

Informe proyecto 1 de Introducción a la Programación: Video Poker

El Video Poker representó un gran desafío para mí porque yo no juego cartas. Muchas veces en la vida se me presentaron oportunidades para jugar estos juegos, pero nunca les otorgué mayor importancia y menos pensé que algún día “me pasarían la cuenta.” Desde el comienzo, el desafío número uno fue aprender a jugar poker, desde las reglas básicas hasta saber qué eran las pintas y cuáles eran los famosos jacko, queena, rey y, por supuesto, la carta As.

Varias manos de Video Poker online, preguntas a la profesora, amigas y amigos más tarde, me senté frente al computador a programar el proyecto. Por supuesto, debido a mi falta de experiencia con el juego, cometí errores que después significaron comenzar de nuevo o reprogramar varias funciones. Es más, en mi carpeta de proyecto terminé con ocho versiones de este (e incluso algunas como proyecto 1 versión 8.2, 8.3, hasta llegar a 8.final).

Recuerdo que el primer problema que se me presentó fue cómo lograr que el programa recordara cuales eran las cartas que el usuario seleccionaba para intercambiar. Después de pensar unos días, se me ocurrió usar las variables a,b,c,d y e, las cuales tenían los valores 10,20,30,40 y 50 respectivamente. De esta forma, tras almacenar el valor de `poker.esperarClick()` en una variable “juego”, a,b,c,d, y e cambiaban sus valores a 0,1,2,3 y 4 respectivamente. Finalmente, usando “if”, “elif” y el comando `poker.cambiarCarta(pos, carta)`, logré intercambiar las cartas seleccionadas por el usuario. Así, por ejemplo, si `a==0`, ejecutaba `poker.cambiarCarta(a, carta)`.

El segundo mayor problema fue encontrar una forma de organizar mis cartas. Solucioné lo anterior creando funciones. Inventé las funciones llamadas `numerocarta(c)`, `pintacarta(x)` y `letracarta(x)`. Las dos primeras recibían las variables c1, c2, c3, c4 y c5 que guardaban el valor de `poker.sacarCarta()` para la primera, segunda, tercera, cuarta y quinta cartas respectivamente. De esta forma, según los valores de las variables antes mencionadas, `numerocarta(c)` y `pintacarta(x)` entregaban el número de la carta, desde 1 hasta 13, y `pintacarta(x)` la pinta, desde 1 hasta 4. Luego, hice la combinación `letracarta(pintacarta(c1))`, por ejemplo, y según el valor de la pinta (1, 2, 3 ó 4), la función devolvía “c”, “p”, “t” y “d”.

Pensando que todo iba de maravilla, continué el proyecto sin mayores problemas: definí funciones para cada combinación ganadora, para la duplicación del premio (función `dup`) e incluso una para cuando el usuario no quisiera duplicar su premio (función `dupfalse`). El mayor problema, sin embargo, se presentó al final, cuando me di cuenta que, al perder el doble o nada, el usuario mantenía su premio y, lo que era peor, ni siquiera perdía la apuesta. Esto se veía imposible de resolver debido a cómo había definido las funciones, pues se restaba la apuesta del crédito al perder internamente.

Tras horas de inspiración frente al computador y escuchar Poker Face un par de veces, pensé en crear dos contadores, uno para cuando ganara una duplicación (`itdup`) y otra para cuando empatara el doble o nada (`itsum`). Luego, definí la variable `pozo`, inicialmente como `pozo=0`, pero dentro del ciclo de duplicación como `pozo=(premio*2*itdup)+(itsum*premio)`. Los contadores iban aumentando en uno cada vez que el usuario ganaba o empataba el doble nada. De esta forma, al perder, programé que el crédito final fuese el resultado de restar la apuesta y el pozo al crédito “que hubiese tenido el usuario si hubiese ganado”, y, ¡Funcionó! ☺

Sección opcional

Es confuso para mí escribir en esta sección puesto que, como estaba aprendiendo poker por primera vez y muchas veces me confundí, aprendí más cosas del juego a la hora de programar y otras que nunca entendí muy bien o que, lamentablemente, comprendí al final. No sé por qué los juegos de cartas tienen un don especial para confundirme y hacerme no retener nada o casi nada de lo recién visto; me “enredo”. Lo que intento decir es que puede ser que haya incluido cosas de más sin saber que lo eran. Un ejemplo del que estoy consciente, y que es el que quería destacar, son las escalas.

Debido a que, en un principio, me guié estrictamente por las ilustraciones del enunciado del Video Poker, pensé que las escalas tenían que ser, por ejemplo, 1,2,3,4,5. Luego, me comentaron que podía ser, en el mismo caso anterior, 5,4,3,2,1. Pensando solucionar el problema, porque cuando buscaba en internet salían varios tipos de poker y premios, le pregunté a mi profesora, quien solo me confirmó todo lo anterior, además de ratificarme que otra combinación posible era, por ejemplo, 2,4,3,5,1.

Por lo tanto, agregué en la función “escala” todas las combinaciones posibles, hasta el grupo 9,10,11,12,13. Tiempo después, me enteré que las combinaciones 10,11,12,13; 11,12,1,2,3; 12,13,1,2,3 y 13,1,2,3,4 eran escalas también y, una de ellas, tenía incluso un nombre especial: la escala real. Al final, incluí todas las escalas, pero me guié estrictamente por los premios que se encontraban en el enunciado, aunque, según lo que pude averiguar, la escala real tenía premios distintos. Sin embargo, debida a mi inexperiencia con el poker, no quise poner algo equivocado o confundirme innecesariamente.

Por último, hay un punto que no puedo clasificar como una dificultad respecto al código en sí, ni como algo que tuve dificultad para resolver, ni tampoco como algo “extra” que incluí, sino que algo que surgió debido a mi escaso conocimiento del poker. Personalmente, veo el tema que trataré a continuación como una “característica especial” del juego que programé y es que cuando el usuario aumenta sus fichas apostadas, no disminuye el crédito.

A pesar de la impresión inicial que esto puede crear, no veo dicha “característica” como algo incorrecto puesto que, luego de jugar poker online otras cuantas veces, me di cuenta que mi programa, por como lo había configurado con las funciones que creé, llegaba al mismo resultado que la versión en videopokercity. Es decir, si ganaba, recuperaba las fichas, y si perdía, las perdía y se disminuía la apuesta. Todo esto se reflejaba en los créditos finales del usuario, que fuese que disminuyeran o no los créditos al principio, el resultado era el mismo. Por lo tanto, dejé el programa igual, puesto que para mí, personalmente, lo que importa es el resultado final en estos juegos.