

Tarea 1

Instrucciones

- Antes de comenzar, lea atentamente las instrucciones, los requerimientos de entrega y las preguntas de la evaluación.
- La evaluación es de carácter individual y está sujeta a las normas del código de honor de la asignatura. Cualquier indicio de intervención de otra persona o de cualquier otro acto sancionable como falta al código de honor significará la calificación mínima al promedio de Laboratorio.
- La solución debe desarrollarse directa y completamente en Python y debe entregarse por UVirtual.
- Estudiantes que entreguen fuera de plazo serán calificados con nota mínima en esta evaluación.
- Entregas que se realicen por vías distintas a Campus Virtual serán calificadas con nota mínima. En caso de que la plataforma presente un problema, puede enviar su archivo como respaldo al correo de contacto del profesor de Laboratorio, y posteriormente subir el archivo a Campus Virtual.
- Estudiantes que entreguen sin seguir los criterios de identificación estipulados en el apartado de «entrega» serán calificados con nota mínima en esta evaluación.
- Considere que cualquier supuesto que el estudiante haga debe ser explicitado en los comentarios de código.
- La subida del archivo es responsabilidad de su autor, por lo que archivos que no estén en el formato estipulado, que vengan corruptos o con problemas para ser leídos no serán revisados.

Entrega

Para cada pregunta, cree un archivo .py con su run. Agregue al encabezado de su programa los siguientes datos de identificación. Considere que, de no agregarlos, el puntaje de su pregunta no contará.

```
# FUNDAMENTOS DE PROGRAMACIÓN PARA INGENIERÍA
# FUNDAMENTOS DE COMPUTACIÓN Y PROGRAMACIÓN
# SECCIÓN DEL CURSO:
# PROFESOR DE TEORÍA:
# PROFESOR DE LABORATORIO:
#
# AUTOR
```



NOMBRE:

RUN:

CARRERA:

Suba su solución al apartado especificado en UVirtual en la pestaña «Evaluaciones», apartado «[T1] — TAREA 1».

1 Blackjack++

El Blackjack es un juego de cartas de casinos, donde el objetivo es juntar cartas cuyos valores sumen lo más cercano a 21 posible, sin pasarse. En esta ocasión, vamos a auditar —o sea, revisaremos si fue correcto lo ocurrido— una variante para dos jugadores.

En esta variante, cada jugador cuenta con cinco (5) vidas y el mazo cuenta con 10 cartas, del as al 10, donde el as puede valer 1 u 11, según lo que le convenga al jugador. Al inicio, cada jugador saca dos cartas, una carta a la vez (es decir, el primer jugador saca una carta, el segundo jugador saca una carta, etc.). Se juega por rondas hasta que uno de los jugadores se quede sin vidas. Los jugadores pueden realizar en su turno cada una de las siguientes acciones:

`pedir` Se pide una carta y se le entrega la primera carta del mazo.

`jugar a 17` En esta ronda el objetivo es sumar 17 y no 21.

`jugar a 23` En esta ronda el objetivo es sumar 23 y no 21.

`pedir X` Se le entrega al jugador la carta *X*, si esta está en el mazo. En caso contrario, no se hace nada.

`devolver mi carta` Se devuelve al mazo la ultima carta obtenida por el jugador y se pone al final del mazo.

`devolver carta rival` Se devuelve al mazo la ultima carta obtenida por el rival y se pone al final del mazo.

`desactivar As` En esta ronda, el as vale 11 obligatoriamente.

`robar carta` Se le quita la última carta obtenida por el rival y se agrega a la mano del jugador.

Cada jugador hace su acción turnándose con el otro, es decir, el primer jugador realiza su primera acción, sigue el segundo con su primera acción, continúa el primero con su segunda acción, etc.

La auditoría que realizará tu programa debe verificar quién ganó en cada ronda y quién gana al final. Para esto, el programa debe pedir el orden en el que estaban las cartas en el mazo con un mensaje y luego, las acciones de cada jugador, uno a la vez, pero todas sus acciones en orden separadas por coma. Por ejemplo, si el orden de las cartas del mazo son

6 , 9 , 2 , 4 , A , 5 , 10 , 8 , 7 , 3 , este se ingresaría como:

6,9,2,4,A,5,10,8,7,3

Nota que el as se representa como A y solo nos interesa el valor, no la pinta.














Luego, si el primer jugador hace las acciones de «pedir», «devolver mi carta», «pedir» y «jugar a 17», sus acciones se ingresarían como:











pedir,devolver mi carta,pedir,jugar a 17

Por su parte, si el segundo jugador realiza las acciones de «pedir 3»,«desactivar As», «pedir 8», «devolver mi carta», sus acciones se ingresarían como:

pedir 3,desactivar As,pedir 8,devolver mi carta

Esto significa que el orden y consecuencia de las acciones ejecutadas fue:

1. Jugador 1 saca  del mazo.
2. Jugador 2 saca  del mazo.
3. Jugador 1 saca  del mazo.
4. Jugador 2 saca  del mazo.
5. Jugador 1 pide carta (se le entrega el ).
6. Jugador 2 pide el  (se le entrega el , porque está en el mazo).
7. Jugador 1 devuelve su carta (pone el  al fondo del mazo).
8. Jugador 2 desactiva el  (ahora vale siempre 11).
9. Jugador 1 pide (se le entrega el ).
10. Jugador 2 pide el  (se le entrega el , porque está en el mazo).
11. Jugador 1 pide jugar a 17.
12. Jugador 2 devuelve su carta (pone el  al final del mazo).

Al final, el jugador 1 tiene en su mano , , , mientras que el jugador 2 tiene en su mano , , , quedando el puntaje total de cada mano en 13 y 16, respectivamente, mientras que el mazo quedó con las cartas , , , . El que gana la ronda es el que tiene la menor distancia al objetivo (17, en este caso) sin haberse pasado, es decir, el jugador 2, por lo que el jugador 1 pierde una vida. El programa debe presentar el mensaje

Total: <ptos jugador 1>-<ptos jugador 2>

reemplazando <ptos jugador 1> y <ptos jugador 2> con el puntaje obtenido por el jugador 1 y el jugador 2, respectivamente. Luego, si un jugador gana la ronda, debe mostrar el mensaje:



Ronda <N° de ronda> para jugador <N° jugador ganador>

reemplazando <N° de ronda> con el número de ronda actual (contando desde 1 para la primera ronda) y <N° jugador ganador> con solo el número del jugador que la ganó (es decir, 1 o 2). En caso de empate, ningún jugador pierde vidas y debe mostrar el mensaje:

Ronda <N° ronda> empatada

Finalmente, debe marcar el recuento de vidas con el mensaje:

Vidas: <vidas_jugador1> vs <vidas_jugador2>

reemplazando <vidas_jugadorX> con las vidas del jugador X. Los mensajes de cada ronda deben ser mostrados uno por línea y al final, una línea en blanco ("`\n`") los separa de la próxima ronda. En otras palabras, después del último mensaje de la ronda, debe ir una línea en blanco. Para este ejemplo, las entradas y el resultado se verían como:

```
Ingrese las cartas del mazo: 6,9,2,4,A,5,10,8,7,3
Ingrese la jugadas del jugador1: pedir,devolver mi carta,pedir,jugar a 17
Ingrese la jugadas del jugador2: pedir 3,desactivar As,pedir 8,devolver mi carta
Total: 13-16
Ronda 1 para jugador 2
Vidas: 4 vs 5
```

Una vez un jugador se queda sin vidas, la victoria se la lleva el otro, lo que se debe señalar con el mensaje:

GANA EL JUGADOR <N° del jugador>!

reemplazando <N° del jugador> con el número del jugador ganador.

1.1 Entradas

En cada ronda, se deben pedir las tres entradas en el mismo orden: la secuencia de cartas en el mazo (separadas por coma y usando A para representar el as), la secuencia de acciones del jugador 1 y la secuencia de acciones del jugador 2. Estas se deben pedir con los siguientes mensajes:

```
Ingrese las cartas del mazo: <cartas del mazo>
Ingrese la jugadas del jugador1: <acciones jugador 1>
Ingrese la jugadas del jugador2: <acciones jugador 2>
```

Todas estas secuencias se ingresan separadas por coma y sin espacios entre los elementos.

1.2 Salida

La salida corresponde a la secuencia de mensajes al final de cada ronda, indicando puntajes totales por jugador, quién ganó la ronda (o si fue un empate) y la cantidad de vidas restantes para cada jugador:

```
Total: <ptos jugador 1>-<ptos jugador 2>  
Ronda <N° de ronda> para jugador <N° jugador ganador>  
Vidas: <vidas_jugador1> vs <vidas_jugador2>
```

En caso de empate, solo cambia la línea declarando ganador por el mensaje de empate:

```
Total: <ptos jugador 1>-<ptos jugador 2>  
Ronda <N° ronda> empatada  
Vidas: <vidas_jugador1> vs <vidas_jugador2>
```

Mientras que cuando un jugador se queda sin vidas y se declara un ganador, se muestra el mensaje declarándolo:

```
GANA EL JUGADOR <N° del jugador>!
```

1.3 Consideraciones

Todas las rondas muestran sus mensajes en el orden pedido acá, sin líneas entre mensajes. Solo se separan con una línea en blanco las rondas sucesivas y el mensaje declarando el ganador (mira el ejemplo).

Recuerda poner atención a todos los detalles de la salida. Como la revisión será automatizada, incluso los caracteres invisibles (como espacios o saltos de línea) cuentan para definir si el programa cumple o no. En otras palabras, para el revisador, los strings "hola" y " hola" son salidas diferentes.

1.4 Restricciones

Para el desarrollo de esta tarea, solo puedes utilizar los tipos básicos vistos en clases y las secuencias de control que hemos visto hasta ahora, es decir, los contenidos de la Unidad 1 de la asignatura.

1.5 Ejemplo

A continuación, veremos un ejemplo de ejecución. Este ejemplo requirió seis rondas para declarar como ganador al jugador 2, quien quedó con 4 vidas. Ten cuidado si vas a copiar las entradas para probar el código, pues el documento corta las líneas muy largas para mejorar la visualización.

```
Ingrese las cartas del mazo: 6,9,2,4,A,5,10,8,7,3  
Ingrese la jugadas del jugador1: pedir,devolver mi carta,pedir,jugar a 17  
Ingrese la jugadas del jugador2: pedir 3,desactivar As,pedir 8,devolver mi carta  
Total: 13-16
```



Ronda 1 para jugador 2

Vidas: 4 vs 5

Ingrese las cartas del mazo: 9,10,8,7,2,A,6,5,4,3

Ingrese la jugadas del jugador1: pedir,robar carta,pedir 5,jugar a 23

Ingrese la jugadas del jugador2: devolver carta rival,pedir 5,devolver mi

↪ carta,pedir

Total: 24-21

Ronda 2 para jugador 2

Vidas: 3 vs 5

Ingrese las cartas del mazo: 6,3,A,4,10,9,2,5,8,7

Ingrese la jugadas del jugador1: pedir 10,desactivar As,devolver carta rival,pedir

↪ 10

Ingrese la jugadas del jugador2: pedir,devolver mi carta,devolver mi carta,pedir 2

Total: 27-2

Ronda 3 para jugador 2

Vidas: 2 vs 5

Ingrese las cartas del mazo: 10,2,7,3,9,6,8,A,4,5

Ingrese la jugadas del jugador1: desactivar As,jugar a 17,pedir 8,jugar a 17

Ingrese la jugadas del jugador2: pedir 8,jugar a 23,jugar a 17,desactivar As

Total: 17-13

Ronda 4 para jugador 1

Vidas: 2 vs 4

Ingrese las cartas del mazo: 7,10,9,2,6,A,4,5,8,3

Ingrese la jugadas del jugador1: devolver carta rival,devolver mi carta,pedir

↪ 4,desactivar As

Ingrese la jugadas del jugador2: devolver carta rival,devolver mi carta,pedir

↪ 7,robar carta

Total: 0-11

Ronda 5 para jugador 2

Vidas: 1 vs 4

Ingrese las cartas del mazo: 8,6,A,7,9,5,10,3,2,4

Ingrese la jugadas del jugador1: devolver carta rival,pedir 8,pedir 6,devolver mi

↪ carta

Ingrese la jugadas del jugador2: desactivar As,jugar a 23,pedir,pedir

Total: 8-20

Ronda 6 para jugador 2

Vidas: 0 vs 4

GANA EL JUGADOR 2!