

Fecha: 1 Junio, 2016

Hora: 3:42 pm

→ Se reúne en grupo para idear una manera correcta de llevar a cabo la tarea. Se comentan distintos algoritmos, luego se plantea una forma para la realización del proyecto.

→ Se utiliza una reserva de 34 ya que se pensó que el número de columnas sería 9, entonces se realizaría el siguiente proceso para obtener la cantidad total de elementos de la matriz:

1) Al número de columnas ingresado por el usuario se le suman dos por los extremos (posiciones iniciales en donde se ubica la liebre y el perro), luego se multiplica por la cantidad de filas en este caso es 3.

→ Si se ingresa como número máximo "9" al sumar 2 y multiplicar por 3 da como 33.

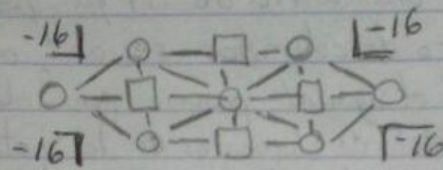
→ Por eso se hace de 34 la reserva.

→ Se continúa con las validaciones, cada uno en su casa, se lo va validando por ejemplo si el número es impar, ya que

con un número por de columnas los temas de ganar en el juego cambian, esto es porque en un cuadro es posible moverse solamente izquierdo, derecha, arriba y abajo, en cambio en un octágono cualquier movimiento se puede hacer.

○ = octágono

□ = cuadro



- Se tiene la validación por si es par o < que 1 ya que no tendría sentido jugar solo con una columna
- Se procede a la impresión de la matriz. Se tienen 2 contadores, el primer contador en cada impresión de la posición del arreglo se le suma 1, al segundo contador también, luego se compara el contador 1 con el número de columnas al ser igual saldrá del ciclo y hora

Bitacora
Josue Rodríguez Alfaro
Luis Alejandro Salas

un cambio de línea, luego se compara el
contador 2 con la cantidad total de elementos.
Si coincide sale del ciclo 2 indicando que
ya finalizó la impresión de la matriz, si
no fuese igual el contador 1 se reinicia y
se hace un "jmp" al ciclo 1.

Fecha: 2 Junio
Hora: 5:30pm

reserva = "28"
reserva[0] = 2

→ Se valida si el usuario ingresa un número superior a 1 dígito

→ Al número recibido se le resta lo ⁴⁸ y el segundo dígito de la reserva es comparado con 28 y a que si fuera igual quiere decir que es solo un dígito, en caso de que no se procede a verificar si los dos primeros dígitos son un número impar, en caso de no solo se vuelve a solicitar el número de columnas.

Si fuese impar se compara el tercer elemento de la reserva (reserva[2] con un 26)

→ Si es igual se imprime la matriz

→ Si no lo es se solicita el número de columnas nuevamente porque nos indica que tiene más de 2 dígitos.

Y en este caso se trabajará como número máximo de columnas 97, debido a que se necesita una pantalla muy grande para observar toda la matriz al ser un número superior.

→ Se utilizan ³ subrutinas:

- 1) La primera subrutina se encarga de colocar en el arreglo por cada posición un -6 que en representación ASCII corresponde a un *, el ciclo termina cuando llega al número de columnas ingresado, más 2 y por 3.
- 2) Segunda subrutina coloca en los esquinas de la matriz un -16 \Rightarrow espacio vacío

Posición es:

- 1) $\text{Arreglo}[0]$
- 2) $\text{Arreglo}[(\text{columnas digitadas} + 2) * 3 - 1]$
- 3) $\text{Arreglo}[(\text{columnas digitadas}) - 1]$
- 4) $\text{Arreglo}[(\text{columnas digitadas} + 2) * 2]$

3) Tercera subrutina Coloca en las posiciones:

$\text{Arreglo}[1]$, $\text{Arreglo}[(\text{columnas digitadas} + 2)]$, $\text{Arreglo}[(\text{columnas digitadas} + 2) * 2 - 1]$, las letras iniciales de los perros y liebre según correspondo.

→ Los nombres de perros son solicitados al inicio se valida que la inicial comience con mayúscula para ello debe estar entre el rango de (17-42), si alguna letra coincide con otro se vuelve

a solicitar el nombre de los Perros.

Fecha 3 Junio:

Hora: 12:17 pm

- Al tener la impresión de la matriz y las iniciales de los Perros y Liebre, se procede a los movimientos.
- Al usuario se le muestra un menú en donde se indica los movimientos
- 1 → Arriba
- 2 → Abajo
- 3 → Izquierdo
- 4 → Derecha
- 5 → Diagonal Arriba Derecha
- 6 → Diagonal Arriba Izquierdo
- 7 → Diagonal Abajo Derecha
- 8 → Diagonal Abajo Izquierdo
- Primero según un indicador que se define al principio como dd., será el turno de la liebre si corresponde a un 0, si es un 1 es el turno del cazador.

- Al jugador se le solicita que ingrese la inicial del alguno de los perros o liebre que quiere mover según correspondo. el turno
- Se tiene una función que recorre la matriz en busca de la letra al encontrarla la posición es guardada, luego se procede con el dígito ingresado por el usuario para realizar el movimiento.
- Según el movimiento se compara si a la posición en que se va mover existe un -6, esto quiere decir que hay un * por lo tanto el movimiento puede ser válido
- La posición en que se encuentra la letra se compara si es par o no

Algoritmo:

- Ingresar un 2 a la pila de flotantes
- Ingresar la posición de la letra
- hacer división y guardar como entero
- Vaciar pila
- Ingresar 2 a pila flotantes
- Ingresar resultado de la división
- hacer multiplicación
- guardar resultado como entero
- vaciar pila

→ El resultado es comparado con la posición de la letra si es par significa que el resultado es igual, si no lo fuera es impar

→ Al ser par se deberá validar que no sea un movimiento diagonal

→ Si fuera impar se realiza el movimiento

Fecha 4 Junio

Hora: 9:10 am

→ Se procede con los movimientos:

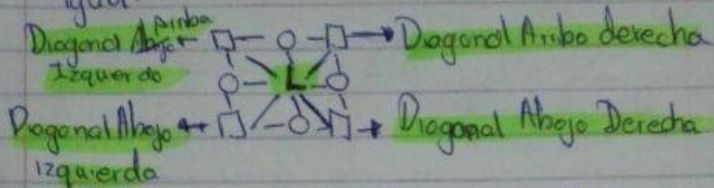
Arriba: Indica que a la posición en que se encuentre ya sea el perro o lebre se le reste el número de columnas +2.

Abajo: A la posición de la letra se le suma el número de columnas +2.

Izquierdo: A la posición de la letra se le resta 1.

Derecha: A la posición de la letra se le suma

→ Los Diagonales llevan el proceso similar a diferencia que arriba se le suma 1 o resta 1 según sea derecha o izquierda y abajo es igual.



- Se documenta el código que se llevo hasta hoy.
- Se termina de trabajar a los 2:30pm.

Fecha 4 Junio:
Hora: 9:30pm.

- Se inicio con las validaciones de gonor.
- Se trabajo en la forma de gonor en que la fiebre se encuentre a la izquierda de todos los perros:

Algoritmo usado:

- Realizar un ciclo en el que se usen 2 contadores.
- Recorrer matriz si se encuentre alguna letra la posición es guardada según corresponda al

$$\begin{aligned} L &= 2 \\ P_1 &= 3 \quad L < P = \checkmark \\ P_2 &= 4 \\ P_3 &= 5 \end{aligned}$$

perro o liebre

- El primer contador se reinicia al haber recorrido cada fila, ese contador es el que indicara la posición de la letra.
- El ciclo ~~de~~ acaba cuando el contador 2 es igual a la cantidad total de elementos de la matriz.
- Una vez finalizado el ciclo se compara si la posición de la liebre es menor que la de los perros si resultara de esta forma quiere decir que la liebre gana si no fuera así se procede con la validación de la segunda forma de ganar.

→ En esta se tuvo problema ya que el contador que indica el número de movimiento de los perros no está reiniciando.

Fecha: 6 Junio

Hora: 7:30 am

- Se ingresa el contador a la pila para ver donde es que no esta aumento al realizar algun perro ~~at~~ un movimiento hacia arriba o abajo
- El problema es solucionado, ya que no se estaba ingresando donde aumentaba el contador
- Se sigue con la verificación, cada vez que un perro se mueva hacia abajo o arriba el contador aumento, la única forma que reinicie es si los perros se mueven hacia la derecha o diagonal
- Si llegara el contador a 10 la liebre gana
- Tercer forma de ganar: Recorrer la matriz en busca de la letra inicial "L" al encontrarlo se compara la posición en donde se encuentra con par o impar. Si es par se verifica arriba, abajo, izquierda y derecha de la posición de la liebre con un número distinto a -6, si fuese así quiere decir que la liebre fue acorralado por lo tanto gana el coraza. Si es impar se debe verificar los lados mencionados anteriormente y tambien

las diagonales en caso de que todas coincidiera con un dígito diferente a -6, el cazador gana la partida

→ Si alguna posición ~~no~~ fuera igual a un -6 el juego continúa normalmente ya que ninguno ha ganado

* → Se hace la función de reiniciar el juego.

→ Se vacían los registros utilizados

→ el indicativo es inicializado en 0

→ el indicador se coloca en 0

* indicativo = contiene los movimientos de los perros hacia arriba o abajo

* indicador = indica el turno de la liebre o el perro según correspondiendo a 0 o 1

→ se hace un jmp a la etiqueta -start

* Se hace función revancha:

→ vaciar registros

→ inicializar indicativo e indicador en 0

→ Realizar jmp a la etiqueta imprimir Columnas

luego imprime la matriz con el tamaño est obtenido
y los nombre de perros iguales al turn anterior

Fecha 6 Junio:

Hora 9:27am

→ Se le da una mejor apariencia al menú del juego

→ Se imprimen los reglos, se valida cuando finaliza el juego que solo se pueda ingresar numero del 1 al 3 ya que solo se tienen 3 opciones

1 → Revoncho

2 → Reiniciar

3 → Salir.

→ Es implementado lo de guardar, se utiliz 2 archivos, en el primer archivo se guarda el arreglo y en el segundo es guardado variables necesarios como lo son:

1) Nombres de Perros

2) Cantidad Columnas

3) Tamaño total del arreglo

4) Indicativo

5) Indicador.

- Se debe elaborar 2 archivos ya que permite un mejor manejo de las variables y una mejor comprensión.
- La opción de guardar se muestra en el menú de controles, si el usuario digita q el juego es guardado.
- Luego finaliza la partida indicando mediante un mensaje:
Archivo Guardado

→ Fecha: 7 Junio
Hora: 8:00am.

- Finalizar la documentación del código.
- Realizar el readme.
- Iniciar con la documentación externa.

