

## Proyecto: "PONJ"

Desarrollar un programa en lenguaje ensamblador para arquitectura Intel x86 que funcione como un juego de Pong™, imprimiendo en pantalla una interfaz gráfica que permita la interacción computadora-usuario a través del *mouse* y del teclado de la computadora. Se deberá mostrar en pantalla la información básica de un juego de Pong™: dos barras controladas a través del teclado que representan a cada uno de los jugadores, una "pelota" que rebota dentro del área de juego, el marcador por cada jugador, y un temporizador; también deberá mostrar algunos botones que permitan controlar cuándo inicia o termina el juego.

### Qué debe hacer el programa:

El programa debe imprimir una interfaz gráfica que muestre un juego de Pong™, donde las barras de juego (jugadores) se puedan controlar con el teclado de la computadora, y que contiene algunos botones con los que se puede interactuar a través del *mouse*.

Dentro de la interfaz gráfica se muestra la información de juego y algunos botones para comenzar o detener el juego, estos botones deben funcionar a través del *mouse*.

El objetivo del juego es hacer que una pelota que se encuentra rebotando dentro del área de juego llegue al extremo opuesto a un jugador, que defiende su extremo controlando una barra. La barra hace que la pelota rebote en sentido contrario, enviando la pelota hacia el lado opuesto. Si la pelota logra pasar a un extremo sin tocar la barra de un jugador, se incrementa el marcador del jugador opuesto.

El tiempo de juego se muestra en un temporizador y cuando el juego se encuentra activo, el temporizador se decrementa cada segundo. Cuando el tiempo llega a 0:00, el juego termina y la pelota se detiene. El ganador será aquel jugador cuyo marcador sea mayor.

En la interfaz gráfica se deben mostrar (al menos) 2 botones, uno de ellos debe permitir iniciar el juego y el otro detenerlo y reiniciar tiempo y marcadores.

Cuando se presiona el botón para iniciar el juego, el temporizador debe comenzar su cuenta regresiva hasta llegar a 0:00. Durante este periodo, la pelota debe estar en movimiento, de manera que ésta avanza de un lado a otro en movimientos diagonales y rebotando dentro del área de juego. Mientras el juego está activo, los jugadores pueden marcar puntos haciendo rebotar la pelota. Cuando el temporizador llegue a 0:00 el juego termina y la pelota se detiene.

Si durante un juego activo, el usuario presiona el botón de parar o detener, el juego se detiene y reinicia el tiempo y los marcadores de cada uno de los jugadores (es opcional reiniciar la posición de la pelota).

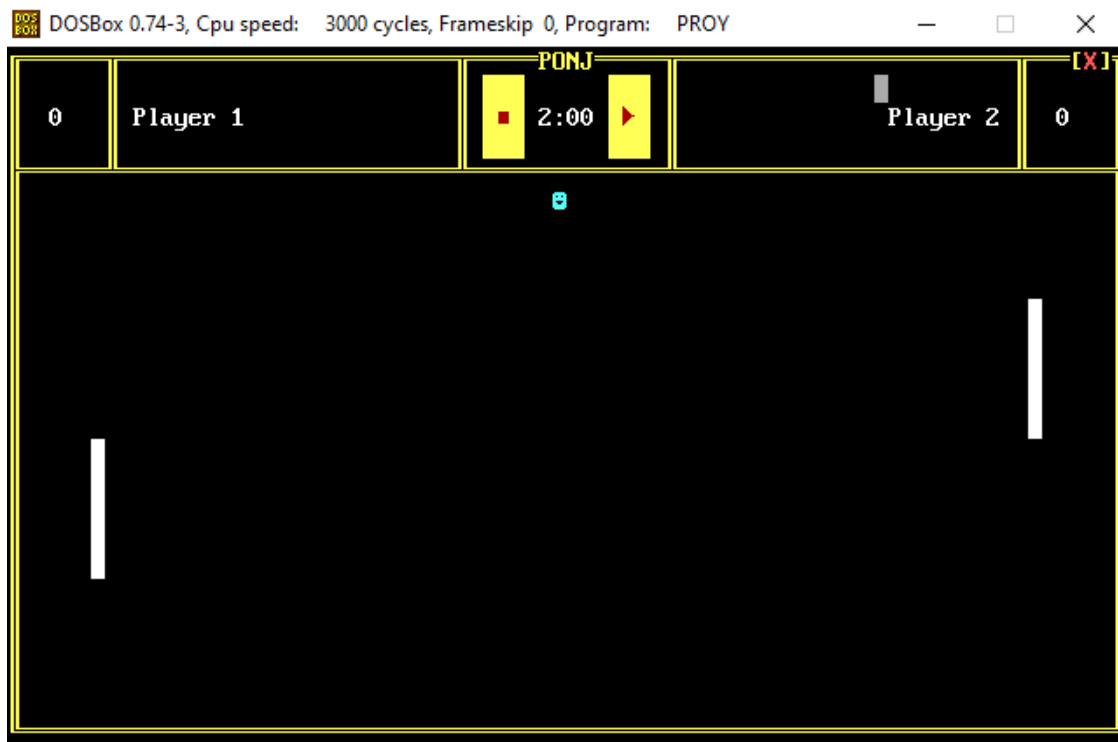
El movimiento de cada barra de un jugador se debe realizar a través del teclado, los movimientos son verticales únicamente, por lo que por cada jugador se requieren dos teclas para mover su barra. Estas teclas pueden ser las que ustedes decidan. Una condición importante es que el desplazamiento de cada jugador solo se haga dentro del área de juego, es decir, agregar restricciones para que el desplazamiento no se haga fuera del área.

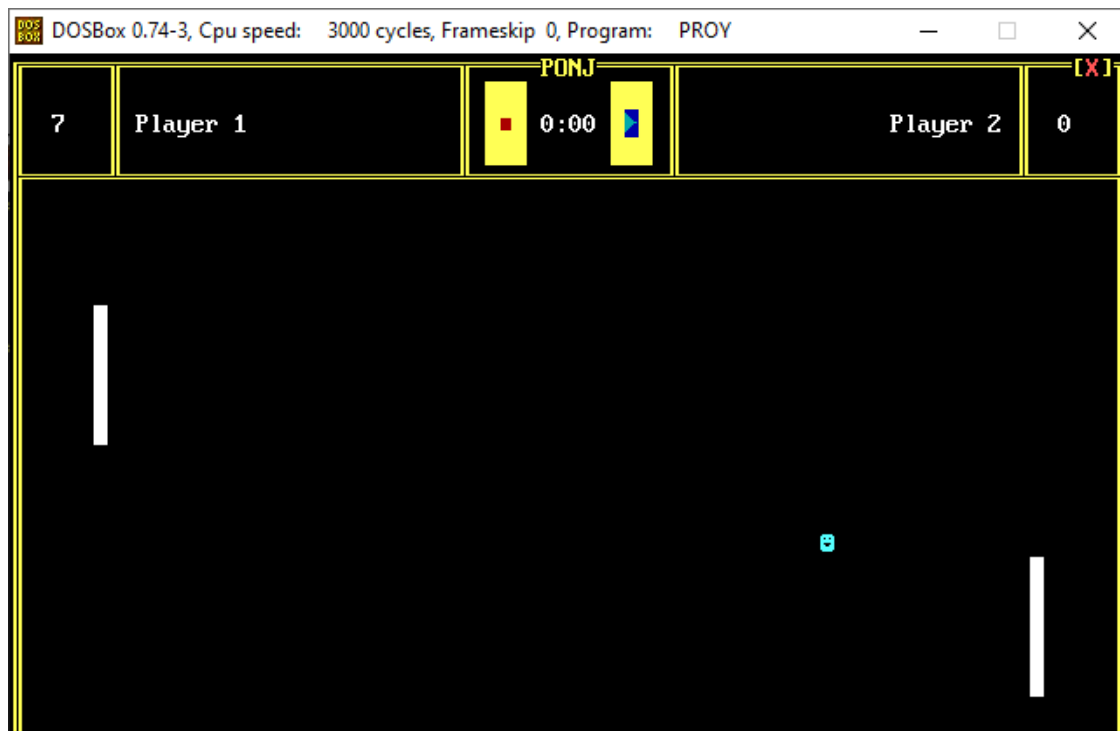
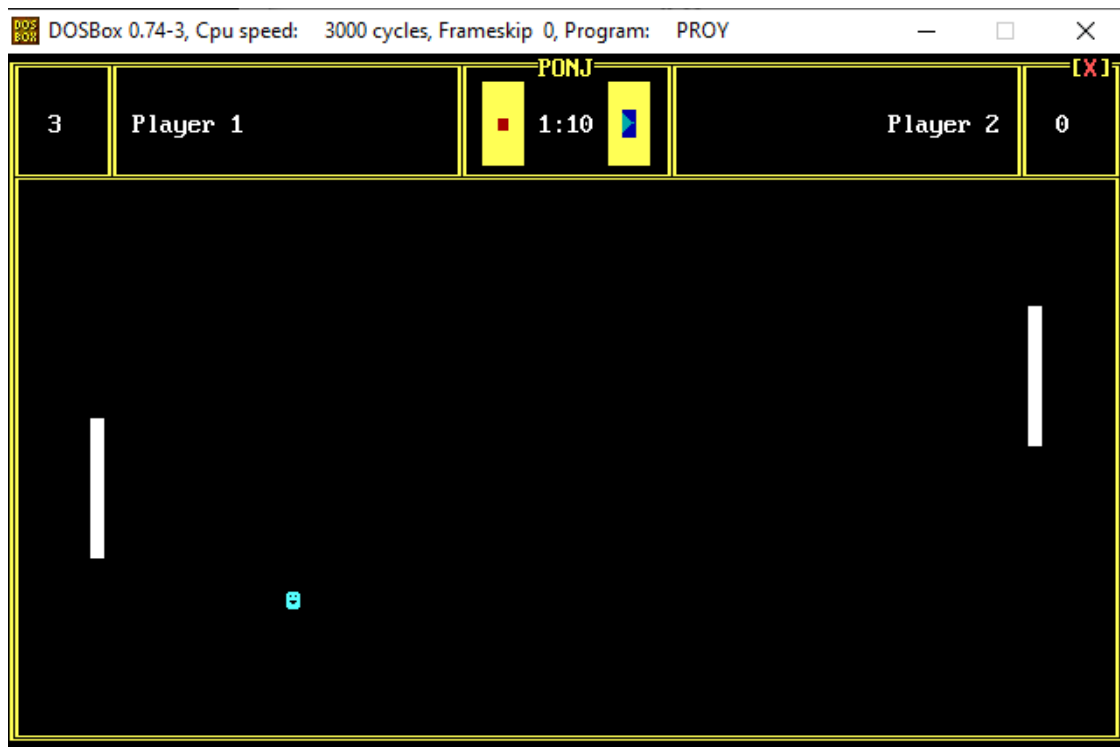
Para finalizar la ejecución del programa, el usuario debe hacerlo a través de la interfaz gráfica, es decir, utilizando el *mouse*.

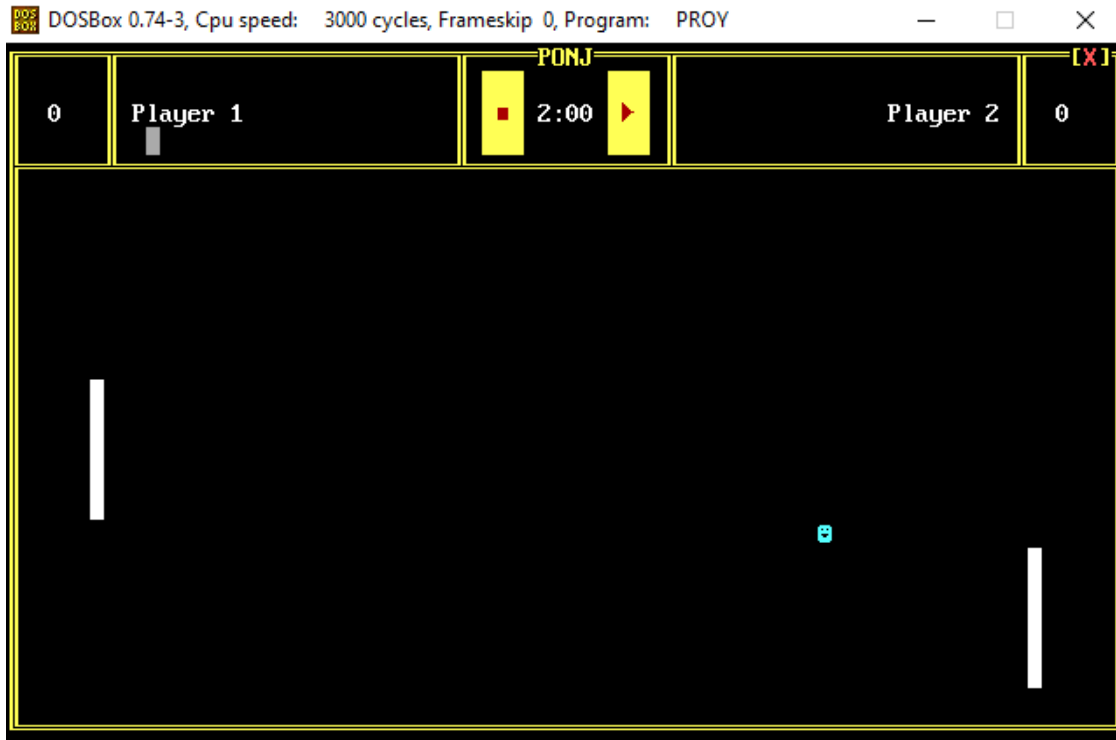
## Indicaciones

- El área de uso del *mouse* de estar delimitada a la zona que no pertenece al área del juego. El *mouse* no debe interferir en la vista del juego.
- El temporizador se debe mostrar en formato M:SS, donde M es el número de minutos restantes y SS los segundos (entre 00 y 59). El tiempo de inicio puede estar definido dentro del programa.
- Funcionalidades extra que no estén especificadas en lo anterior se tomarán en cuenta, pero el programa deberá cumplir con lo solicitado, al menos, para ser consideradas. Se puede obtener hasta un punto extra, cada funcionalidad extra tendrá valor de medio punto.

## Ejemplos:







El programa final no necesariamente debe verse igual que el ejemplo anterior, pero sí debe realizar y mostrar lo que se solicita en este documento.

#### Consideraciones:

- La interrupción 10h permite trabajar con servicios de pantalla. Permite establecer modo de video de la pantalla, resolución, paleta de colores, etc.
- El temporizador se puede manejar utilizando la int 1Ah, específicamente opciones 00h y 01h. Investigar su funcionamiento.
- Cuando se introduce un carácter desde el teclado se obtiene su valor en ASCII (ver tabla ASCII).
- Los caracteres numéricos tienen el valor ASCII siguiente:

Carácter numérico	ASCII (hexadecimal)
'0'	30h
'1'	31h
'2'	32h
'3'	33h
'4'	34h
'5'	35h
'6'	36h
'7'	37h
'8'	38h
'9'	39h

Para imprimirlos en pantalla, se debe utilizar su valor ASCII.

### Funcionalidades extra (sugeridas) [opcionales]

Esta es una lista de posibles funcionalidades agregadas. Cualquier otra funcionalidad se acepta.

- Agregar botones para modificar el tiempo límite de un juego.
- Modificar la pendiente y la velocidad de rebote de la pelota dependiendo la zona de rebote en un jugador.
- Una pantalla que muestre el resultado cuando finalice el juego, indicando al ganador y con la opción de volver a la pantalla de inicio e iniciar un juego nuevo.
- Que cada jugador pueda modificar su *username* dentro de la interfaz gráfica cuando el programa se encuentra en ejecución.
- Que uno de los jugadores sea controlado por la CPU (**sin que sea imposible ganarle**).
- Que el juego funcione a través de la red, para que dos jugadores conectados desde diferentes computadoras puedan participar en una misma partida, obviamente sincronizada en cada computadora (esta funcionalidad se sale del objetivo del curso y requiere otros conocimientos, pero quien lo logre tendrá puntos extra asegurados).

### Entregables:

La entrega de proyecto se puede hacer de manera individual o en equipo de hasta 3 integrantes.

Se deberá subir a Google Classroom lo siguiente:

- Documento
  - o Introducción
    - Descripción y planteamiento del problema
  - o Desarrollo
    - Planteamiento y justificación de la solución
    - Diagrama de flujo y pruebas de escritorio
  - o Conclusión
    - Conclusiones individuales (por cada integrante del equipo)
- Código fuente (extensión .asm)
- Archivo ejecutable (extensión .exe)
- Archivo de texto plano (.txt) con el nombre de los integrantes del equipo.

\*Cada integrante debe subir una copia del proyecto, con sus debidos archivos.

**La fecha límite para la subida de los archivos será el día 16 de agosto de 2021, antes de las 10:00 hrs. No se revisarán entregas retrasadas.**

Además de la entrega de los archivos antes mencionados, se deberá realizar una muestra 'presencial' en la que los integrantes del equipo mostrarán el funcionamiento de su proyecto y responderán a los cuestionamientos hechos por el profesor. De manera pseudoaleatoria, el profesor puede seleccionar a un integrante del equipo y hacer un cuestionamiento. Si la respuesta no es satisfactoria, puede tener repercusión en el equipo e individualmente.

La fecha de la muestra presencial será el día **16 de agosto de 2021, a partir de las 11:00 hrs y a través de Google Meet (en la liga de clase)**. A cada equipo se le asignará un número y el orden en el que

deberán mostrar su proyecto. La asignación se hará días previos a la muestra presencial. Cada equipo tendrá un máximo de 15 minutos para hacer su revisión. Al momento de la revisión, solo deberán estar conectados los integrantes del equipo dentro de la plataforma.

Será requisito que todos los integrantes del equipo estén presentes en la muestra presencial, no se hará revisión si falta algún integrante. Se les solicita que preparen lo necesario previo a su revisión para agilizarla. Consideren que, dependiendo el número de equipos, la muestra presencial se puede extender varias horas.