

Proyecto 1: The Bootloader

Tecnológico de Costa Rica
Escuela de Ingeniería Electrónica
Profesor: Ernesto Rivera Alvarado

I. INTRODUCCIÓN

En este proyecto, el estudiante se expondrá a los detalles de una arquitectura computacional sin sistema operativo y del proceso que conlleva inicializar un procesador. El *bootloader* es un código que se ejecuta antes del sistema operativo para inicializar el procesador. El objetivo de este proyecto es que el estudiante pueda correr una aplicación específica en una computadora de uso general o empotrada que no tenga sistema operativo. La aplicación específica que ejecutará será el juego *shooter* “Lead”. En el BIOS de la computadora se deshabilitarán los discos duros y el juego se correrá desde una llave USB.

II. DETALLES IMPORTANTES

- El objetivo de este proyecto es que el estudiante explore las diferentes opciones de bootloaders que existen tanto para la arquitectura x_86 como para ARM.
- El estudiante puede escoger cualquier de estas dos arquitecturas para realizar el proyecto.
- Es de suma importancia que entienda a cabalidad el *bootloader* que utilizó, por lo que se le solicita que haga un resumen que explique los detalles indispensables del *bootloader* utilizado.
- En ambas arquitecturas se requiere que el procesador al menos opere en modo de 32 bits.
- Como términos para buscar información puede leer la respuesta en Quora que se adjunta en el siguiente link: <https://www.quora.com/Where-can-I-learn-bare-metal-coding-on-x86-I-want-to-use-C-as-well-as-try-other-languages>
- En general puede utilizar la información y guías proporcionadas en los siguientes links:
 - [https://github.com/cirosantilli/x86-bare-metal-examples](https://github.com/cirosantilli/x86-bare-metal-examples/tree/bc74cd1819ffbd8beccabeea0b82f9ba7014f74#multiboot-hello-world)
 - <https://github.com/cirosantilli/x86-bare-metal-examples>
 - <https://www.raspberrypi.org/forums/viewforum.php?f=72>
- El término de búsqueda sobre lo que ustedes quieren hacer se denomina “*bare metal programming*”.
- Pueden utilizar los lenguajes de programación ensamblador o C.

III. DOCUMENTACIÓN SOLICITADA

El estudiante deberá presentar la siguiente documentación:

- Resumen (incluyendo diagramas) que explique el modo real, protegido y anillos de protección de la arquitectura x_86.
- Resumen que explique con detalle las características del *bootloader* que utilizó.

IV. DESCRIPCIÓN DEL JUEGO *shooter* “LEAD”

Todos los gráficos del juego serán en modo texto. Para referencia, los estudiantes pueden revisar el siguiente video de YouTube que muestra la dinámica del juego:

<https://www.youtube.com/watch?v=pJuJIqd1Uhw>

Dentro de los requisitos, el juego debe mostrar una pantalla de inicio que indique los integrantes del grupo, el nombre del curso, profesor y nombre del juego. Para iniciar el juego, se debe presionar cualquier tecla en la pantalla de inicio. El juego tendrá una resolución suficiente para incorporar la cantidad de enemigos y las paredes límite que se muestran en el video de YouTube adjunto. Para diferenciar entre distintos enemigos, el estudiante puede hacer grupos de distintos caracteres, por ejemplo para denotar un enemigo puede utilizar:

XX

De la misma manera, debe seleccionar los caracteres que utilizará para el jugador y las balas. Dentro de la mecánica del juego, el estudiante debe incorporar el “*score*”, que se muestra en la parte inferior de la pantalla. El juego incluirá los niveles con la complejidad que se muestra en el video. Considere que el juego debe ejecutarse fluidamente y no podrá bloquearse a la espera de una entrada del teclado.

V. REQUISITOS INDISPENSABLES

- Todo el código debe de estar escrito en ensamblador o en C.
- Correr el programa en QEMU en vez de una memoria USB conllevará una penalización de 10 puntos de la nota.

VI. EVALUACIÓN Y COMPLETITUD DEL PROYECTO

Todas las cualidades observadas en el video de referencia deben integrarse en un solo proyecto. Por cada característica que falte se pierden 2^k puntos, donde k es el número de características que usted no incluyó. Esto aplica para $k > 1$. Se hace la aclaración de que es inadmisibles la presentación de diferentes características en programas diferentes que no estén incluidas en un solo proyecto. La evaluación de este proyecto está sujeta a la presentación oral del mismo, en la que el estudiante debe mostrar un dominio completo del trabajo realizado a nivel de código.

VII. DESARROLLO DEL PROYECTO

El proyecto está pensado para desarrollarse individualmente, sin embargo, los estudiantes que deseen reforzar la habilidad de trabajo en equipo pueden entregar el proyecto en grupos de hasta tres personas. El profesor les hace la aclaración de que el trabajo con compañeros conlleva dificultades de coordinación, división de trabajo y sobre todo de “pegar o juntar ambas partes”. En experiencias propias del profesor, se les comenta que en ocasiones el trabajo de juntar, acoplar y corregir partes desarrolladas por diferentes personas conlleva más tiempo y trabajo que la realización individual.

VIII. FECHA DE ENTREGA

Las demostraciones se harán en **semana 5** en clase. No se le revisará a los estudiantes que no se encuentren al inicio de la clase.

La carpeta comprimida .zip de su proyecto debe contener únicamente los archivos fuentes y un archivo de texto que describa con detalle el procedimiento para ejecutar y echar a andar el proyecto. El nombre de la carpeta comprimida serán los apellidos de los integrantes del grupo de trabajo (por ejemplo `rivera-alvarado.zip`) y este debe de ser subido al Tec-Digital el día de la revisión antes de las 6 am.