
Llenguatges de Programació

Curs 2006 – 07 / Bloc #2 - Pràctica #4

Avaluació del Bloc 2



Condicions de lliurament:

És MOLT IMPORTANT que llegiu i enteneu bé les condicions de lliurament, per evitar malentesos el dia de la correcció al laboratori. Penseu que **NO S'ACCEPTARÀ CAP PRÀCTICA QUE NO SEGUEIXI LES NORMES** que especifiquem a continuació:

1. Heu de fer el lliurament a la sessió de pràctiques del grup on esteu apuntats.
2. TOTS els components del subgrup, hauran de venir el dia de la correcció.
3. La pràctica de cada subgrup, haurà d'estar penjada al campus virtual, abans de les 24:00h. del dia anterior a la correcció (vegeu taula adjunta).
Donat que només podeu penjar-ho una vegada, aneu amb MOLT de compte de penjar la versió correcta, ja que la versió penjada (i no cap altra) serà la que descarregareu i executareu el dia de la correcció.
Tingueu en compte que si la vostra pràctica consta de varis fitxers, haureu de penjar un fitxer comprimit .zip.
4. La (poca) gent que no tingui accés al Campus Virtual s'haurà de posar en contacte amb el/la vostre/a professor/a de pràctiques, amb uns dies d'antelació al lliurament, per determinar la forma de lliurament.

Nº de grup	Data Límit de lliurament al Campus Virtual
Grups 1, 2, 3, 4	Diumenge 17 de desembre a les 23:59
Grups 6, 7, 8	Dilluns 18 de desembre a les 23:59
Grups 10, 11	Dimarts 19 de desembre a les 23:59

1. Introducció

Després de l'èxit del joc 'La Ruleta, v.1.0' que va fer al bloc 1, el propietari del nostre Casino Virtual està molt interessat en incorporar algunes millores al joc. Per tant, en aquesta pràctica realitzareu la versió 2.0 del joc de la Ruleta. Les millores que se'ns demanen per la nova versió es poden resumir en els següents punts:

- Solucionar tots els 'bugs' (errades de funcionament) detectats en la fase de prova (entrega del bloc 1) de la versió 1.0.
- Per augmentar la rendibilitat del negoci, s'ha de permetre que puguin jugar múltiples jugadors simultàniament.
- Per evitar els problemes que donen els diners en efectiu, cal implementar el sistema de fitxes electrònic (veieu el projecte conjunt d'AP-LP).

En els següents apartats es detallen les especificacions de la versió 2.0 de 'La Ruleta' que heu d'implementar. És important que ho llegiu amb atenció perquè el programa ha de fer exactament el que se us demana i no cap altra cosa.

2. La Ruleta, versió 2.0

En aquesta segona versió de La Ruleta, quan entreu al programa, ha de sortir un menú amb les següents opcions:

1. Registrar jugador
2. Fer partida
3. Sortir

i s'ha de demanar que se'n triï una. Si es tria una opció no vàlida s'ha d'indicar amb un missatge per pantalla. El programa ha de permetre que es vagin triant tantes opcions com es vulgui fins triar l'opció "Sortir".

Opció "Registrar jugador al Casino"

Si es tria l'opció "Registrar jugador" s'ha de procedir a demanar les dades del jugador que es vol registrar i emmagatzemar la informació en la base de dades. Les dades que cal demanar, són les dades personals que es demanaven a la versió 1.0:

1. Número de DNI
2. Nom i cognoms
3. Adreça
4. Data de naixement

i, a més, ara també s'ha de demanar la quantitat de diners que es vol canviar per fitxes electròniques. Per tant, cal demanar una dada més:

5. Diners que es vol canviar per fitxes

El Casino aplicarà una comissió al canvi de diners, de forma que a la quantitat de diners que el jugador vulgui canviar per fitxes, se li descomptarà el percentatge corresponent a la comissió.

Opció "Fer partida"

Cada cop que es triï l'opció "Fer partida" cal demanar a cada jugador registrat l'aposta que vol fer, és a dir, cal preguntar la quantitat (en fitxes) que vol apostar i permetre que faci la seva aposta sobre el taulell verd. De moment, les apostes que es podran realitzar a La Ruleta 2.0 són les mateixes que a la versió 1.0:

- Senzilla
- Doble
- Quadrat
- Columna

Cap jugador pot quedar-se a números vermells dins la seva fitxa electrònica. Per tant, si un jugador intenta fer una aposta per un import superior al que disposa, se li indicarà amb un missatge que no té fitxes suficients per fer l'aposta i se li mostrarà quin és el valor de les fitxes de què disposa. Per tant, la pregunta de la quantitat que vol apostar es repetirà fins que introdueixi un valor inferior a la quantitat de diners de què disposa.

Una vegada el jugador ha fet la seva aposta, cal emmagatzemar quins són els números inclosos en l'aposta realitzada i mostrar per pantalla un missatge indicant quins són tots aquests números. També cal restar de la seva fitxa electrònica, la quantitat en fitxes que ha apostat.

Quan tots els jugadors registrats han fet la seva aposta, el programa ha de generar aleatòriament el número guanyador. Aleshores, s'ha de comprovar, per cada jugador, si ha

guanyat o ha perdut i s'ha d'actualitzar la seva fitxa electrònica. És a dir, a aquells jugadors que hagin guanyat se'ls ha de sumar la quantitat del premi. A més, mostrarem per pantalla un missatge per cada jugador indicant el seu nom i la quantitat que ha guanyat o ha perdut:

```
Jugador XXXX XXXX XXXX ha guanyat 200 euros
Jugador YYY YYY YYY ha perdut 700 euros
```

Opció "Sortir"

Si es tria l'opció "Sortir", s'ha de mostrar per pantalla un llistat amb la informació de tots els jugadors registrats. En aquest llistat, a cada línia hi ha d'haver la informació d'un jugador amb el següent ordre:

```
Número_DNI    Nom Cognoms    Adreça    Data_naixement    Diners
```

Una vegada mostrat el llistat amb les dades dels jugadors, el programa s'acaba.

3. Detalls d'implementació

Per implementar la segona versió del joc, podeu aprofitar la major part del codi que teniu de la versió 1.0. De tota manera, haureu de fer algunes modificacions en alguna part per adaptar-ho a les especificacions de la versió 2.0.

Recordeu els consells d'implementació que se us van donar en l'enunciat de la pràctica d'avaluació del Bloc 1 i que teniu resumits a continuació:

- Utilitzeu el model de pantalla completa per executar el programa (ALT+Intro).
- Utilitzeu registres (`struct`) per emmagatzemar les dades dels jugadors i les apostes.
- Per definir els nous tipus de dades s'ha d'utilitzar el `typedef`.
- Per a realitzar les apostes se us proporcionen algunes funcions en tres fitxers que heu d'incloure en el vostre projecte (RutinesB2.c, RutinesB2.h i MouseUTILS.h).
- La funció "ApostaRuleta" us mostra el taulell per pantalla i permet clicar a sobre per fer una aposta (Vegeu els detalls de funcionalment a la pràctica del Bloc 1).
- La funció "NumAleatori" us permet generar un número aleatori. Per a que funcioni correctament cal posar la següent instrucció al principi de la funció `main()`:

```
srand( (unsigned)time( NULL ) );
```

i al principi del programa l'inclou:

```
#include <time.h>
```

- Per esborrar la pantalla podeu fer servir la instrucció `system` de la llibreria `stdlib.h`:
`system("cls")`
- Abans de fer la lectura d'una cadena de caràcters cal posar la instrucció:

```
fflush(stdin);
```

A continuació us detallarem alguns punts del funcionament del programa i us expliquem com implementar algunes parts de la pràctica.

Estructures de dades

Les estructures de dades que havíeu de definir a la versió 1.0 (un registre per la informació del jugador i un registre per la informació de l'aposta) ens serviran de base per la versió 2.0.

Al registre del jugador, haurem d'afegir un camp per guardar la quantitat de diners en fitxes que té cada jugador. A més, com que ara hi haurà múltiples jugadors i cada jugador farà una aposta, al registre del jugador, també haurem d'afegir un camp per guardar la informació de l'aposta. Si teniu definit un tipus registre per l'aposta, simplement caldrà que afegiu un camp d'aquest tipus al jugador.

El número de jugadors registrats en cada moment, no el podem saber a priori. Per tant, necessitem definir una estructura dinàmica per emmagatzemar la informació dels jugadors registrats. L'estructura que hem d'implementar és una llista simplement encadenada que, de moment, no ha d'estar ordenada. Aquesta llista, serà la base de dades del Casino.

Per tant, per utilitzar el registre del jugador com a node bàsic de la llista encadenada, també necessitarem afegir un camp 'següent' (podeu donar-li el nom que vulgueu) que sigui un apuntador al següent node de la llista.

IMPORTANT – Recordeu que als apunts de l'assignatura "Algorismes i Programació" teniu el pseudo-codi de les funcions necessàries per treballar amb la llista encadenada. Haureu de fer alguna petita modificació per adaptar-lo a la vostra estructura de dades. Bàsicament, les funcions que us poden ser d'utilitat són:

```
void Crear_Llista_Buida(Jugador **primer)

int AfegirPrimer (Jugador **primer, Jugador *jug)

int AfegirUltim (Jugador **primer, Jugador *jug)

int EliminarPrimer(Jugador **primer)

int EliminarUltim(Jugador **primer)
```

on Jugador és el tipus definit com a node de la llista.

Registrar jugador

En registrar un jugador, heu de demanar les dades personals i els diners que vol canviar per fitxes. Segons teniu a l'enunciat del Projecte conjunt d'AP-LP, la fórmula per fer el canvi de diners a fitxes del casino és:

$$\text{Valor en fitxes} = \text{Valor diners} \times (100 - \text{Percentatge Comissió}) / 100$$

El percentatge de la comissió que es quedarà el Casino en cada canvi serà del 5%. Per tant, la fórmula per fer aquest càlcul queda:

$$\text{Valor en fitxes} = \text{Valor diners} \times 0.95$$

Emmagatzemar tota la informació del jugador a la base de dades equivaldrà a fer una inserció a la llista. Donat que la llista no cal que estigui ordenada, podeu fer la inserció sempre pel principi o pel final.

Fer partida

L'opció de fer partida, implicarà fer dos recorreguts a la llista de jugadors. La primera per a que cada jugador faci la seva aposta. La segona, després de generar el número guanyador, per comprovar qui ha guanyat i actualitzar els diners de la fitxa de cada jugador.

Sortir

Per mostrar per pantalla la informació de tots els jugadors registrats, cal recórrer la llista i anar mostrant les dades emmagatzemades a cada node fins que arribem al final de la llista.

Donat que estem treballant amb memòria dinàmica, cal alliberar tota la memòria que hem reservat al llarg del programa. Recordeu que per alliberar la memòria utilitzada per una llista encadenada, no hi ha prou amb alliberar el primer node, si no que cal recórrer la llista i anar alliberant node a node.

Per estalviar temps i operacions, no recorreu dos cops la llista per mostrar la informació i alliberar. Feu un únic recorregut i a mesura que mostreu la informació de cada node, l'allibereu.

4. Avaluació

Us recordem que, a l'igual que en la pràctica del bloc 1, en l'avaluació de la pràctica és valorarà:

- El correcte funcionament del programa
- El correcte ús del llenguatge C
- Resposta a preguntes sobre el codi

Els dos primers punts són igual d'importants. Si una pràctica funciona correctament però no està ben implementada no s'aprovarà. D'igual forma, si una pràctica està ben implementada però no funciona bé, tampoc estarà aprovada.

MOLT IMPORTANT: La informació dels jugadors s'ha de guardar en una llista encadenada dinàmica. Aquest punt és imprescindible per aprovar la pràctica.

Les preguntes sobre el codi s'han de respondre correctament per demostrar que tots els components del grup han participat en la implementació de la pràctica entregada.

Altres punts que es tindran en compte positivament:

- Claredat i estructura del codi: tabulació del codi, separacions entre blocs del programa, inclusió de comentaris, noms dels identificadors de les variables...
- Presentació per pantalla: missatges clars, la pantalla s'esborra cada cop que es mostra el menú o s'entra a una opció, etc...
- Control d'errors en l'entrada de dades (p.ex. controlar que la data de naixement sigui vàlida)
- Millores sobre el que es demana per facilitar el funcionament del programa

No es valoraran:

- Presentacions complicades per pantalla: dibuixos, colors, etc...
- Implementació de punts de la pràctica final que encara no es demanen.

Abans de lliurar la pràctica proveu que funciona correctament en tots els casos. Proveu totes les possibilitats per assegurar que funciona bé sempre.