con número 1), que **pueden valer 1 o 11** según le sea más conveniente al jugador. Los palos no juegan ningún papel en este juego².
2. Un **juego** consiste en la realización de una serie de partidas de un único jugador contra el croupier, en nuestro caso se corresponde con una ejecución completa de nuestro programa. Al principio del juego se crea un **mazo** juntando **2 barajas** y barajando (desordenando) sus cartas. Durante el desarrollo de las partidas se van extrayendo las cartas del mazo hasta que se acabe, momento en el que se genera un nuevo mazo. El llevar la cuenta de las cartas que han salido para estimar las probabilidades de éxito de las acciones del jugador es la base de muchas de las estrategias que intentan maximizar las ganancias en este juego³.
3. Denominaremos **mano** a cada conjunto de cartas con el que está jugando en un momento dado uno de los participantes (jugador o croupier). Veremos posteriormente que existe la posibilidad de que en una partida el jugador este jugando con **varias manos** a la vez (el croupier siempre juega con una única mano). En un momento dado cada mano puede estar en uno de los siguientes **estados**:
 - **Pasada**: Cuando el valor de la mano es mayor que 21.
 - **Abierta**: Disponible para pedir cartas adicionales.
 - **Cerrada**: Cuando el jugador o croupier se ha plantado.
4. El **valor de una mano** es la suma de los valores de sus cartas. Si hay un as en la mano, su valor es 11 salvo que con ello la mano supere el valor 21, en cuyo caso su valor se toma como 1. Si existen varios ases solo uno de ellos como mucho podría valer 11 (dos ases contando como 11 valdrían 22).
5. Cada mano del jugador (y recordad que el jugador puede tener varias) tiene asociada una cantidad de dinero denominada **apuesta** de la mano.
6. Al principio de **cada partida** se elige el valor de la apuesta para esa partida. Como el jugador comienza con una única mano a esa mano se la asigna inicialmente ese valor de apuesta (veremos que en el transcurso de la partida la apuesta puede duplicarse). Las cantidades de apuesta permitidas son 2€ (apuesta baja), 10€ (apuesta media) o 50€ (apuesta alta).

¹ Aunque en una partida puede participar más de un jugador, los jugadores no compiten entre sí, sino que cada jugador juega individualmente contra el Croupier y el hecho de que existan otros jugadores no afecta para nada a sus resultados, por lo que a todos los efectos se pueden considerar como una serie de partidas individuales (jugador contra croupier) que se desarrollan en paralelo.

² Existe una versión en la cual el obtener una jugada en el primer reparto con valor 21 en la que intervenga una jota de palo negro (black Jack) se paga x10 (el nombre del juego proviene de esa versión), pero esa regla no la vamos a seguir en esta práctica.

³ En algunos casinos está prohibido el usar este tipo de métodos en este juego. Los profesores de la asignatura no se responsabilizan de posibles ilegalidades cometidas por los alumnos de la asignatura usando los conocimientos adquiridos en la realización de esta práctica.

7. A diferencia de otras versiones, todas las cartas se reparten boca arriba, es decir se conoce inmediatamente su valor.
8. **Reparto Inicial:** Al comienzo de cada partida se reparte una carta al croupier y dos cartas al jugador. Si con sus cartas el jugador suma 21 se produce un **blackjack** y se termina **inmediatamente** la partida, obteniendo el jugador un beneficio de $\frac{3}{2}$ el valor de la apuesta base.

Atención: Solo se considera blackjack si se produce en el reparto inicial, el conseguir 21 en el desarrollo posterior de la partida no tiene ningún significado especial ni tiene una contabilidad diferente en el reparto de beneficios.
9. Tras el reparto inicial, si no se ha producido blackjack, se lleva a cabo el desarrollo de la partida que consta de tres etapas: El turno del jugador, el turno del croupier, y la contabilización de resultados.
10. **Turno del jugador:** Mientras el jugador tenga alguna mano abierta, se pregunta al jugador para cada una de esas manos la opción deseada:
 - **Pedir** carta adicional: Se añade la siguiente carta del mazo a esa mano. Si la suma es mayor que 21 la mano cambia al estado **pasada**, si no la mano sigue estando abierta (incluyendo el caso en que la suma es exactamente 21).
 - **Doblar** apuesta: En este caso el jugador duplica el valor de la apuesta de esa mano, se añade la siguiente carta del mazo a esa mano y si la suma es mayor que 21 la mano cambia al estado **pasada**, en caso contrario pasa al estado **cerrada** (es decir, doblar implica pedir una única carta).
 - **Cerrar** mano: No se pide carta, la mano cambia al estado **cerrada**.
 - **Separar** mano: Esta opción solo aparece si la mano tiene exactamente dos cartas y ambas tienen el mismo valor facial. En este caso se crea una nueva mano conteniendo únicamente la segunda carta (la cual desaparece de la mano original). Ambas manos (la antigua y la nueva) tienen el estado **abierta**.
11. **Turno del croupier:** Si el jugador se ha pasado en **todas** sus manos, el croupier no pide ninguna carta y se pasa a la etapa de contabilización de resultados. En caso contrario el croupier pide cartas hasta que **iguala o supera el valor 17**. El juego del croupier está prefijado y no depende de las jugadas del contrario: Por ejemplo si el jugador tiene una mano de valor 18 y el crupier llega al valor 17, el crupier no puede pedir una carta adicional.
12. **Contabilización del resultado:** Cada mano del jugador se compara de forma independiente con la mano del crupier.
 - Si ambas manos están **pasadas** o tienen el mismo valor, ninguno obtiene beneficio.
 - Si el crupier se ha pasado o la mano del jugador es mayor, el croupier paga al jugador la apuesta de esa mano.
 - Si el jugador se ha pasado o su mano es menor que la del croupier, el jugador paga al croupier la apuesta de esa mano.
 El beneficio o pérdida del jugador en esa partida será la suma de los resultados de todas sus manos.
13. Tras esta última etapa, se pregunta al jugador si desea otra partida y en caso afirmativo se pasa a la etapa 7.

2. Descripción Detallada

El programa va a tener dos modos de ejecución: **Juego** y **Análisis**. Al inicio de la ejecución se le pregunta al usuario en qué modo se va a ejecutar. La reglas del juego y la forma de trabajar del programa es básicamente la misma en ambos modos, las únicas diferencias son las siguientes:

- **Modo Juego:** Al principio de la partida se pide al usuario el valor de apuesta (de entre los tres disponibles). En el turno del jugador se pide al usuario que introduzca la opción deseada por teclado. Al finalizar cada partida se pregunta al usuario si desea jugar otra.
- **Modo Análisis:** Se pide al usuario que indique el número de partidas que se van a jugar y a partir de ese momento no se vuelve a pedir ninguna entrada por teclado. El valor de la apuesta y las opciones que juegan en el turno del jugador se obtienen mediante un algoritmo (se explicará más adelante). La salida por pantalla será la misma que en el modo Juego.

Se muestra a continuación un ejemplo de ejecución del programa en modo juego con dos partidas. Las entradas del usuario se muestran resaltadas en color rojo:

```
*** BLACKJACK - PARADIGMAS DE PROGRAMACIÓN 2023/24 ***
¿Modo de ejecución? [J]uego [A]nálisis: J

--- INICIO DE LA PARTIDA #1 --- BALANCE = 0 €
¿Apuesta? [2] [10] [50] 10

REPARTO INICIAL
<A>Croupier (11): [A]
<A>Mano (21): [10][A]
*****
*** BLACKJACK ***
*****

Ha ganado 15 €!
¿Otra partida? [S/N] S

--- INICIO DE LA PARTIDA #2 --- BALANCE = 15 €
¿Apuesta? [2] [10] [50] 10

REPARTO INICIAL
<A>Croupier (8): [8]
<A>Mano (20): [J][10]

TURNO DEL JUGADOR
¿Jugada para Mano? [P]edir [D]oblar [C]errar [S]eparar S

<A>ManoA (10): [J] | <A>ManoB (10): [10]
¿Jugada para ManoA? [P]edir [D]oblar [C]errar P
¿Jugada para ManoB? [P]edir [D]oblar [C]errar P

<A>ManoA (12): [J][2] | <A>ManoB (20): [10][J]
¿Jugada para ManoA? [P]edir [D]oblar [C]errar P
¿Jugada para ManoB? [P]edir [D]oblar [C]errar [S]eparar S

<A>ManoA (20): [J][2][8] | <A>ManoBA (10): [10] | <A>ManoBB (10): [J]
¿Jugada para ManoA? [P]edir [D]oblar [C]errar C
¿Jugada para ManoBA? [P]edir [D]oblar [C]errar P
¿Jugada para ManoBB? [P]edir [D]oblar [C]errar P

<C>ManoA (20): [J][2][8] | <A>ManoBA (20): [10][K] | <A>ManoBB (20): [J][10]
¿Jugada para ManoBA? [P]edir [D]oblar [C]errar [S]eparar S
¿Jugada para ManoBB? [P]edir [D]oblar [C]errar [S]eparar C

<C>ManoA (20): [J][2][8] | <A>ManoBAA (10): [10] | <C>ManoBB (20): [J][10] | <A>ManoBAB (10): [K]
¿Jugada para ManoBAA? [P]edir [D]oblar [C]errar P
¿Jugada para ManoBAB? [P]edir [D]oblar [C]errar P

<C>ManoA (20): [J][2][8] | <A>ManoBAA (12): [10][2] | <C>ManoBB (20): [J][10] | <A>ManoBAB (19): [K][9]
¿Jugada para ManoBAA? [P]edir [D]oblar [C]errar D
¿Jugada para ManoBAB? [P]edir [D]oblar [C]errar P

<C>ManoA (20): [J][2][8] | <C>ManoBAA (20): [10][2][8] | <C>ManoBB (20): [J][10] | <P>ManoBAB (23): [K][9][4]

TURNO DEL CROUPIER
<A>Croupier (8): [8]
<A>Croupier (18): [8][Q]

FIN DE PARTIDA
<C>Croupier (18): [8][Q]
<C>ManoA (20): [J][2][8] | <C>ManoBAA (20): [10][2][8] | <C>ManoBB (20): [J][10] | <P>ManoBAB (23): [K][9][4]

CONTABILIZACIÓN DE RESULTADOS
* Croupier: 18, ManoA: 20 -> +10
* Croupier: 18, ManoBAA: 20 -> +20
* Croupier: 18, ManoBB: 20 -> +10
* Croupier: 18, ManoBAB: 23 -> -10
Resultado de la partida: +30
¿Otra partida? [S/N] N

BALANCE FINAL: 45 €
```

2.1. Entradas del usuario

Se puede apreciar en el ejemplo que en todas las interacciones con el usuario se le pide que escoja entre una serie de opciones. El comportamiento va a ser siempre el mismo: Si el usuario introduce una cadena vacía (pula directamente Enter) entonces se entiende que ha elegido la opción por defecto (modo juego, apuesta 10, acción pedir y si jugar otra partida). Si introduce un texto que no es una de las opciones disponibles se vuelve a realizar la petición (no es necesario mostrar un mensaje de error). Por último, no se distingue entre mayúsculas y minúsculas en las opciones.

2.2. Nombre de las manos

Dado que un jugador puede tener varias manos, es necesario identificarlas para que el usuario pueda saber a cuál de ellas nos referimos al solicitar una acción. Por eso **toda mano va a tener asociado un nombre** (un string). El nombre de la única mano del croupier será "Croupier", y el de la mano inicial del jugador será "Mano". Cuando una mano del jugador se divida en dos al elegir el usuario la opción **Separar**, se añade el carácter 'A' a la mano original, y la nueva mano tendrá el nombre de la original más el carácter 'B'.

2.3. Representación de las manos

A la hora de mostrar las jugadas del croupier y del jugador existen dos posibilidades, que influyen en la calificación:

2.3.1. Representación cutre

Las mano se separan entre sí por el carácter '|', y para cada una de ellas se indica entre corchetes angulares el estado (A -> Abierto, C -> Cerrado, P -> Pasado), su nombre, el valor de la mano entre paréntesis y las cartas que la componen. Las cartas se indican encerrando entre corchetes cuadrados su número/letra. Si se elige esta representación la práctica se evalúa sobre 8 puntos.

```
<P>ManoA (22): [J][4][8] | <C>ManoBA (19): [10][9] | <A>ManoBB (10): [J]
```

2.3.2. Representación deluxe

La representación 'deluxe' es más compleja porque implica componer varias líneas de texto, alineando adecuadamente los distintos elementos. Se utilizan caracteres Unicode para representar los bordes y el palo de las cartas y además se añade la información de la apuesta asociada a cada mano (solo en el caso del jugador, esa información no se presenta en la mano del croupier):

ManoA: (22) 10€ PASADA	<div>J</div> <div>4</div> <div>8</div>	ManoBA: (19) 20€ Cerrada	<div>10</div> <div>9</div>	ManoBB: (10) 10€ Activa	<div>J</div>
---------------------------------	--	-----------------------------------	----------------------------	----------------------------------	--------------

Si se elige esta representación la práctica se evalúa sobre 10 puntos. Se recomienda utilizar los métodos asociados a la clase string (join, split, rjust). Si se usan técnicas de programación funcional (list comprehensions, map, zip, unpack (*)) el código se puede abreviar de forma considerable.

2.4. Estructura del programa

Tras elegir el modo de ejecución, el programa consistirá en un bucle de partidas, que terminará cuando el usuario indique que no desea jugar otra partida (modo jugador) o cuando se haya jugado el número de partidas indicado (modo análisis). Cada partida consta de las siguientes fases:

2.4.1. Petición de la apuesta

Se indica el número de partida, el balance (dinero ganado o perdido) actual y se pide al usuario que indique la apuesta (solo hay 3 valores permitidos: 2, 10 y 50 euros). En el modo análisis el valor de la apuesta se obtiene algorítmicamente (sección 2.5 y 2.6)

2.4.2. Reparto inicial

Se reparte una carta al croupier y dos cartas al jugador (ver sección 2.5) y se muestra la jugada. Si el jugador tiene blackjack (su mano vale 21) se indica tal como aparece en el ejemplo y se termina la partida, omitiéndose el resto de las fases.

2.4.3. Turno del jugador

Consta de un bucle doble: Mientras el jugador tenga manos activas, se pide al usuario la opción deseada para cada una de sus manos que esté todavía activa (en el caso de modo análisis la opción se obtiene algorítmicamente, ver secciones 2.5 y 2.6) y tras esas peticiones se muestra el estado de las manos del jugador.

En el ejemplo podéis ver que al preguntar se indica el nombre de la mano, para que el usuario pueda identificarla, y que la opción de **Separar** solo está disponible si la composición de la mano lo permite (exactamente 2 cartas y del mismo valor).

El turno del jugador termina cuando no queda ninguna mano activa. Si todas sus manos están **pasadas** entonces se omite la siguiente fase (turno del croupier)

2.4.4. Turno del croupier

Consta de un bucle donde mientras el valor de la mano del croupier sea **menor que 17** se le reparte una carta y se muestra su mano en pantalla.

2.4.5. Final de partida y contabilización de resultados

Se muestra el estado final de las jugadas y se desglosa, con una línea por cada mano del jugador, el resultado de comparar la mano del croupier con la del jugador, indicando el beneficio o pérdida de esa jugada. Se escribe al balance total de la partida.

2.5. Elementos externos: CartaBase, Mazo y Estrategia

Para simplificar el desarrollo y evaluación de la práctica se proporcionará un fichero, **externo.py**, que contiene las definiciones de tres clases para que las utilizéis en vuestros programas:

2.5.1. Clase CartaBase

El objetivo de esta clase es representar una carta de la baraja de la forma más sencilla y minimalista posible, ya que las clases Mazo y Estrategia necesitan conocer el tipo de datos que representa una carta.

Esta clase representa cada carta mediante un entero en el rango 0..51, que se le proporciona en el constructor. También define una propiedad, **valor**, que devuelve un entero en el rango 1..10 indicando el valor facial de la carta (los ases devuelven 1).

Lo ideal sería que vosotros definierais una nueva clase, **Carta**, que heredara de CartaBase y a la cual añadiríais los atributos y métodos que consideréis necesarios (por ejemplo un método para obtener la representación en texto de la carta), pero no es obligatorio.

2.5.2. Clase Mazo

La clase Mazo representa el mazo del cual se extraen las cartas del juego. En vuestro programa la **única forma de crear objetos que representen cartas** es utilizando un objeto que pertenezca a esta clase.

El objeto Mazo tiene un único método, **reparte**, que no tiene parámetros y devuelve objeto de tipo carta. El código de este método garantiza que su comportamiento sea similar al de un mazo real, volviendo a llenarse de cartas ordenadas al azar cuando se vacía.

Su constructor tiene dos parámetros: Una referencia a una clase (que debe ser CartaBase o una clase que herede de ella) y un objeto de clase Estrategia:

- El pasar al constructor la referencia a la clase se hace para que podáis crearos vuestra propia clase que represente una carta y que el objeto Mazo os pueda proporcionar objetos **de esa clase** al llamar al método **repartir**.
- El objetivo de proporcionar el objeto de clase Estrategia al mazo es que al repartir una carta el mazo le puede pasar esa información al objeto Estrategia.

2.5.3. Clase Estrategia

El objeto de clase Estrategia simula el comportamiento de un jugador contador de cartas 'experto' en este juego. Este objeto solo se utilizará en el **modo análisis** del juego. Este objeto realiza una estimación de la distribución de cartas restantes en el mazo, de forma que se puedan apostar cantidades mayores cuando la distribución sea favorable al jugador y menores cuando sea desfavorable. También indica la mejor acción (pedir, doblar, cerrar o separar) dadas las manos del croupier y el jugador.

Dispone de los siguientes métodos:

- El método **apuesta** indica si se debe hacer una apuesta de valor alto, medio o bajo. Recibe como parámetros los valores de esos tipos de apuesta (2, 10 y 50 en nuestro caso) y devuelve el valor sugerido.
- El método **jugada** recibe como parámetros la carta del croupier y la lista de cartas de la mano del jugador que se está jugando en ese momento, y devuelve una letra ('P', 'D', 'C' o 'S') indicando la mejor acción.
- Existe un método adicional, **cuenta_carta**, que sirve para actualizar su información sobre las cartas del mazo que ya han salido. Nuestro programa **no debe llamar** a este método, porque es el objeto Mazo el que se encarga de ello.
- Su constructor solo tiene un parámetro, un entero que indica el número de barajas de que consta el mazo utilizado.

2.5.4. Uso de las clases en vuestro programa

Vuestro programa debe crear al principio un objeto de la clase Estrategia y un objeto de la clase Mazo. El objeto mazo se usará cada vez que haya que obtener una carta (llamando a su método **repartir**) y el objeto estrategia solo se utilizará si el modo es análisis, para sustituir la entrada del usuario cuando hay que obtener el valor de apuesta (llamando a su método **apuesta**) y cuando hay que obtener la opción de jugada para cada mano (llamando a su método **jugada**). Este sería el esquema básico de vuestro programa respecto a la importación y creación de los objetos:

```
from externo import Mazo, Estrategia, CartaBase

class Carta(CartaBase):
    ... métodos adicionales ...

... otras clases y funciones ...

def main():
    estrategia = Estrategia(Mazo.NUM_BAJAS)
    mazo = Mazo(Carta, estrategia)
    ...
```

Atención: Tened en cuenta que el fichero **experto.py** que se encuentra en el Campus Virtual va a ser sustituido por otro en los distintos casos de prueba y en la defensa de la práctica. Todos los ficheros *experto.py* van a definir las 3 clases y cada clase va a contener los métodos y propiedades que se han descrito, pero el código y los atributos internos de cada clase pueden cambiar, por lo tanto debéis absteneros de (1) corta-pegar código de *experto.py* en vuestro programa para ahorraros el importar el fichero o parte de él y (2) que vuestro código dependa de características internas de la versión de *experto.py* que está en el Campus Virtual.

3. Criterios de Evaluación

Se deben cumplir las siguientes **condiciones** para que la práctica **sea evaluada** (en caso contrario se calificará con un cero):

- La práctica se debe realizar individualmente o en pareja (dos alumnos).
- Si se realiza en pareja, es posible que la calificación de ambos sea distinta, según el desarrollo de la defensa.
- El código presentado debe haber sido desarrollado **en su totalidad** por el/los alumno(s).
- El código debe poder evaluarse sin errores durante la defensa. En el caso de errores triviales los alumnos deben ser capaces de corregirlos in situ inmediatamente.
- Durante la defensa se solicitará a los alumnos que **modifiquen su código** para implementar un cambio sencillo en las condiciones de la práctica. Para que se evalúe la práctica los alumnos deben ser capaces de realizar la modificación y obtener un código funcional.
- No son necesarias ni se permite el uso de ninguna librería. Si necesita usar alguna primero debe contar con la aprobación del profesor.
- No se permite el uso de **variables globales** (ver <https://realpython.com/python-use-global-variable-in-function/>)
- Debe existir una función denominada **main** cuya ejecución inicie el programa.
- Se permite entregar una versión reducida de la práctica en la que **no exista la opción de separar una mano**, pero en ese caso la nota se dividirá por 2.

En la evaluación de la práctica se tendrán en cuenta, entre otros, los siguientes aspectos:

- La correcta resolución del problema y la implementación de los distintos aspectos mencionados en el enunciado.
- La modularidad y estructuración de la solución (división en funciones y clases). La utilización de orientación a objeto no es obligatoria, pero se valorará muy positivamente.
- La documentación interna. Deben existir al menos comentarios que indiquen el propósito y la forma en que se debe usar cada función y clase del programa, así como los comentarios adecuados para cada parte del código donde se vaya a realizar alguna tarea no trivial.
- El uso de las técnicas impartidas en la asignatura.

4. Presentación y Evaluación de la práctica

La evaluación de la práctica se divide en dos etapas:

1. Presentación electrónica del fichero o ficheros ***.py** que contienen el código de la práctica (sin incluir **experto.py**). Se habilitará en el Campus Virtual de la asignatura una tarea de subida de ficheros cuya fecha límite será el **domingo 7 de abril a las 23:59**. Al principio de todos los ficheros debe aparecer un comentario con el nombre de quienes la han desarrollado.
2. Evaluación **presencial**, en laboratorio, ante el profesor. Se realizará en el lugar, día y hora correspondiente al horario de prácticas del subgrupo al que pertenezca durante la semana del 8 al 12 de abril.

Anexo A – Ejemplo en modo 'deluxe'

*** BLACKJACK – PARADIGMAS DE PROGRAMACIÓN 2023/24 ***
 ¿Modo de ejecución? [J]uego [A]nálisis: J

--- INICIO DE LA PARTIDA #1 --- BALANCE = 0 €
 ¿Apuesta? [2] [10] [50] 10

REPARTO INICIAL

Croupier:
 (11) A
 Activa

Mano:
 (21) 10 A
 10€
 Activa

 *** BLACKJACK ***

Ha ganado 15 €!
 ¿Otra partida? [S/N] S

--- INICIO DE LA PARTIDA #2 --- BALANCE = 15 €
 ¿Apuesta? [2] [10] [50] 10

REPARTO INICIAL

Croupier:
 (8) 8
 Activa

Mano:
 (20) J 10
 10€
 Activa

TURNO DEL JUGADOR

¿Jugada para Mano? [P]edir [D]oblar [C]errar [S]eparar S

ManoA:
 (10) J
 10€
 Activa

ManoB:
 (10) 10
 10€
 Activa

¿Jugada para ManoA? [P]edir [D]oblar [C]errar P
 ¿Jugada para ManoB? [P]edir [D]oblar [C]errar P

ManoA:
 (12) J 2
 10€
 Activa

ManoB:
 (20) 10 J
 10€
 Activa

¿Jugada para ManoA? [P]edir [D]oblar [C]errar P
 ¿Jugada para ManoB? [P]edir [D]oblar [C]errar [S]eparar S

ManoA:
 (20) J 2 8
 10€
 Activa

ManoBA:
 (10) 10
 10€
 Activa

ManoBB:
 (10) J
 10€
 Activa

¿Jugada para ManoA? [P]edir [D]oblar [C]errar C
 ¿Jugada para ManoBA? [P]edir [D]oblar [C]errar P
 ¿Jugada para ManoBB? [P]edir [D]oblar [C]errar P

ManoA:
 (20) J 2 8
 10€
 Cerrada

ManoBA:
 (20) 10 K
 10€
 Activa

ManoBB:
 (20) J 10
 10€
 Activa

¿Jugada para ManoBA? [P]edir [D]oblar [C]errar [S]eparar S
 ¿Jugada para ManoBB? [P]edir [D]oblar [C]errar [S]eparar C










ManoA:
 (20) J 2 8
 10€
 Cerrada












ManoBAA:
 (10) 10
 10€
 Activa

ManoBB:
 (20) J 10
 10€
 Cerrada


ManoBAB:
 (10) K
 10€
 Activa



¿Jugada para ManoBAA? [P]edir [D]oblar [C]errar P
 ¿Jugada para ManoBAB? [P]edir [D]oblar [C]errar P

ManoA: (20) 10€	  	ManoBAA: (12) 10€	 	ManoBB: (20) 10€	 	ManoBAB: (19) 10€	 
Cerrada		Activa		Cerrada		Activa	
¿Jugada para ManoBAA? [P]edir [D]oblar [C]errar D							
¿Jugada para ManoBAB? [P]edir [D]oblar [C]errar P							



ManoA: (20) 10€	  	ManoBAA: (20) 20€	  	ManoBB: (20) 10€	 	ManoBAB: (23) 10€	  
Cerrada		Cerrada		Cerrada		PASADA	












TURNO DEL CROUPIER

Croupier:
(8)
Activa 

Croupier:
(18)
Activa  

FIN DE PARTIDA

Croupier:
(18)
Cerrada  

ManoA: (20) 10€	  	ManoBAA: (20) 20€	  	ManoBB: (20) 10€	 	ManoBAB: (23) 10€	  
Cerrada		Cerrada		Cerrada		PASADA	

CONTABILIZACIÓN DE RESULTADOS

* Croupier: 18, ManoA: 20 -> +10
 * Croupier: 18, ManoBAA: 20 -> +20
 * Croupier: 18, ManoBB: 20 -> +10
 * Croupier: 18, ManoBAB: 23 -> -10
 Resultado de la partida: +30
 ¿Otra partida? [S/N] N


BALANCE FINAL: 45 €



Anexo B – Ejemplo de ejecución del programa en modo análisis

```
*** BLACKJACK - PARADIGMAS DE PROGRAMACIÓN 2023/24
***
¿Modo de ejecución? [J]uego [A]nálisis: A
¿Número de partidas? 3
```

```
--- INICIO DE LA PARTIDA #1 --- BALANCE = 0 €
¿Apuesta? [2] [10] [50]: 10
```

REPARTO INICIAL

Croupier:
(11)
Activa 

Mano:
(21)
10€  


Activa



*** BLACKJACK ***

Ha ganado 15 €!

```
--- INICIO DE LA PARTIDA #2 --- BALANCE = 15 €
¿Apuesta? [2] [10] [50]: 2
```

REPARTO INICIAL

Croupier:
(8)
Activa 

Mano:
(20)
2€  

Activa


TURNO DEL JUGADOR



¿Jugada para Mano? [P]edir [D]oblar [C]errar
[S]eparar C




Mano:
(20)
2€  

Cerrada




TURNO DEL CROUPIER

Croupier:
(8)
Activa 

Croupier:
(10)
Activa  

Croupier:
(20)
Activa   

FIN DE PARTIDA

Croupier:
(20)
Cerrada   


Mano:
(20)
2€  



Cerrada

CONTABILIZACIÓN DE RESULTADOS
* Croupier: 20, Mano: 20 -> +0
Resultado de la partida: +0

```
--- INICIO DE LA PARTIDA #3 --- BALANCE = 15 €
¿Apuesta? [2] [10] [50]: 2
```

REPARTO INICIAL

Croupier:
(8)
Activa 

Mano:
(20)
2€  

Activa


TURNO DEL JUGADOR



¿Jugada para Mano? [P]edir [D]oblar [C]errar
[S]eparar C



Mano:
(20)
2€  

Cerrada


TURNO DEL CROUPIER

Croupier:
(8)
Activa 

Croupier:
(10)
Activa  

Croupier:
(19)
Activa   

FIN DE PARTIDA

Croupier:
(19)
Cerrada   

Mano:
(20)
2€  

Cerrada

CONTABILIZACIÓN DE RESULTADOS
* Croupier: 19, Mano: 20 -> +2
Resultado de la partida: +2

BALANCE FINAL: 17 €