

## Proyecto II

### Computer Engineering

#### Star Force Remake, May the Force Be With You

*Trying to read our DNA is like trying to understand software code - with only 90% of the code riddled with errors. It's very difficult in that case to understand and predict what that software code is going to do.*

*Elon Musk*

*To make an embarrassing admission, I like video games. That's what got me into software engineering when I was a kid. I wanted to make money so I could buy a better computer to play better video games - nothing like saving the world.*

*Elon Musk*

*My vision is for a fully reusable rocket transport system between Earth and Mars that is able to re-fuel on Mars - this is very important - so you don't have to carry the return fuel when you go there.*

*Elon Musk*

## Introduction

The second computer programming project is based on the game Star Force. The game was released for arcade machines, by Video Ware as Mega Force [1], in 1984. It was also published in 1985 by Hudson Soft, both for the home computer (home desktop, nowadays) MSX, and for the Family Computer (Famicom) in Japan. The North American version for Nintendo Entertainment System (NES), was released in 1987 by Tecmo. [2].

## Introducción

El segundo proyecto programado está basado en el juego Star Force. El juego fue lanzado para las máquinas arcade, por Video Ware como Mega Force [1], en 1984. También fue publicado en 1985 por Hudson Soft, tanto para la computadora del hogar (home desktop, hoy día) MSX, como para la Family Computer (Famicomp) en Japón.

La versión norteamericana para Nintendo Entertainment System (NES), fue liberada en 1987 por Tecmo. [2].

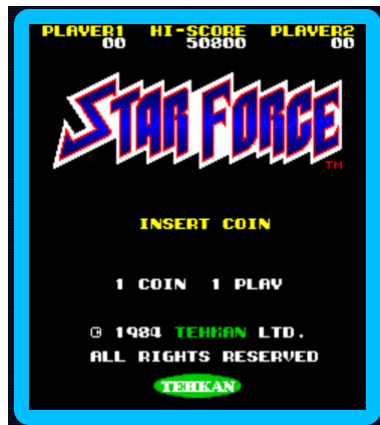


Figura 1. Versión de Tekhan del juego Star Force, corría en un MP Z80 de 4MHz [3].

## Descripción

El juego estará compuesto por una pantalla principal, en la cual se escoge al piloto que se desea utilizar, después el usuario deberá escoger el modo de juego de su preferencia. Los modos de juego son: destrucción de asteroides y la prueba de maniobras.

Se tienen un total de 20 pilotos diferentes para escoger. Cada vez que se seleccione alguno de los 2 modos de juego, se les asignará un puntaje aleatorio a los 20 pilotos, usted deberá ordenar los resultados mediante alguno de los algoritmos vistos en clase. No se pueden utilizar funciones propias de python para ordenar los resultados. Los valores generados aleatoriamente deben estar en un intervalo razonable, de forma que no se den valores muy altos.

Una vez se ha seleccionado el modo, se debe continuar guardando las 5 mejores puntuaciones de usuarios. Esto quiere decir que no se cuentan los puntajes asignados aleatoriamente. Estos puntajes se deben visualizar de forma ordenada de mayor a menor en la ventana de altos puntajes.

Un jugador puede personalizar el nombre del piloto desde la pantalla de configuración. La nave que utilizan los pilotos, en ambos modos de juego, necesita de energía para seguir funcionando. Fuentes de energía deben aparecer de forma aleatoria en la pantalla, cada cierto tiempo. Conforme el jugador avanza el tiempo de juego más

demora en aparecer la energía. El jugador debe pasar sobre la energía para obtenerla.

Los modos de juego se explican a detalle en la sección de pantalla de juego.

### **Importante conexión con el Proyecto III**

El proyecto 3 utilizará los módulos de este proyecto y se integrarán. Es por esto que debe crear las funciones de indicadores y manejo de la nave de forma modular y que puedan recibir y enviar señales externas.

A continuación se explican las diferentes pantallas del juego:

#### Pantalla principal

Esta pantalla es la primera que se muestra al usuario una vez se ejecuta el programa, en la misma se debe poder acceder a las pantallas de: acerca de, altos puntajes, configuración y al modo de juego seleccionado. Además, debe permitir al usuario cerrar el programa.

Debe tener una buena presentación, una imagen alusiva al juego y una melodía acorde.

#### Pantalla About

En esta pantalla se debe incluir nombre de la institución, carrera, nombre del profesor, estudiante, carnet, versión, fecha de última modificación y una fotografía personal. Se debe poder retornar a la pantalla principal.

#### Pantalla altos puntajes

En esta pantalla se deben de mostrar los 5 mejores puntajes. Los puntajes deben ser almacenados en un archivo de texto (.txt). Se debe poder retornar a la pantalla principal.

#### Pantalla de Configuración

En esta pantalla se podrán personalizar los nombres de los 20 pilotos que puede escoger el usuario. Se debe poder retornar a la pantalla principal.

## Pantalla de juego

En esta pantalla se debe escoger al piloto a utilizar y el modo de juego deseado. Como se mencionó anteriormente existen 2 modos de juego, destrucción de asteroides y la prueba de maniobras.

En ambos modos de juego el usuario puede ver la cantidad de energía restante. La misma disminuye con el paso del tiempo, situación que debe verse en pantalla mediante algún indicador (analógico o de barras). Además del piloto seleccionado y el puntaje obtenido.

## Destrucción de asteroides

La nave se encuentra en el espacio, en medio de un cinturón de asteroides, el usuario debe poder esquivar o destruir los asteroides. Un ejemplo se puede apreciar en la Figura 1, los asteroides vienen de frente a la nave. Cada asteroide destruido suma puntos, los que se esquivan no suman, la cantidad de asteroides restantes debe aparecer en pantalla en un sitio adecuado para no interferir con el juego. El máximo de asteroides es de 9.

Si un asteroide golpea a la nave el juego termina y pasa a mostrar el resultado. Se debe devolver a la pantalla inicial para continuar jugando.



Figura 2. Visualización de la pantalla con asteroides [2]

### Prueba de maniobras

Como se puede ver en la figura 2, en este modo de juego aparecen aros por los cuales el jugador debe pasar. A medida que la nave se aproxima el aro, éste se hace más grande, pero no tan grande que cubra el área de la pantalla. Si se choca con el borde del aro, se termina el juego. Si no se pasa en medio del mismo ese aro no sumará puntaje.



Figura 3. Ejemplo de una nave y los aros por los que debe pasar en su ruta, para obtener puntos. [2]

### Aspectos generales

- Se debe poder perder y ganar el juego. Su dificultad **NO** debe ser imposible ni nula.
- Cualquier duda, omisión o contradicción en la especificación se debe aclarar con el profesor y se difundirá a través del asistente.
- El proyecto es en **PAREJAS**. Y se deberá entregar el día: Jun 1 2018. Cualquier indicio de copia significará un cero en la nota.
- Se debe imprimir la documentación (archivo .doc) ésta debe contar con lo siguiente: Introducción; Conclusiones; Recomendaciones; Análisis de Resultados que debe incluir: bitácora, resultados; y Literatura o Fuentes Consultadas.
- Para la realización de la parte gráfica se pueden usar bibliotecas complementarias Python (Tkinter) u otras que considere de ayuda.

- El código debe estar suficientemente documentado de tal forma que Ud. se pueda orientar en el fácilmente durante la defensa.
- Se debe presentar la tabla de evaluación a la hora de defender el proyecto.
- Se deben definir los requerimientos mínimos si se desea revisar en una computadora de un laboratorio y avisar con antelación para poder ejecutar el programa.
- Se evaluarán el código y la modularización.

## Bibliografía

[1] juegos.com. (2018). Star Force. [online] disponible en: <http://www.juegos.com/juego/star-force>. S. F. Obtenido 17 May 2018.

[2] "Hudson - Shooting game [NES] (Archive)". Hudson. Hudson. S. F. Obtenido el 16 Mayo 2018.

[2] <https://www.arcade-history.com/?n=star-force&page=detail&id=2614> Obtenido May 18 2018.