

Universidad Rafael Landívar
Facultad de Ingeniería
Curso de Microprogramación
Sección: 01



MANUAL DE USUARIO

Proyecto Práctico 01: Snake

Integrantes:

Carlos Lam 1193916
Max Díaz 1146916
Alexander Solorzano1243717
Salvador Grave 1321013
Adolfo López 1203612

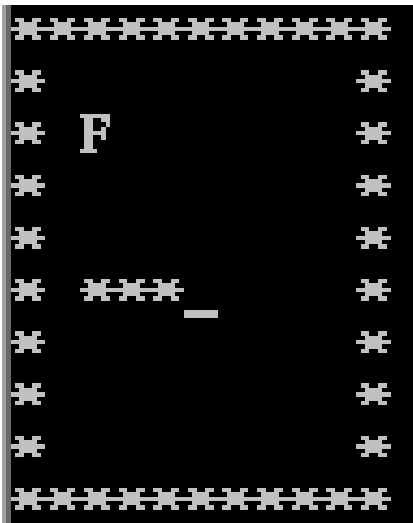
Ciudad de Guatemala, 17 de octubre del 2019.

Objetivo

El objetivo del proyecto es aplicar los conocimientos aprendidos en clase para realizar el juego Snake. El usuario controla una criatura semejante a una serpiente que se maneja alrededor de un plano delimitado, recogiendo alimento, tratando de no golpear su propio cuerpo o las paredes del juego.

Juego

1. Tablero



Fuente: Propia.

2. Inicio del juego

- 1) Al iniciar se mostrará una pantalla de consola la cual contiene el tablero de juego que esta bordeado por el carácter de asterisco.
- 2) En el extremo superior izquierdo estará ubicada la manzana que debe comer la serpiente.
- 3) En el centro del tablero estará ubicado la serpiente del juego, el cual el usuario podrá mover en las direcciones permitidas: izquierda, derecha, arriba y abajo.
- 4) Después de consumir la manzana la serpiente crecerá en tamaño.

3. Movilidad

Son todas las acciones permitidas al usuario que puede realizar en todo momento durante la ejecución del juego:

- W:** Desplazamiento hacia arriba.
- S:** Desplazamiento hacia abajo.
- D:** Desplazamiento hacia la derecha.
- A:** Desplazamiento hacia la izquierda.
- X:** Salir del juego en cualquier momento

Restricciones:

- Si el usuario intenta retroceder se mostrará un mensaje de movimiento no permitido.

4. Fin del juego

El fin del juego se puede dar por los siguientes casos:

- 1) Si el usuario presiona la tecla 'X'.
- 2) Si la cabeza de la serpiente toca uno de los bordes del tablero.
- 3) Si el usuario realiza movimientos que lo lleven a colisionar con la cola de la serpiente.

En los casos anteriores, el juego mostrara en pantalla un mensaje que el juego ha terminado. Esperara a que el usuario presione una tecla y cerrara el programa.

Desarrollo del juego

El juego de la serpiente fue programado en lenguaje ensamblador. Se utilizaron instrucciones, operadores, directivas las cuales son:

1. Directivas

Segmentos simplificados

.MODEL *modelo*: Debe estar ubicada antes de otra directiva de segmento.

El *modelo* utilizado es el siguiente:

- **SMALL**: Los datos caben en un segmento de 64 KB y el código cabe en otro segmento de 64 KB.

.STACK *Define el segmento de pila de la longitud especificada.*

.CODE Define el segmento de código.

.DATA: Define un segmento de datos NEAR con valores iniciales

El siguiente símbolo está definido cuando se usan las directivas anteriores:

- **@data**: Nombre del segmento definido con la directivas **.DATA**, **.DATA?**, **.CONST** y **.STACK** (los cuatro están en el mismo segmento).

Definición de datos

Ubica memoria para un ítem de datos y opcionalmente asocia un nombre simbólico con esa dirección de memoria y/o genera el valor inicial para ese ítem.

DB define datos que pueden ser cadenas o una expresión numérica por medio de bits.

DUP define una cantidad de datos y la repetición del valor que debe contener (el valor que queremos que contenga va entre paréntesis)

Definición de segmento

Organizan el programa para utilizar los segmentos de memoria.

Control del ensamblador

END: Debe ser la última sentencia del código fuente.

2. Registros

Los registros de datos, también llamados “de propósito general”, como su nombre lo indica tienen generalmente datos. Aunque tienen distintos nombres, cuentan con básicamente con la misma funcionalidad.

AX es a menudo llamado acumulador.

BX se puede usar como registro base en algunos modos de direccionamiento, es decir, para apuntar a posiciones de memoria con él.

CX es usado por algunas instrucciones como contador (en ciclos, rotaciones)

DX o registro de datos; a veces se usa junto con AX.

Cada registro de estos está dividido a su vez en dos registros de 8 bits, que pueden ser leídos o escritos de manera independiente:

AX = AH AL	BX = BH BL
CX = CH CL	DX = DH DL

En donde AH es la parte alta del registro AX y AL es la parte baja del registro AX. (H viene de High y L de Low)

DS: registro de segmento adicionales, el primero llamado de datos (Data) y el segundo Extra. Con ellos apuntaremos a los segmentos donde tengamos nuestros datos (ya que del código se encarga CS), esto es, nuestras variables.

SI: registro índice, Empleado para direccionar datos fuente en forma indirecta y utilizarlos con las instrucciones de cadenas o arreglos.

Instrucciones de transferencia de datos

MOV destino, fuente. Copia el contenido del operando fuente en el destino.

Instrucciones aritméticas

INC destino. Incrementa en uno el destino.

DEC destino. Decrementa en uno el destino.

Instrucciones lógicas

XOR destino, fuente. Realiza la operación lógica XOR, almacenando el resultado en destino.

Instrucciones comparativas

CMP destino, fuente. Realiza una operación (destino – fuente)

Instrucciones de transferencia de control

CALL etiqueta. Llama a etiqueta

RET Retorno de procedimiento.

Instrucciones de saltos

JMP etiqueta. Salto incondicional hacia etiqueta.

JZ dir/ **JE** dir Saltar si el resultado es cero ($Z = 1$).

JNE dir. Saltar a *dir* si no es igual ($Z=0$).

Interrupciones

INT n. Ejecuta el manejador de la interrupción especificada en el operando.

LOOP etiqueta. Salta a etiqueta si $CX \neq 0$, decrementando CX.

Operadores analíticos

OFFSET Devuelve el desplazamiento de una ubicación de memoria.

3. Interrupciones

AH	Interrupción	Propósito	Descripción
03H	10H	Lectura del cursor	Utilizada principalmente para limpiar la pantalla al inicio del programa
09H	21H	Impresión de cadena	Muestra una cadena de caracteres en pantalla
07H	21H	Entrada de consola	Espera hasta recibir un carácter desde el teclado. Lo entrega en AL
02H	21H	Exhibe una salida	Imprime un carácter en pantalla utilizando el registro DL
02H	10H	Posición del cursor	Se define la posición en donde queremos colocar el cursor y su número de página
4CH	21H	Terminar proceso	Finaliza el proceso de ejecución del programa
06H	10H	Desplazar líneas de texto hacia arriba	Desplaza hacia arriba un número determinado de líneas en la ventana especificada mediante los registros CX y DX. Las líneas desplazadas, quedan vacías, rellenándose con blancos. El color utilizado en estas líneas vacías se indica mediante el registro BH.
08H	10H	Lee atributo y carácter en el cursor	BH = página en exhibición, AL = carácter y AH = atributo.