

**UNIVERSIDAD DE BUENOS AIRES**  
**Facultad de Ingeniería**  
**Departamento de Computación**

**Algoritmos y Programación I (75.40)**

**Trabajo Práctico N° 1**

**Curso: 005 - Día: Lunes - Horario: 15 – 21 hs.**

**Primer Entrega**

**Profesor: M. Ing. Fernando J. Lage**

**ATP:**

**Lic. Esteban Calabria**

**Ing. Guido Costa**

**Ing. Fernando Salgueiro**

Apellido y nombre	Padrón	E-mail
Rinaldi, Lautaro Ezequiel		

**Calificación:**

**Firma:**

**Observaciones:**



## **ALGORITMOS Y PROGRAMACIÓN**

### **PRIMER TRABAJO PRÁCTICO**

Prof. Mg. Ing. F. J. LAGE

Alumno: Rinaldi, Lautaro Ezequiel.

Nº Padrón:

A.T.P.

Lic. Esteban Calabria

Ing. Guido Costa

Ing. Fernando Salgueiro

Primer Entrega

## **Índice.**

<b><u>ÍNDICE.</u></b>	<b><u>2</u></b>
<b><u>ENUNCIADO.</u></b>	<b><u>3</u></b>
<b><u>CONSIDERACIONES E HIPÓTESIS.</u></b>	<b><u>4</u></b>
<b><u>CRONOGRAMA DE TRABAJO DETALLADO.</u></b>	<b><u>5</u></b>
<b><u>PROBLEMÁTICA Y RESOLUCIÓN CORRESPONDIENTE A CADA MÓDULO.</u></b>	<b><u>6</u></b>
<b><u>MANUAL DEL USUARIO.</u></b>	<b><u>10</u></b>



## **ALGORITMOS Y PROGRAMACIÓN**

### **PRIMER TRABAJO PRÁCTICO**

Prof. Mg. Ing. F. J. LAGE

Alumno: Rinaldi, Lautaro Ezequiel.

Nº Padrón:

A.T.P.

Lic. Esteban Calabria

Ing. Guido Costa

Ing. Fernando Salgueiro

Primer Entrega

## **Enunciado.**



El Tetris fue inventado en 1985 por Alexey Pajitnov, cuando se encontraba trabajando en la Academia de Ciencias de Moscú. El juego ha vendido más de 50 millones de copias en el mundo y se transformó en uno de los más famosos.

### **Mecánica de juego.**

Siete tipos de piezas bidimensionales de 4 bloques en distintas posiciones, llamadas Tetriminos, cayendo desde la parte superior de la pantalla.



El jugador las debe rotar ( $0^\circ$ ,  $90^\circ$ ,  $180^\circ$ ,  $270^\circ$ ) y/o desplazar por una superficie definida de juego llamada tablero y decidir como colocarlas de forma que se formen líneas completas. Cuando una línea horizontal se completa, esa línea desaparece y todas las piezas que están por encima descienden una posición, liberando espacio de juego y por tanto facilitando la tarea de situar nuevas piezas.

Se pide implementar el reconocido juego de computadora “Tetris” utilizando el compilador freepascal utilizando el modo texto de 80x25 caracteres.

En cualquier momento de la partida el jugador puede presionar la tecla ESC y el juego le mostrará un mensaje dándole la oportunidad de abandonar el juego o continuarlo.

El juego consta de distintos niveles, en donde la velocidad de caída de las piezas se va acelerando agregándole dificultad al juego. La condición para incrementar el “nivel” es que el jugador complete 5 líneas.



## **ALGORITMOS Y PROGRAMACIÓN**

### **PRIMER TRABAJO PRÁCTICO**

Prof. Mg. Ing. F. J. LAGE

Alumno: Rinaldi, Lautaro Ezequiel.

Nº Padrón:

A.T.P.

Lic. Esteban Calabria

Ing. Guido Costa

Ing. Fernando Salgueiro

Primer Entrega

## **Consideraciones e Hipótesis.**

- Se adoptó el tamaño del tablero de juego de 10 columnas por 20 filas, que es el tamaño universal.
- Se decidió que las fichas que bajan son aleatorias entre los 7 tipos, pero nunca bajan rotadas, sino que lo hacen en su posición original.
- Se decidió mostrar la ficha actual, ya que al bajar incrementalmente, el jugador desconoce de cual se trata hasta que ya haya bajado varias veces, perdiendo tiempo sin poder decidir que hacer con ella.
- Se consideró como rotación al acto de girar la ficha y comprobar que no quede superpuesta con ninguna otra, o exceda los límites del tablero; Si fuera así, no puede rotar.
- Se decidió que las fichas rotan en un sólo sentido (contrario al de las agujas del reloj), para no confundir al jugador haciéndola rotar en ambos.
- Avanzar de nivel significa que el tiempo que tarda la ficha en descender una posición disminuye. En este caso, por nivel disminuye 75 milisegundos, siendo el tiempo en nivel 1 de 375 milisegundos, y en nivel 5 de 75 milisegundos.
- Se decidió que el juego tenga 5 niveles, que como aclara el enunciado, se necesita completar un mínimo de 5 líneas para avanzar (pueden ser más, por ejemplo, si se completaron 4 líneas en un determinado nivel, y la próxima vez, se completan dos líneas simultáneamente, la sexta inclusive, se considera del nivel actual, ya que la completó sin que se haya acelerado el tiempo todavía).
- El puntaje obtenido por borrar una línea se calcula como 10 veces el nivel. Así, borrar una línea en nivel 1 son 10 puntos, y una en nivel 4 son 40.
- Se considera que el jugador pierde, cuando hay fichas en el tablero que impiden bajar a la siguiente, o cuando parte de la ficha actual ya no pueda bajar más (porque se superpondría con otra), y quede excediendo el tablero hacia arriba.



## **ALGORITMOS Y PROGRAMACIÓN**

### **PRIMER TRABAJO PRÁCTICO**

Prof. Mg. Ing. F. J. LAGE

Alumno: Rinaldi, Lautaro Ezequiel.

Nº Padrón:

A.T.P.

Lic. Esteban Calabria

Ing. Guido Costa

Ing. Fernando Salgueiro

Primer Entrega

## **Cronograma de Trabajo detallado.**

<b>Fecha de Inicio.</b>	<b>Fecha de Finalización.</b>	<b>Tarea realizada.</b>
21/04/2008	21/04/2008	Entrega y explicación del enunciado.
25/04/2008	25/04/2008	Análisis general del sistema y su fraccionamiento en módulos principales.
26/04/2008	27/04/2008	Creación del Tablero. Generación aleatoria de fichas. Evitar superposición de fichas entre sí. Perder.
29/04/2008	29/04/2008	Investigación de lectura de teclas de función del teclado. Desplazamiento horizontal de las fichas y no superposición (horizontal) entre ellas.
03/05/2008	04/05/2008	Fijado de número de columnas y filas en pantalla. Instrucciones en caso de tecla ESC. Mostrar Ficha Actual. Prueba de colores.
05/05/2008	08/05/2008	Mostrar Próxima ficha. Ajuste de colores. Optimización de código. Rotación.
10/05/2008	11/05/2008	Ajuste de Perder. Borrar líneas completas. Nivel y Puntaje. Ganar. Instrucciones. Menú.



## **ALGORITMOS Y PROGRAMACIÓN**

### **PRIMER TRABAJO PRÁCTICO**

Prof. Mg. Ing. F. J. LAGE

Alumno: Rinaldi, Lautaro Ezequiel.

Nº Padrón:

A.T.P.

Lic. Esteban Calabria

Ing. Guido Costa

Ing. Fernando Salgueiro

Primer Entrega

## **Problemática y Resolución correspondiente a cada Módulo.**

### **LEERTECLA.**

Como no se pudo hacer que READKEY lea las teclas de dirección, entonces se buscó dentro del manual de la UNIT KEYBOARD que función o procedimiento si podía hacerlo. Luego de mucha investigación y prueba se logró desarrollar esta función capaz de detectar cualquier tecla que se presiona en el teclado. El método es el siguiente:

- La función POLLKEYEVENT lee la próxima tecla que se presiona, y la almacena en la lista de eventos pendientes. Posee la ventaja que no espera a que se presione una (como el READKEY). Devuelve 0 si no se ha presionado ninguna tecla, y otro número entero si se ha presionado alguna (varía según la tecla presionada).
- La función GETKEYEVENT quita la tecla leída de la lista de eventos pendientes. Habilita para leer la siguiente tecla del teclado. Sin esta instrucción es como si la tecla quedase presionada.
- La función TRANSLATEKEYEVENT convierte el código único que tiene la tecla en un código ASCII de eventos.
- La función GETKEYEVENTFLAGS nos dice que tipo de tecla se ha presionado (si es de tipo ASCII, de función, o de otros tres tipos que a nosotros no nos interesan).
- En caso de ser una tecla con ASCII, la función GETKEYEVENTCHAR nos devuelve el carácter que le corresponde al ASCII de la tecla presionada; En caso de ser una tecla de función, transformamos el código ASCII que teníamos con la función GETKEYEVENTCODE en un código que la función FUNCTIONKEYNAME nos transformará en el nombre de la tecla de función presionada (en inglés).

### **GUARDARFICHASEN.**

En este procedimiento, el valor de cada celda de la matriz ficha, se guarda en un vector como string. Lo ideal sería que estén definidas como constantes, pero, al no encontrar la manera de utilizar constantes que puedan diferenciarse por un índice como a los vectores, se optó por guardarlos así.

### **PROXIMAFICHA.**

En este procedimiento, se utiliza el vector generado por GUARDARFICHASEN para cargar todas las celdas de la matriz de la ficha (Convierto el vector de string en una posición aleatoria, en una matriz).



## **ALGORITMOS Y PROGRAMACIÓN**

### **PRIMER TRABAJO PRÁCTICO**

Prof. Mg. Ing. F. J. LAGE

Alumno: Rinaldi, Lautaro Ezequiel.

Nº Padrón:

A.T.P.

Lic. Esteban Calabria

Ing. Guido Costa

Ing. Fernando Salgueiro

Primer Entrega

#### **CREARTABLERO.**

A través de este procedimiento, se muestran en pantalla los bordes del tablero en juego. Se entiende que esta forma no es la más eficiente, y es bastante rebuscada, pero sin embargo es efectiva. Cambiarla a una más sencilla como escribir la primera y la última línea con un WRITE cada una, y utilizar un FOR para los extremos laterales únicamente, sería copiar la idea de otro compañero, debido a este motivo no fue reemplazado.

#### **DONDEESTALAFICHA.**

Este procedimiento determina cuáles son la primera fila y la última columna ocupadas dentro de la matriz ficha. Esta información posteriormente se utilizará para comprobar, por ejemplo, que no se superpongan las fichas.

En un principio, también se buscaban la primera columna y la última fila, pero debido a la forma en que se decidió hacer que roten las fichas, se consideró que era mejor que esos datos permanezcan como constantes.

#### **MOSTRARFICHA.**

Este procedimiento, dadas determinadas posiciones dentro del tablero, se encarga de mostrar o de borrar la ficha en pantalla (tanto dentro del tablero, como en ficha actual o en próxima ficha).

#### **VACIARTABLERO.**

Este procedimiento inicializa en FALSE todas las posiciones del tablero lógico, para que durante las comparaciones no se produzcan errores.

#### **CARGARTABLERO.**

En el caso en que una ficha no pueda seguir bajando, la ficha debe ser grabada en la matriz que contiene al tablero. Este procedimiento se encarga justamente de guardar las posiciones de la ficha que quedan dentro del tablero.

#### **SESUPERPONEV.**

Esta función se encarga de verificar si la ficha puede bajar nuevamente o no: Controla que no se encuentre en la última fila del tablero, y que no haya celdas ocupadas donde ella tendría que estar si volviese a bajar.

En caso de poder bajar (ya controló toda la matriz de la ficha), devuelve FALSE; En caso contrario, cuando detecte la primer celda que haga que se superpongan devuelve TRUE (o si anteriormente se encontró con que estaba en la última fila del tablero), indicándole al programa que deje la ficha actual donde está, y haga bajar una nueva.

El principal inconveniente que hubo en este módulo fue que en un principio, sólo se comparaba la última fila de la ficha con la siguiente del tablero, sin tener en cuenta que hay fichas que tienen celdas ocupadas en las filas anteriores, pero en la última no. Por eso se hace verificar todas las filas ocupadas de la matriz ficha.



## **ALGORITMOS Y PROGRAMACIÓN**

### **PRIMER TRABAJO PRÁCTICO**

Prof. Mg. Ing. F. J. LAGE

Alumno: Rinaldi, Lautaro Ezequiel.

Nº Padrón:

A.T.P.

Lic. Esteban Calabria

Ing. Guido Costa

Ing. Fernando Salgueiro

Primer Entrega

#### **SESUPERPONEH.**

Esta función se encarga de controlar si al mover una ficha, hacia la derecha o hacia la izquierda, va a superponer con alguna otra. En el caso en que el tablero no contenga ninguna ficha en la posición en la que quedaría la actual al moverla, devuelve FALSE, permitiendo el movimiento; En caso contrario, devuelve TRUE, evitando que el programa la mueva.

No controla si la ficha se encuentra apoyada contra un límite lateral del tablero.

#### **TECLAVALIDA.**

Esta función nos devuelve el primer carácter (en mayúscula) de la tecla presionada, si es una que nos interese usar (está pensada para que no haya problemas con las minúsculas, o en el caso de las teclas de función, LEERTECLA nos devuelve un STRING, y a nosotros nos interesa tan sólo el primer carácter de la misma, para convertirla en una variable ordinal y poder usarla en un CASE...OF).

#### **GIRAR3X3.**

Este procedimiento se encarga de hacer rotar la ficha en juego dentro de la matriz lógica, (previamente verifica si se cumplen las condiciones para que ello ocurra).

En caso de ser el cuadrado, no realiza ninguna operación. Está pensado exclusivamente para las fichas que pueden encajarse y rotarse dentro de una matriz de 3 columnas y 3 filas, y se agrega como excepción, la rotación de la línea (que requiere de una matriz de 4 columnas y 4 filas).

#### **TECLADO.**

Este procedimiento, si se ha presionado una tecla válida, selecciona qué sentencias tiene que ejecutar el programa de acuerdo a la tecla presionada. Encierra parte del código que necesitan distintos procedimientos para funcionar, pero que no se han puesto en ellos (como por ejemplo, que la ficha al desplazarse horizontalmente no exceda los límites del tablero). Actúa como un menú implícito, que no tiene visualización en pantalla.

Contiene las llamadas a los procedimientos controlar si se ha presionado una tecla, si dicha tecla es de nuestro interés, de rotar la ficha, de verificar la superposición al desplazarla horizontalmente, y también es la responsable del tiempo que tarda en bajar una posición la ficha. Lee, como máximo, 15 teclas por unidad de tiempo que tarda en bajar la ficha.

El mayor inconveniente que presentó su desarrollo, fue decidir que tiempos eran los más apropiados para cada nivel.





## **ALGORITMOS Y PROGRAMACIÓN**

### **PRIMER TRABAJO PRÁCTICO**

Prof. Mg. Ing. F. J. LAGE

Alumno: Rinaldi, Lautaro Ezequiel.

Nº Padrón:

A.T.P.

Lic. Esteban Calabria

Ing. Guido Costa

Ing. Fernando Salgueiro

Primer Entrega

#### **PERDISTE.**

Esta función se encarga de verificar, una vez que la ficha actual ya no pueda seguir bajando, y haya sido guardada en el tablero lógico, si el jugador pierde o no.

#### **AJUSTOPANTALLA.**

Este procedimiento se encarga de fijar la pantalla en 80 columnas por 25 filas, como aclara el enunciado del problema a resolver.

#### **FILASLLENAS.**

Esta función verifica si se completó alguna línea horizontal. En caso de ser así, la borra del tablero lógico, y baja las restantes una posición. También cuenta que cantidad de líneas se borraron (para que otro procedimiento verifique el nivel y el puntaje).

#### **RECARGATABLERO.**

Este procedimiento, en caso de haberse borrado alguna línea, actualiza el tablero en pantalla para que coincida con el tablero lógico.

#### **PUNTOSYNIVEL.**

Este procedimiento, actualiza el puntaje obtenido, y lleva el control de la cantidad de líneas borradas en el nivel actual. Se encarga de, en caso de corresponder, incrementar el nivel, y controlar si el jugador ganó (es decir, superó los 5 niveles).

#### **PRINCIPAL.**

Este procedimiento, es el programa JUGAR en sí mismo, hace todas las llamadas a funciones y procedimientos necesarios en cada caso, como demás instrucciones necesarias para que funcionen todos ellos. Todas las variables pasadas por referencia a los distintos módulos, corresponden a este.

#### **ENQUECONSIESTE.**

Muestra en pantalla cuál es el objetivo del juego, y qué le permite hacer éste. Básicamente, coincide con el enunciado del problema a resolver.

#### **COMOJUGAR.**

Muestra en pantalla las teclas que le corresponden a cada acción dentro del juego.

#### **MENU.**

Es el menú principal del juego, que contiene a las opciones básicas (Jugar, ¿En qué consiste el juego?, ¿Cómo jugar?, Salir).



## ALGORITMOS Y PROGRAMACIÓN

### PRIMER TRABAJO PRÁCTICO

Prof. Mg. Ing. F. J. LAGE

Alumno: Rinaldi, Lautaro Ezequiel.

Nº Padrón:

A.T.P.

Lic. Esteban Calabria

Ing. Guido Costa

Ing. Fernando Salgueiro

Primer Entrega

## Manual del usuario.

Una vez que haya ejecutado el archivo TETRIS.EXE se desplegará en pantalla el Menú Principal del Programa (Figura 1).

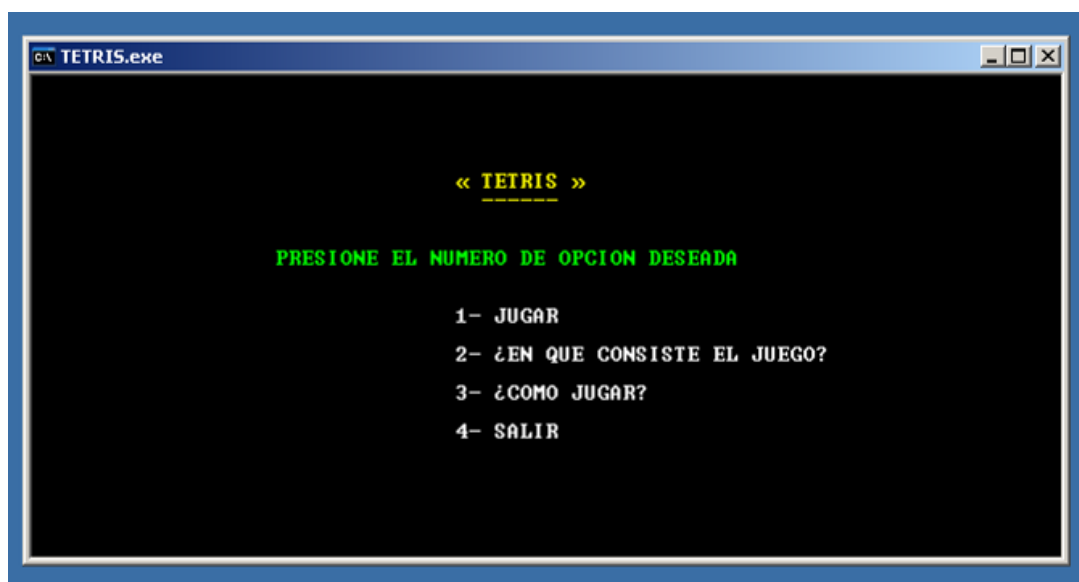


Figura 1: Menú Principal.

**NOTA:** No lo lleve a pantalla completa, utilícelo maximizado.

Presionando los números del 1 al 4 podrá ingresar en las distintas opciones que figuran en pantalla.

### **Opción 1:**

Ingresando en esta opción, el juego comienza inmediatamente. Las fichas irán descendiendo por el tablero de juego progresivamente a una velocidad que se irá incrementando a medida que aumente el nivel.

La ficha en juego se muestra en un color distinto a las que ya quedaron fijas en el tablero, para no confundirlas (Figura 2).



## ALGORITMOS Y PROGRAMACIÓN

### PRIMER TRABAJO PRÁCTICO

Prof. Mg. Ing. F. J. LAGE

Alumno: Rinaldi, Lautaro Ezequiel.

Nº Padrón:

A.T.P.

Lic. Esteban Calabria

Ing. Guido Costa

Ing. Fernando Salgueiro

Primer Entrega

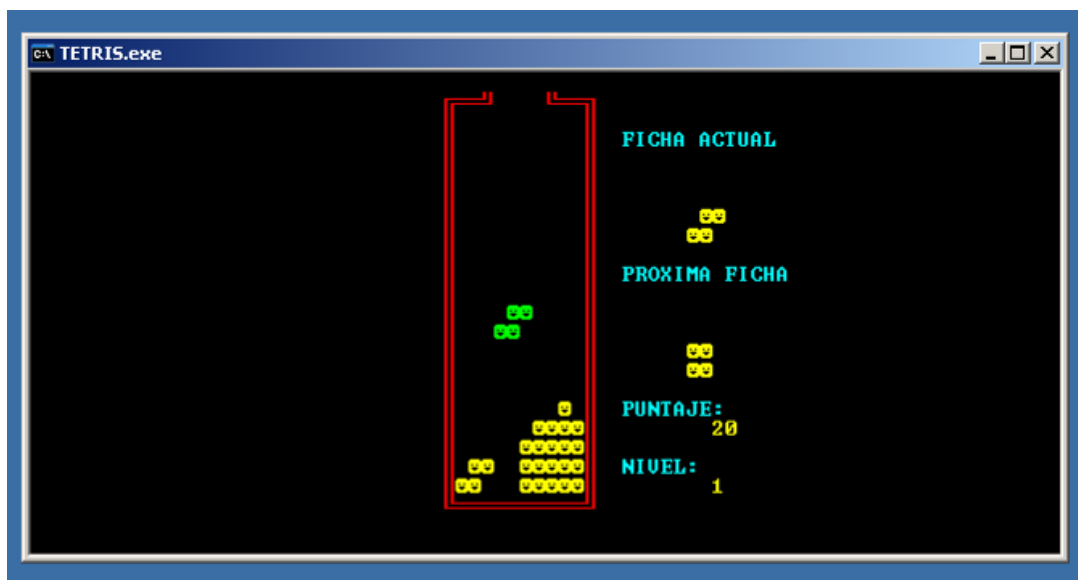


Figura 2: Color de Ficha en Juego.

A su derecha, usted podrá visualizar la ficha actual, para que, apenas comienza a bajar, usted ya sepa de cual se trata y no las confunda (es común no haber mirado con anterioridad cuál va a ser la próxima ficha u olvidarlo).

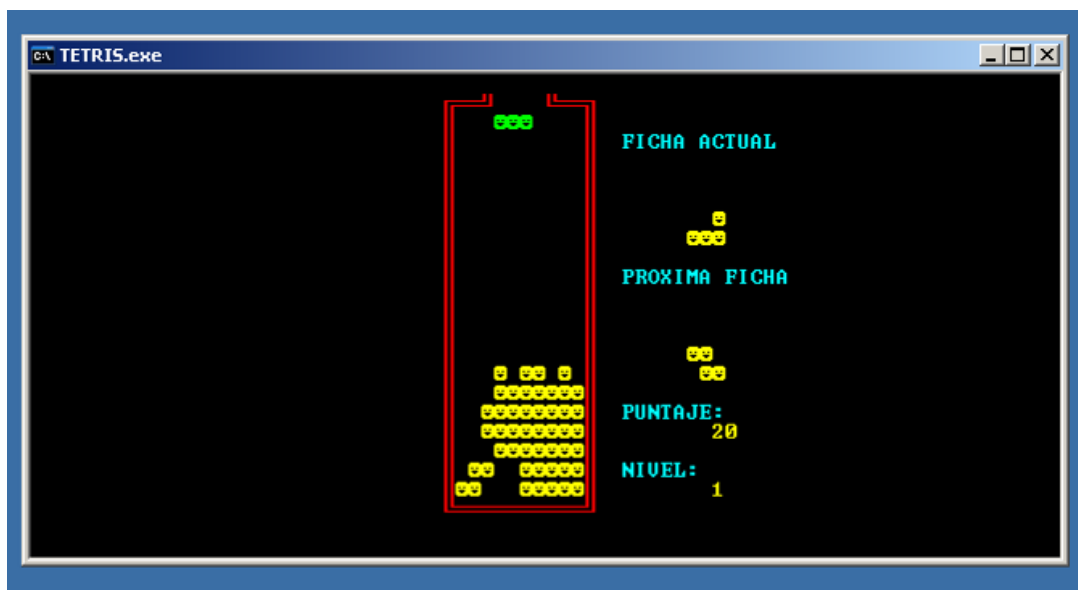


Figura 3: Ficha Actual apenas aparece en tablero.



## ALGORITMOS Y PROGRAMACIÓN

### PRIMER TRABAJO PRÁCTICO

Prof. Mg. Ing. F. J. LAGE

Alumno: Rinaldi, Lautaro Ezequiel.

Nº Padrón:

A.T.P.

Lic. Esteban Calabria

Ing. Guido Costa

Ing. Fernando Salgueiro

Primer Entrega

Un ejemplo de su utilidad se muestra en la Figura 3, donde de no estar la ficha actual, usted no sabría si se trata de:



Figura 4: Confusión de Fichas.

También visualiza debajo de la ficha actual, la próxima ficha que descenderá por el tablero de juego, en caso de que con la actual no pierda la partida.

Las fichas pueden ser rotadas 0°, 90°, 180°, 270° (Figura 5), en sentido contrario al de las agujas del reloj, presionando la tecla "Z" (Figura 6) una o más veces seguidas (pudiendo apreciar sus cambios tanto en el tablero de juego como en sector de ficha actual), siempre que al rotar la ficha no quede superpuesta con ninguna otra, o no se exceda de los límites del tablero de juego.



Figura 5: Rotación de Fichas.



Figura 6: Tecla "Z".

También puede desplazarlas hacia la derecha (Figura 7) o izquierda (Figura 8) con las teclas de dirección o incrementar la velocidad de bajada de las fichas manteniendo presionada la tecla abajo (Figura 9).



## ALGORITMOS Y PROGRAMACIÓN

### PRIMER TRABAJO PRÁCTICO

Prof. Mg. Ing. F. J. LAGE

Alumno: Rinaldi, Lautaro Ezequiel.

Nº Padrón:

A.T.P.

Lic. Esteban Calabria

Ing. Guido Costa

Ing. Fernando Salgueiro

Primer Entrega



Figura 7: Tecla Derecha.



Figura 8: Tecla Izquierda.



Figura 9: Tecla Abajo.

Durante cualquier momento de la partida puede presionar la tecla ESC (Figura 10), y el programa le mostrará un alerta (Figura 11) dándole la opción de continuar la partida actual o salir (Presionando “S” (Figura 12) continua, y con cualquier otra tecla vuelve al menú principal).



Figura 10: Tecla ESC.



## ALGORITMOS Y PROGRAMACIÓN

### PRIMER TRABAJO PRÁCTICO

Prof. Mg. Ing. F. J. LAGE  
Alumno: Rinaldi, Lautaro Ezequiel.  
Nº Padrón:

A.T.P.  
Lic. Esteban Calabria  
Ing. Guido Costa  
Ing. Fernando Salgueiro

Primer Entrega

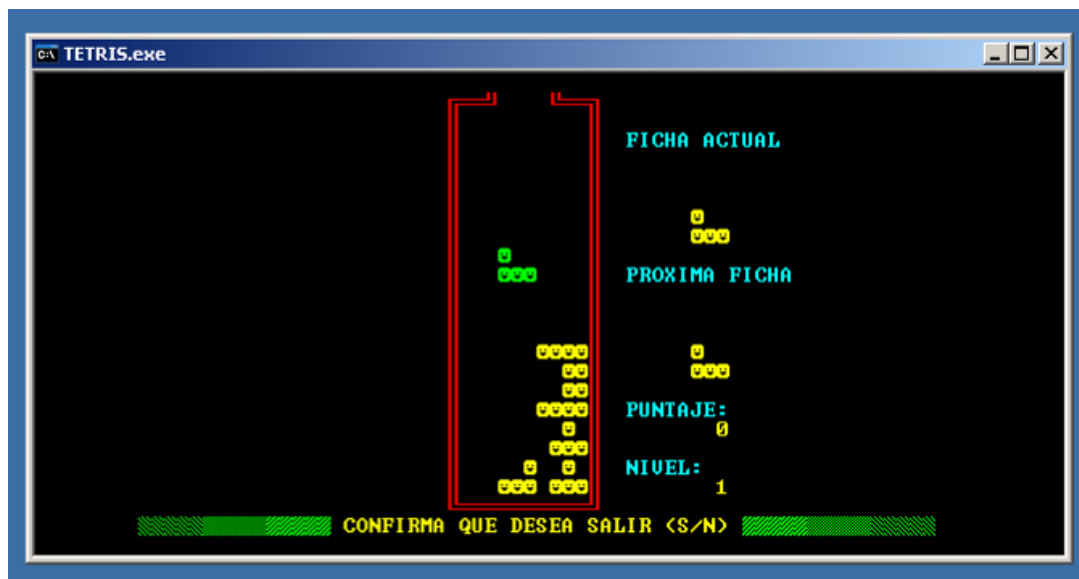


Figura 11: Alerta en Pantalla.



Figura 12: Tecla "S".

Usted debe decidir como colocar las fichas de forma que se formen líneas completas. Cuando una línea horizontal se completa, esa línea desaparece y todas las piezas que están por encima descienden una posición, liberando espacio de juego y por tanto facilitando la tarea de situar nuevas piezas.

El juego consta de 5 niveles, en donde la velocidad de caída de las piezas se va acelerando agregándole dificultad al juego. La condición para incrementar el "nivel" es que el jugador complete 5 líneas como mínimo.

El puntaje que obtiene por borrar una línea se calcula como 10 veces el nivel.



## ALGORITMOS Y PROGRAMACIÓN

### PRIMER TRABAJO PRÁCTICO

Prof. Mg. Ing. F. J. LAGE

Alumno: Rinaldi, Lautaro Ezequiel.

Nº Padrón:

A.T.P.

Lic. Esteban Calabria

Ing. Guido Costa

Ing. Fernando Salgueiro

Primer Entrega

#### Opción 2:

Le explica la mecánica del juego en sí misma (Figura 13). Presionando cualquier tecla regresa al menú principal.

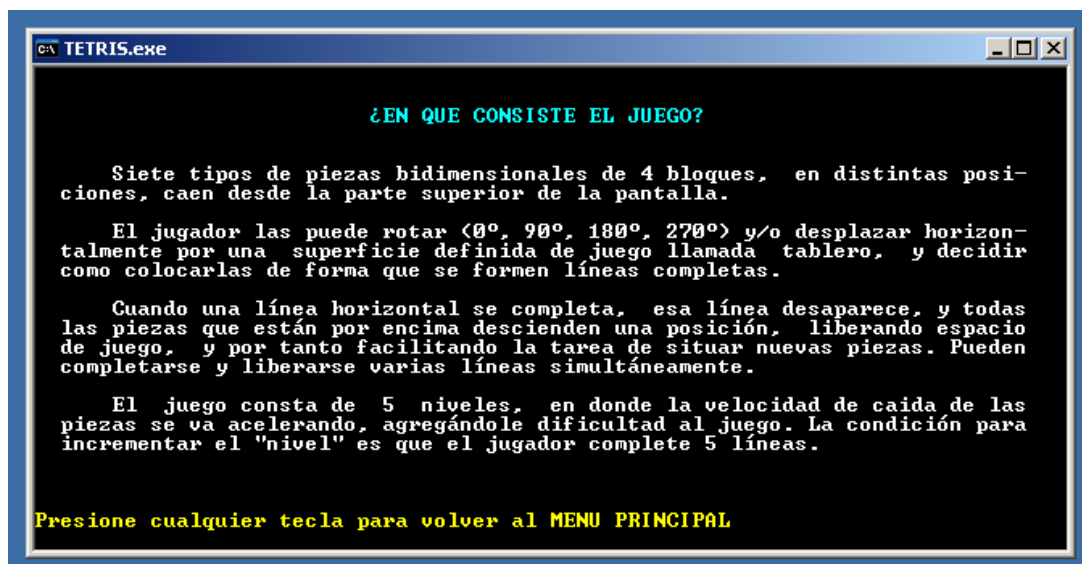
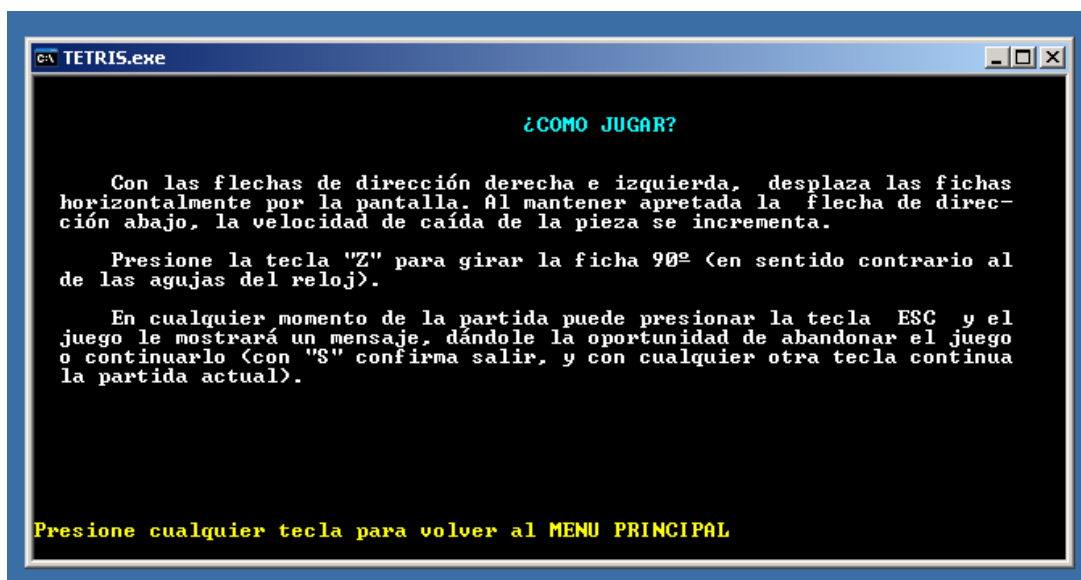


Figura 13: Opción 2, ¿En qué consiste el juego?

#### Opción 3:

Le explica los métodos de teclado para girar una ficha, desplazarla horizontalmente o acelerar su velocidad de caída. Presionando cualquier tecla regresa al menú principal.





## **ALGORITMOS Y PROGRAMACIÓN**

### **PRIMER TRABAJO PRÁCTICO**

Prof. Mg. Ing. F. J. LAGE

Alumno: Rinaldi, Lautaro Ezequiel.

Nº Padrón:

A.T.P.

Lic. Esteban Calabria

Ing. Guido Costa

Ing. Fernando Salgueiro

Primer Entrega

#### **Opción 4:**

Sale completamente del juego sin pedir ningún tipo de confirmación.