

Instituto Tecnológico de Costa Rica Área Académica de Ingeniería en Computadores (Computer Engineering Academic Area) Programa de Licenciatura en Ingeniería en Computadores (Licentiate Degree Program in Computer Engineering) Curso: CE-5303 Introducción a los Sistemas Embebidos (Course: CE-5303 Introduction to Embedded Systems) Profesor: M.Sc. Ing. Jeferson González Gómez. (Professor) Semestre: II, 2019 (Semester) Nombre de estudiante: _____ (Student's full name) Carné: _____ (Student's ID) Nombre de estudiante: _____ (Student's full name) Carné: _____ (Student's ID)	Examen Final (Final Exam) Fecha: 8 de noviembre de 2019 (Date) Grupo: 1 Valor: 50 pts. (Value: 50 pts.) Puntos obtenidos: _____ (Score) Nota: _____ (Percentage)
-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

INSTRUCCIONES GENERALES.

- Trabaje individual o en parejas.
- La modalidad de entrega de este examen es a través del TecDigital.
- Cada estudiante debe subir una solución a este examen, aunque el trabajo se realice en parejas.
- Adjunte este enunciado como parte de su solución.

1. I Parte. Respuesta breve. 10 puntos

A continuación, se le presentan 10 preguntas de respuesta corta. Responda cada una de ellas de manera correcta. (1 punto c/u)

1. Mencione 2 características que deben tener los sistemas operativos para sistemas embebidos.
2. Lucía trata de portear el paquete alsa-utils para raspberry pi zero, utilizando Yocto. Al realizar el comando `bitbake rpi-basic-image`, tiene el siguiente error: "Nothing provides alsa-utils". ¿Cómo podría solucionarle el error a Lucía? (ya Lucía se aseguró que el nombre del paquete está bien escrito)
3. ¿A qué comando o acción, en espacio de usuario, corresponde el método `module_open(struct inode *inode, struct file *filp)`?
4. Escriba 2 características que deben tener los Modelos de Computación para describir el comportamiento de sistemas embebidos.
5. En el comando `$(CC) -o this isright.cpp -I../include -L../lib -lsomething` ¿Cuál es el nombre de archivo completo de la biblioteca dinámica enlazada?
6. Escriba un comando (bash) que escriba en un archivo `dir.log` una lista de archivos y subdirectorios de un directorio actual.
7. Mencione dos características que tienen que tener los modelos de computación (MoC) para modelar Sistemas Embebidos?
8. ¿Qué es un toolchain?

9. Al tratar de compilar un código fuente, Juanito tiene el siguiente error “undefined reference to ‘sqrt’ ”. Juanito dice que “ya incluyó math.h”, pero el error no se corrige. ¿Cómo solucionaría el problema que tiene Juanito?
10. ¿Cuál es la diferencia principal un driver en espacio de kernel y uno en espacio de usuario?

2. II parte. Desarrollo (40 puntos)

A continuación, se le presentan 2 problemas de desarrollo, los cuales se deben contestar con el mayor nivel de detalle posible, mostrando todos los pasos requeridos para llegar a la solución. Se deberá enviar un archivo con las respuestas de los ejercicios, así como el archivo de programación correspondiente. Puede trabajar individual o en parejas si así lo desea. De ser en parejas, debe asegurarse que el documento lleve el nombre y carné de ambos integrantes.

1. El código exam.c presenta una descripción del comportamiento de un sistema embebido a diseñar, para la adquisición, filtrado y procesamiento de una señal analógica de sonido. A partir de dicha descripción:
 - a) Realice un modelo (gráfico) de alto nivel del código utilizando máquinas de estado-proceso (PSM). No es necesario que muestre la descripción totalmente funcional de cada proceso, pero sí debe describir qué hace en alto nivel. Represente adecuadamente elementos de jerarquía y concurrencia (5 puntos)
 - b) Muestre detalladamente los pasos de una síntesis manual, para obtener un modelo estructural de un sistema que pueda implementar dicha aplicación. Asuma que seguirá una metodología de diseño basado en plataforma, para un dispositivo Altera Cyclone® V SE 5CSEMA4U23C6N (ver Altera De0-Nano-SoC, como referencia). Puede utilizar componentes básicos de sistema, para dicha descripción (memorias, buses, procesadores, sistemas operativos, bloques de IP, HW específico, etc), siempre y cuando estén disponibles de una forma u otra en el dispositivo. Muestre el diagrama del modelo estructural del sistema obtenido luego de la síntesis. Justifique detalladamente toda decisión de de diseño. (20 puntos)
2. Parte del proceso de encodificación JPEG para frames de imágenes en sistemas embebidos, corresponde al algoritmo de run-zero encoding (RZE). En este tipo de encodificación, ante un arreglo de elementos discretos, proveniente de un FIFO, se deberá generar un arreglo salida que contenga, para cada número natural x (no incluye al 0), la cantidad de ceros que haya hasta x , seguido del número entero x correspondiente.

Por ejemplo,

$in = [0, 0, 1, 0, 3] \rightarrow out = [2, 1, 1, 3]$

$in = [1, 2, 3] \rightarrow out = [0, 1, 0, 2, 0, 3]$

$in = [0, 0, 0, -1, 255, 3] \rightarrow out = [3, -1, 0, 255, 0, 3]$

Realice un modelado utilizando redes de procesos de Kahn el algoritmo de **encodificación RZE** (mostrado arriba), es decir que ante una entrada del algoritmo mostrado arriba (in), produzca en arreglo de salida correspondiente (out). Incluya en su modelo una representación gráfica (5 puntos), una descripción de cada proceso (5 puntos) y una representación ejecutable descrita en Python (adjuntar archivo .py)completamente funcional del algoritmo.(5 puntos)