

Instituto Tecnológico de Costa Rica

Sede Interuniversitaria Alajuela

Arquitectura de Computadoras

II Tarea Programada

Estudiantes:

Josue David Rodríguez Alfaro

Luis Alejandro Salas Rojas

Fecha de entrega

10 de Junio

2016

II Proyecto Programado: La liebre y los perros cazadores

El Juego de la liebre y los perros cazadores es un juego de estrategia abstracta para dos jugadores, donde uno representa a la liebre y el otro a 3 perros cazadores. El primero intenta escabullirse de los perros mientras que el otro jugador procura acorralarla.

Este proyecto programado consiste en la implementación de un software para jugar «La liebre y los perros cazadores» utilizando lenguaje ensamblador nasm para Intel x86 en GNU/Linux. El objetivo es afianzar los conceptos y técnicas de programación estudiados durante el curso, específicamente: estructuras de control, subrutinas, uso de la pila, manejo de la memoria de forma indirecta (a través de índices, como se hace en los arreglos en lenguajes de alto nivel); así como invitar al estudiante a investigar con más detalle otras herramientas, servicios y técnicas para resolver adecuadamente el proyecto

Una explicación de la solución empleada desde el punto de vista algorítmico y de uso de la memoria (datos):

En un inicio se reunió como grupo para idear una manera adecuada de la realización del código, luego de compartir algoritmos posibles a utilizar se plantea una forma del desarrollo del código.

La matriz dinámica emplea una reserva llamada arreglo, la cual al principio se hizo de un tamaño de 34, ya que si el usuario digita 9 columnas se le deberían sumar dos por los extremos y luego multiplicar por filas, dando como resultado 33. Esta reserva aumento debido a la utilización de un número mayor de columnas, se valida que el dígito ingresado por la persona quien lo juegue sea impar, mayor a uno y menor que 97, por razones de que se le debe sumar 2; finalmente la reserva queda de 300 bytes por la multiplicación de las filas.

En caso de que el usuario ingrese un número mayor a dos dígitos se valida y se indica que es incorrecto, para esta validación se compara el dato que ingrese, si el primer elemento (posición 0) de la reserva es un 218 al introducir en pila, quiere decir que es de un solo dígito, si no lo es, se envía a la función en la que se verifica si los dos primeros dígitos son un número impar, al ser impar se procede a comparar el tercer elemento en la reserva (posición 2), si fuese distinto de 218 se indica que es un número mayor de 3 dígitos, por lo que se envía de nuevo a solicitar un número que cumplan las condiciones solicitadas. Si la respuesta brindada por el usuario es correcta se continúa el juego.

La manera en que se colocan los perros en el arreglo es mediante una función en la que tres nombres de perros son solicitados, cada nombre es guardado en una reserva de un tamaño de 10, por razón de que el nombre puede ser un poco extenso, luego se hace una serie de validaciones, las cuales son:

- Se valida si la inicial de cada nombre de perro está en el rango de 17 y 42, debido a que debe ser una letra en mayúscula entre la A–Z. Cualquier otro carácter que se ingrese ha sido validado con esta restricción, incluyendo números.

- Se compara si la primer letra de alguno de los perros coincide con otra, es decir son iguales, también se compara con la letra “L”, ya que corresponde a la inicial de la liebre.
- En caso de no cumplirse algún aspecto se solicita el nombre de los perros nuevamente.

Antes de la primer impresión de la matriz se hace el uso de tres subrutinas, la primer subrutina consiste en la llamada de una función encargada de recorrer el arreglo e ir colocando en cada posición un -6 (es equivalente a un * en ASCII), la condición de parada será cuando llegue a la posición: columna ingresada por el usuario, más dos, por tres.

La segunda subrutina se encarga de cambiar el contenido de las siguientes posiciones por un -16 en ASCII es igual a un espacio vacío:

- Arreglo+0
- Arreglo + (columnas digitadas+ 2)-1
- Arreglo + (columnas digitadas + 2)*2
- Arreglo + ((columnas digitadas +2)*3)-1

Las posiciones mencionadas corresponden a las esquinas de la matriz. Luego se hace uso de una tercer subrutina la cual coloca a la Liebre y a los perros en sus posiciones iniciales.

- Liebre: colocada en ((Arreglo + columnas digitadas+2) *2)-1
- Perro1: colocado en (Arreglo + 1)
- Perro2: colocado en (Arreglo + columnas digitadas)+2
- Perro3: colocado en ((Arreglo + columnas digitadas+2) *2)+1

La siguiente función tiene el objetivo de imprimir el número de columnas mediante las letras del abecedario, así el usuario se sentirá más cómodo con la apariencia del juego. Para esta función se emplea un contador inicializado en 0, el ciclo finaliza cuando el contador sea igual al número de columnas digitadas por el usuario más dos.

Se continúa con la impresión del contenido de cada posición del arreglo, dando como resultado la forma de una matriz, para ello el uso de dos contadores es necesario, se inicia con la impresión, se les debe sumar 48 para su correcta representación, luego el primer contador es comparado con el número de columnas más dos, si es igual saldrá del ciclo y realizara un salto de línea en donde es comparado el segundo contador, en caso de ser igual a la cantidad de elementos del arreglo que son distintos de 0 saldrá de la función, si no fuese de esta forma se devuelve al primer ciclo con el contador uno inicializado nuevamente en cero.

Al jugador se le pregunta que letra quiere mover en caso de ser el turno de los perros, si fuera el de la libre se le solicita que ingrese la letra L, luego se recorre el arreglo en busca de esa letra, cuando sea igual se guarda el contador utilizado para recorrer el arreglo y brinca a la función en que se muestran los controles. Para determinar cuál turno es, se utiliza un dd llamado indicativo el cual tendrá valores de 0 y 1, cuando la liebre realiza un movimiento

efectivo el indicativo cambiara a 1, así mismo con los perros solamente que el indicativo no cambiara a 1 sino a 0.

El siguiente paso es mostrar al jugador un mensaje en donde se indiquen los posibles movimientos, para los controles se utilizan números del 1 al 8, en donde el 1 es arriba, 2 abajo, 3 derecha, 4 izquierda, 5 diagonal derecha arriba, 6 diagonal derecha abajo, 7 diagonal derecha abajo y 8 diagonal izquierda abajo. Existe una opción 9, al ser presionado se guarda el juego y finaliza la partida, el juego termina porque se pensó que si un jugador quiere guardar el juego es por razones de que no tiene tiempo para seguir jugando o porque no cuenta con ánimos de finalizar la partida en ese momento.

Lo ingresado por el jugador es debidamente validado, solamente se podrá ingresar números del 1 al 9, si se digitara algo no mencionado se vuelve a solicitar el movimiento.

Cuando ha sido una opción válida se procede a una comparación en donde se envía a distintas funciones según corresponda el movimiento.

Antes de movilizar la letra se hace una verificación si la posición a la que se va mover contiene un -6 de esta forma nos damos cuenta si puede ser un movimiento correcto. También se evita sobrescribir una letra o colocar en una posición indebida, ocasionando luego el mal desarrollo del juego.

Se dirige a validar si la posición en que se encuentra es impar o no, para ello se utiliza el siguiente algoritmo:

- Se introduce un dos en la pila de flotantes.
- Se introduce el contador de la posición en la pila de flotantes.
- Se hace la división.
- El resultado se almacena en un dd como entero.
- Se saca el otro dígito de la pila, es almacenado en la reserva basura ya que no se utilizara.
- Se ingresa un dos a la pila de flotantes.
- Se ingresa el resultado que dio la división.
- Se realiza la multiplicación (fmul st1).
- El resultado es almacenado en un dd.
- Se vacía la pila.

Se compara el resultado con el contador de la posición en donde se encuentra la letra, si es igual quiere decir que es par, por lo tanto tendrá limitado los movimientos diagonalmente.

En el caso de que sea un perro tiene restringido los movimientos hacia la izquierda. Si un usuario digita que quiere movilizarse hacia "x" diagonal y se encuentra en una posición par, se devuelve a solicitar un movimiento valido.

Al realizar el movimiento se llama a las funciones encargadas de verificar los ganos posibles en el juego, la primer función recorre la matriz en busca de la liebre, al encontrarla compara la posición si es par o no, en caso de que si sea par verifica a la izquierda, derecha, arriba y

debajo de la posición en donde se ubica y compara con un -6 si son diferentes quiere decir que la liebre quedo acorralada.

Si fuera impar a parte de las direcciones mencionadas anteriormente debe verificar las diagonales, si coincidieran con un dígito distinto a -6, el cazador gana.

La segunda función se encarga de comparar un contador “y” con 10, el contador “y” aumenta cada vez que un perro se mueve hacia arriba o abajo, el contador se reinicia al movilizar “x” perro hacia alguna otra dirección, en caso de que llegara a 10, quiere decir que los perros no avanzaron por lo que la liebre gana.

La tercer función emplea un contador, se recorren todos los elementos del arreglo, cada vez que se recorran el número de elementos correspondiente el número de columnas, el contador es reiniciado, al encontrar alguna letra se guarda la posición según corresponda al perro1, perro2, perro3 o liebre.

Al recorrer el arreglo se procede a comparar la posición de la liebre con la posición de cada uno de los perros, si fuese menor el ganador seria la liebre.

Al finalizar la partida se brindan tres opciones:

- Revancha (Opción 1)
- Reiniciar juego (Opción 2)
- Salir (Opción 3)

Opcion1: Se llama a la subrutina encargada de colocar a los perros y liebre en sus posiciones iniciales al comenzar la partida, se reinicia el dd indicativo y el indicador de los turnos, luego se imprime las columnas y matriz.

Opcion2: Cada registro es limpiado, las variables utilizadas se inicializan en 0, se hace un jmp a la función _start, en la que se le pregunta al usuario si quiere jugar un nuevo juego o cargar partida.

Opcion3: se hace la llamada a la función salida, en donde el registro eax utiliza el servicio sysexit.

Se hace la validación en caso de que el usuario ingresa un número o carácter distinto a 1,2 o 3.

Al comenzar el juego se muestra una serie de aspectos llamativos, por ejemplo: las reglas, dibujo en ASCII, menú con dos opciones, cargar o juego nuevo.

Si cargar es seleccionado se procede a una función en la que se verifica la existencia del archivo, al no haber ninguna partida guardada se muestra un mensaje indicando carga incorrecta, luego se envía de nuevo a _start. Si un archivo estuviese creado la partida es cargada adecuadamente y continua donde fue guardada.

Para cargar y guardar fue necesario dos archivos llamados:

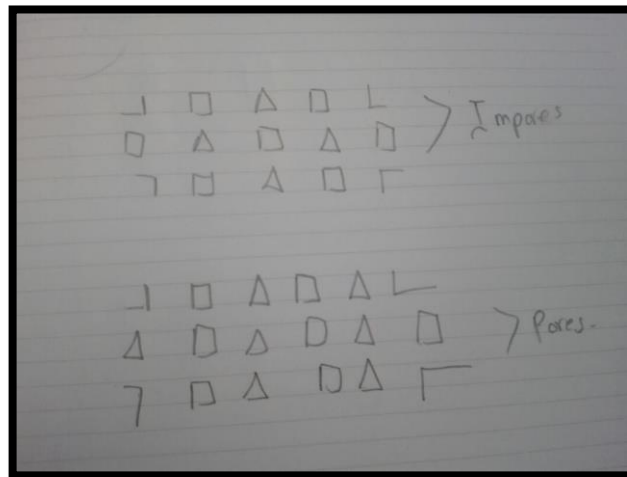
- Ranura.txt
- Ranura2.txt

En el primer txt se guarda el arreglo, en el segundo es guardado las variables necesarios por ejemplo: indicativo (contador de movimientos consecutivos del perro hacia arriba y abajo), indicador (muestra cual turno corresponde), reserva (contiene la cantidad de columnas), reserva2 (posee la cantidad total de elementos de la matriz) y nombres de los perros.

Se decidió elaborar dos archivos por razones de un mejor manejo de las variables a la hora de la lectura, también por razones de orden y un mejor entendimiento.

El número de columnas debe ser impar, ya que si fuese par las formas de ganar no serían las mismas, debido a que las filas quedarían con un orden igual tanto de cuadrados como de octágonos, en el cuadro se puede movilizar hacia la derecha, izquierda, arriba y abajo, en cambio en el octágono es posible cualquier movimiento a excepción del perro que tiene limitado el movimiento hacia la izquierda.

Se adjunta una fotografía tomada para una mejor comprensión:



Manual de usuario

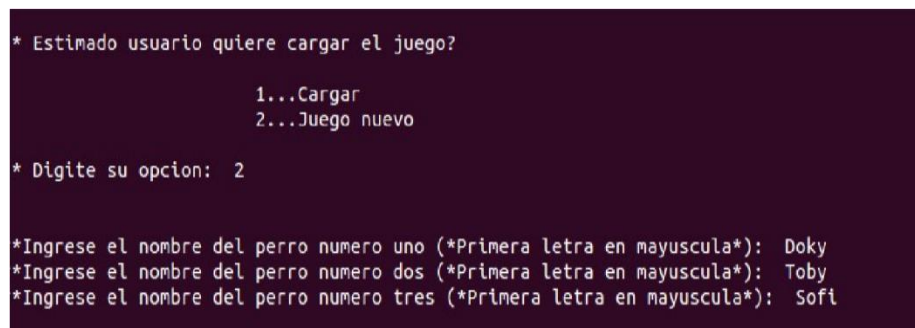
- 1) Para abrir el juego es necesario dirigirse a la carpeta en donde encuentra (terminal), es utilizado el comando “cd” , al llegar a la carpeta se hace make y luego “./tarea2”, al realizar esto se muestra una pantalla con dos opciones:

- Cargar juego
- Juego nuevo

Se debe elegir ya sea opción 1 o 2, al seleccionar 1 se dirige a cargar la partida que haya sido guardada, en caso de que no se encuentre ningún documento guardado se muestra un mensaje indicando la carga incorrecta, luego vuelve al inicio como se ve en la siguiente imagen. Al digitar algún número o carácter diferente a los mencionados se devuelve al inicio, ya que todo ello fue validado respectivamente.



- 2) La opción 2 inicia un nuevo juego, se solicita los nombres de los perros, es importante mencionar que sus iniciales no pueden ser iguales, además la primera letra debe ir en mayúscula como se muestra en la imagen. En caso de que digite algo incorrecto es validado.



- 3) Se digita la cantidad de columnas, esto queda a criterio del usuario, la condición debe ser que el número sea impar, menor que 99 y mayor que 1, a esta cantidad de columnas se le suma una al principio y otra al final.

```

**Ingrese el numero de columnas que necesita (*Impar*): 3

+ = Representacion Cuadrados-Octagonos

A   B   C   D   E
-----
T   D   *   *   L
    *   *   *
    S   *   *

```

- 4) Luego se imprime el tablero, en este paso se le solicita al usuario que digite la letra inicial de liebre o del nombre del perro, una vez que la letra sea presionada se muestra un menú cuyo contenido son los controles indicando mediante números del 1 al 8 todos los posibles movimientos, en caso de que el usuario presione la tecla 9 el juego será guardado.

```

+ = Representacion Cuadrados-Octagonos

A   B   C   D   E
-----
T   D   *   *   L
    *   *   *
    S   *   *

*Indique la inicial de la liebre para movillarla: L

-----CONTROLES-----

Moverse hacia arriba.....Digite *** 1
Moverse hacia abajo.....Digite *** 2
Moverse hacia la derecha.....Digite *** 3
Moverse hacia la izquierda.....Digite *** 4
Moverse diagonal derecha arriba.....Digite *** 5
Moverse diagonal izquierda arriba.....Digite *** 6
Moverse diagonal derecha abajo.....Digite *** 7
Moverse diagonal izquierda abajo.....Digite *** 8
Guardar Juego.....Digite *** 9

*Indique el movimiento que quiere hacer mediante el numero correcto:

```

- 5) Dependiendo de la posición en que se encuentre la liebre o el perro será válido el movimiento, si la posición hace referencia a un número par se puede dirigir hacia la izquierda, derecha, arriba o abajo (en caso de que sea un perro no es posible el movimiento a la izquierda), si la posición indica un número impar se moverá en cualquier dirección siempre y cuando el lugar del arreglo en donde se vaya a colocar contenga un -6 (asterisco en ASCII) , como se mencionó anteriormente el perro no cuenta con movimiento a la izquierda esto incluye la diagonal. Con un mensaje se le hace saber al usuario si el movimiento fue efectivo o no. En la imagen es mostrada el turno de la liebre.

```

*Indique el movimiento que quiere hacer mediante el numero correcto: 6

*****
El movimiento que ha efectuado se realizo satisfactoriamente
*****

+ = Representacion Cuadrados-Octagonos

A   B   C   D   E
-----
T   D   *   L   *
    *   *   *
    S   *   *

*Indique la inicial del perro que quiere mover:

```


- 6) El turno de los perros es efectuado de la misma manera que el de la liebre, si se ve internamente la función es la misma por razones de eficiencia, a diferencia que los perros no se mueven hacia atrás. A continuación se puede visualizar en la imagen.

```
**Indique la inicial del perro que quiere mover: D
```

-----CONTROLES!-----

```
+ Moverse hacia arriba.....Digite *** 1
+ Moverse hacia abajo.....Digite *** 2
+ Moverse hacia la derecha.....Digite *** 3
+ Moverse hacia la izquierda.....Digite *** 4
+ Moverse diagonal derecha arriba.....Digite *** 5
+ Moverse diagonal izquierda arriba....Digite *** 6
+ Moverse diagonal derecha abajo.....Digite *** 7
+ Moverse diagonal izquierda abajo.....Digite *** 8
+ Guardar juego.....Digite *** 9
```

```
**Indique el noviniendo que quiere hacer mediante el numero correcto: 3
```

```
*****
el noviniendo que ha efectuado se realizo satisfactoriamente
*****
```

*

 - Representacion Cuadrados-Octagonos

| A | B | C | D | E |
|---|---|---|---|---|
| x | T | D | T | |
| | S | * | L | * |

- 7) Cuando el 9 es digitado el archivo se guarda y sale del programa, se pensó de esta manera ya que si un usuario guarda el juego es por razones de que no quiere jugar más o no tiene tiempo, en la imagen se puede ver el mensaje mostrado al llevar a cabo esta acción.

```

-----CONTROLES!!-----
+ Moverse hacia arriba.....Digite *** 1
+ Moverse hacia abajo.....Digite *** 2
+ Moverse hacia la derecha.....Digite *** 3
+ Moverse hacia la izquierda.....Digite *** 4
+ Moverse diagonal derecha arriba.....Digite *** 5
+ Moverse diagonal izquierda arriba.....Digite *** 6
+ Moverse diagonal derecha abajo.....Digite *** 7
+ Moverse diagonal izquierda abajo.....Digite *** 8
+ Guardar juego.....Digite *** 9

**Indique el movimlento que quiere hacer mediante el numero correcto: 9

--- Archivo guardado ---

```

- 8) Al inicio si se elige la opción cargar, lee lo que contiene los archivos ranura.txt y ranura2.txt, en el primero se lee el arreglo y la posición de las letras según la partida guardada y en ranura2 se lee variables como lo es el contador de movimiento de los perros (arriba, abajo), turno en el que quedo, nombre de los perros y cantidad de columnas.

[illegible]

- 9) Primera forma de ganar, al encontrarse la liebre al lado izquierdo de todos los perros, en el caso de la imagen el usuario mueve el perro a la derecha y queda la liebre a la izquierda. Por lo tanto gana.

```
A B C D E
* L D * *
* T * * *
* S * * *

**Indique la inicial del perro que quiere mover: T

-----CONTROLES!-----

Moverse hacia arriba.....Digite *** 1
Moverse hacia abajo.....Digite *** 2
Moverse hacia la derecha.....Digite *** 3
Moverse hacia la izquierda.....Digite *** 4
Moverse diagonal derecha arriba.....Digite *** 5
Moverse diagonal izquierda arriba.....Digite *** 6
Moverse diagonal derecha abajo.....Digite *** 7
Moverse diagonal izquierda abajo.....Digite *** 8
Guardar juego.....Digite *** 9

**Indique el movimiento que quiere hacer mediante el numero correcto: 3

*****
El movimiento que ha efectuado se realizo satisfactoriamente
*****

FIN DEL JUEGO !!
**** GANADOR: LIEBRE ****
```

- 10) Otra opción en el que la liebre gana es si los perros se muevan arriba y abajo 10 veces, esto indica que no se avanza en el juego por lo cual gana la liebre.

- 11) La tercera opción de ganar es en caso de que los perros acorralen a la liebre, esto significa que no tendrá movimiento alguno. En la imagen el perro con la inicial S se mueve hacia arriba, por lo que la liebre queda sin movimientos posibles.

```
* - Representación Cuadrados-Octagonos
A B C D E
* T L * *
* * D S *
* * * * *

**Indique la inicial del perro que quiere mover: S

-----CONTROLES!-----

Moverse hacia arriba.....Digite *** 1
Moverse hacia abajo.....Digite *** 2
Moverse hacia la derecha.....Digite *** 3
Moverse hacia la izquierda.....Digite *** 4
Moverse diagonal derecha arriba.....Digite *** 5
Moverse diagonal izquierda arriba.....Digite *** 6
Moverse diagonal derecha abajo.....Digite *** 7
Moverse diagonal izquierda abajo.....Digite *** 8
Guardar juego.....Digite *** 9

**Indique el movimiento que quiere hacer mediante el numero correcto: 1

*****
El movimiento que ha efectuado se realizo satisfactoriamente
*****

FIN DEL JUEGO !!
**** GANADOR: CAZADOR ****
```

- 12) Opciones después de terminar la partida:

- Reiniciar el juego con los mismos nombres de perros y mismo tamaño de tablero.
- Reiniciar el juego desde cero, solicitando de nuevo nombres y tamaño del tablero.
- Salir del juego.

```
FIN DEL JUEGO !!
**** GANADOR: CAZADOR ****

Opciones, digite segun lo indicado:

1.....Jugar con los mismos jugadores
2.....Reiniciar juego
3.....Salir

** Su opcion es ....
```

Referencias:

Juego de la liebre y los perros cazadores. (2016, 1 de junio). Wikipedia, La enciclopedia libre. Fecha de consulta: 09:01, junio 8, 2016 desde https://es.wikipedia.org/w/index.php?title=Juego_de_la_liebre_y_los_perros_cazadores&oldid=91434924.