

SNAKE TIMER

EASY68K

Maria Kupriyenko Chupaida - 49779924N

Joan Jiménez Rigo - 45613706Z

1. Introducció

En aquest document es presenta el desenvolupament del videojoc SNAKE TIMER amb Easy68K, implementat com a part de la pràctica de l'assignatura Estructura dels Computadors II. Aquest projecte té com objectiu principal l'aplicació dels conceptes de programació en assembleador adquirits en classe i la gestió de perifèrics.

El joc consisteix en una versió adaptada del clàssic joc de la serp, on el jugador controla una serp i augmenta la puntuació a mesura que menja pomes. El joc acaba passat un temps determinat o quan colisiona amb els costats de la pantalla.

Aquest informe està estructurat per oferir una visió completa del projecte, començant amb una descripció de l'organització del codi, els principals desafiaments trobats durant el desenvolupament i les millores realitzades al codi proporcionat. També inclou una breu guia per jugar, juntament amb un llistat de les característiques implementades i la seva integració tècnica.

2. Estructura del Codi

2.1 Organització dels fitxers

- Carpeta LIB: Fitxers UTLCODE, UTLVARS, UTLSCONST proporcionats pel professor
- Carpeta SND: arxius .WAV amb els sons GAMEOVER i EAT
- Fitxer APPLE: Lògica relacionada amb la poma
- Fitxer CONST: Constants del joc
- Fitxer GAME: Inicialització, actualització i dibuixat del joc
- Fitxer GAMEOVER: Pantalla de game over
- Fitxer HEADSRP: Lògica del cap de la serp, colisions i ús de fitxer
- Fitxer INSTRUCTIONS: Pantalla d'instruccions
- Fitxer INTRO: Pantalla d'introducció
- Fitxer MAIN: Programa principal
- Fitxer MATRIZ: Lògica relacionada amb la matriu
- Fitxer SCORE: Lògica de la puntuació del joc
- Fitxer SEGSRP: Lògica relacionada amb els segments de la serp
- Fitxer STATES: Codi proporcionat pel professor per la gestió dels estats de les pantalles

- Fitxer SYSCONST: Constants del sistema
- Fitxer SYSTEM: Lògica del sistema, sons i teclat
- Fitxer SYSVARS: Variables del sistema
- Fitxer TABLE: Lògica de la pantalla de puntuacions
- Fitxer TAILSRP: Lògica de la cua de la serp
- Fitxer VARS: Variables del joc
- Document PUNTOS.txt: Fitxer on es guarda la puntuació

2.2 Funcions principals i les seves relacions

Al programa principal es troba: els INCLUDE dels fitxers utilitzats, la inicialització del sistema, del UTLCODE i del fitxer STATES i un bucle on es troba la lectura del teclat, l'actualització del joc, la sincronització, el dibuixat i el doble buffer.

Al fitxer sistema estan les subrutines de sistema, del so i del teclat.

Al fixter STATES, STAUPD gestiona la inicialització i l'actualització de les diferents pantalles del joc. STAPLOT gestiona el dibuixat d'aquests.

El joc compta amb 5 pantalles: introducció, amb inicialització, actualització (on es troba la gestió del ratolí) i dibuixat; instruccions, amb les mateixes subrutines anteriors; joc, on es troba la gestió del joc; game over, amb les tres subrutines, i puntuacions registrades, amb actualització i dibuixat.

Al fitxer joc es troba la gestió de la serp, de la poma, de la matriu i de la puntuació.

Per desenvolupar el joc s'ha creat una matriu amb cel·les inicialitzades a 0 on les cel·les ocupades amb la serp o la poma s'han marcat amb altres valors. En aquest fitxer, a part de la inicialització, actualització i dibuixat de la matriu, també es troben les subrutines necessàries per calcular l'índex d'una cel·la, marcar la cel·la com ocupada o buida, obtenir les coordenades x i y d'una cel·la, comprovar si una està ocupada i dibuixar el fons del joc.

La serp s'inicialitza al principi al centre de la pantalla. El moviment s'ha implementat de forma que el cap avança amb velocitat i la resta de segments actualitzen la seva posició en funció del segment anterior. La comprovació amb el teclat es fa al el cap, que depenent de la tecla polsada canvia ESTDCBZ i la seva direcció. El cap també comproba les colisions amb els costats de la

pantalla i amb la poma, que canvia a una altra posició aleatòria, reproduint els sons corresponents. A més, hi ha la tasca de llegir fitxers.

El fitxer dels segments els inicialitza, actualitza les seves posicions i dibuixa dos segments i el fitxer cua es comporta de manera similar.

SCORE té les tres subrutines principals que gestionen la puntuació i el temps del joc, dibuixats amunt a l'esquerra de la pantalla.

Finalment, el fitxer de la poma utilitza el codi proporcionat pel professor per generar una posició aleatòria de la pantalla i dibuixar-la.

3. Principals Dificultats

Originalment el joc es va pensar com l'original on la serp augmenta de tamany cada vegada que menja la poma però la dificultat de trobar la manera d'implementar-ho, la falta de temps i la probabilitat que hauríem de reestructurar per complet el codi que ja teniem van fer que canviéssim l'enfocament.

L'altra dificultat principal que vam passar va ser la captura i gestió del moviment de la serp amb les fletxes del teclat.

Els altres aspectes del desenvolupament els vam poder implementar gairebé sense molts retrastos de temps.

4. Principals Afegits

- Les pantalles de menú s'han modificat per implentar la tasca del ratolí i per afegir altres tecles de canvi d'estat.
- Per utilitzar UTLRND s'ha creat una subrutina a APPLE per generar una posició aleatòria de la matriu.
- Al fitxer STATES s'han modificat les pantalles que implementa el nostre joc.
- En general s'ha seguit l'estructuració amb els encapçalats recomanats pel professor.

5. Manual d'usuari

El joc comença amb la pantalla d'introducció on hi ha dos opcions: un botó “CLICK HERE TO START” on s'ha de prémer per començar a jugar i prémer el botó ‘P’ fa que es mostri la pantalla amb les puntuacions registrades. Si passa un temps on el jugador no fa res, es mostra la pantalla d'instruccions on s'ha d'esperar per tornar a la pantalla principal.

Per jugar s'ha de controlar la serp amb les fletxes del teclat intentant menjar totes les pomes possibles abans que el temps s'acabi. Si la serp colisiona amb els costats de la pantalla, el joc s'acaba.

En la pantalla de GAME OVER per tornar a la pantalla principal, es pot esperar un cert temps o prémer el botó ‘S’.

Finalment, en la pantalla de puntuacions s'ha de prémer el botó ‘S’ per tornar a la pantalla d'introducció.

6. Característiques escollides

6.1 Ús de fitxers

Una de les característiques que hem triat és utilitzar fitxers per fer una taula de puntuacions. Per dur-ho a terme, vam utilitzar l'estratègia del centinel·la per saber quan acaba un bloc de dades i quan es que acaba el fitxer. El caràcter que hem utilitzat per al bloc de dades és ‘?’, i per al final de fitxer és ‘#’. És així com vam decidir utilitzar aquesta estratègia perquè així era més fàcil per a nosaltres detectar aquests elements. Per tant, el que feiem és comprovar si el fitxer existeix i, si és així, llegim fins a trobar el ‘#’. Una vegada ho trobem ho sobreescrivim amb la puntuació a escriure seguit d'un ‘?’ i un ‘#’. I per a llegir, cada vegada que trobem un ‘?’ imprimim per pantalla fins a trobar un ‘#’.

6.2 Ús del ratolí

L'altre característica que hem decidit afegir és l'ús del ratolí per navegar al menú d'introducció. Per aconseguir-ho hem utilitzat les tasques 60 i 61 del TRAP #15 que habilita l'entrada del ratolí i obté les coordenades d'aquest juntament amb la pulsació del botó esquerre. Després, hem calculat i guardat les coordenades del text on s'ha de pulsar i les hem comparades amb les coordenades del ratolí. Si coincideixen, el joc comença.

7. Conclusions

La realització d'aquesta pràctica ens ha permès aprofundir en la programació en ensamblador i entendre millor els reptes associats al treball amb recursos limitats i l'accés directe al maquinari. Hem après a estructurar el codi, implementant funcionalitats complexes, l'ús de TRAPs, gestió de perifèrics, a desenvolupar l'estructura d'un joc, entre altres.

Tot i les dificultats trobades, hem aconseguit un projecte funcional que compleix els requisits establerts.