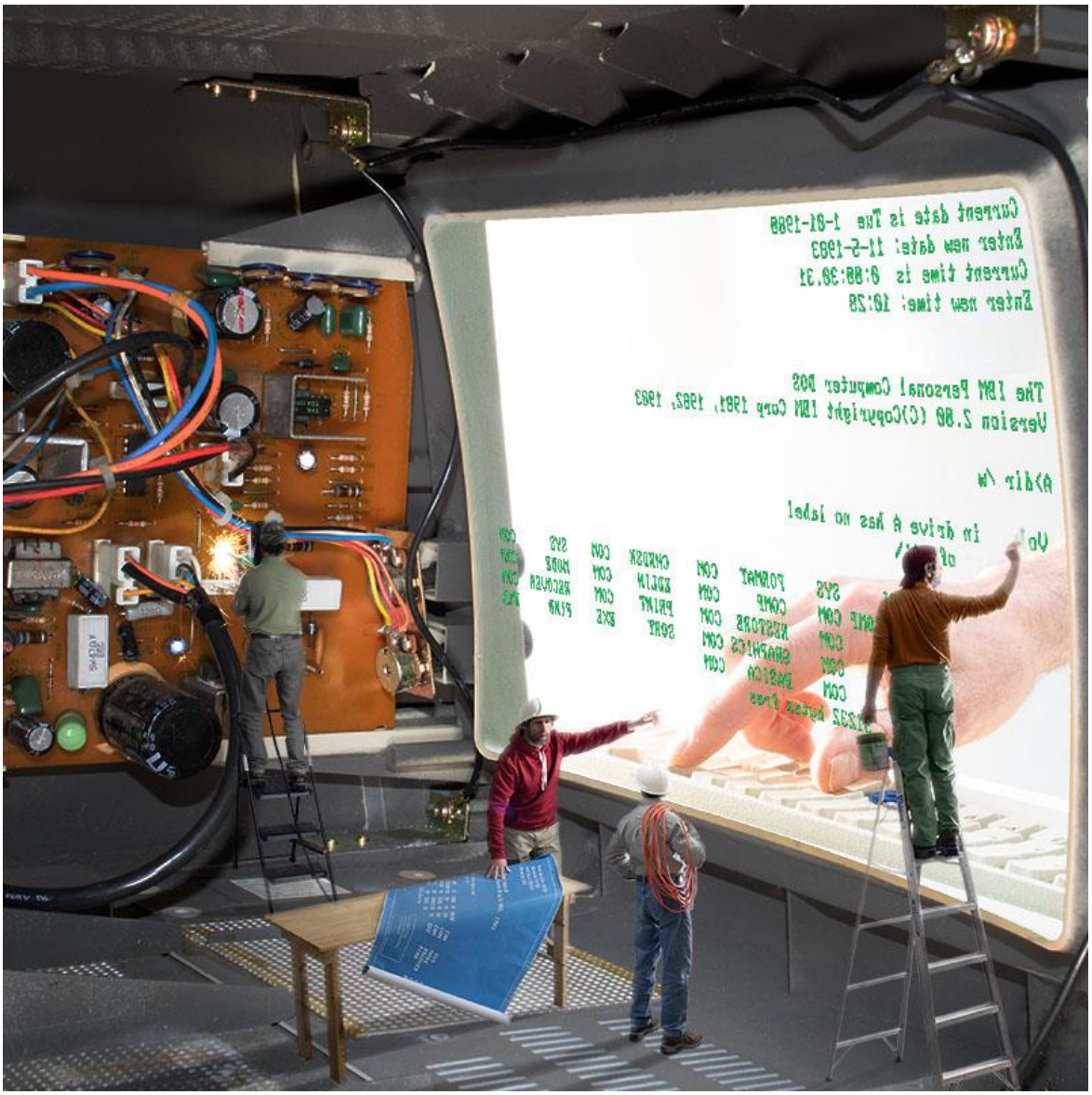

ARQUITECTURA DE COMPUTADORAS

GRUPO 4

PROYECTO 1

DESCRIPCIÓN GENERAL

Escribir un programa ensamblador que, para cualquier número primo menor o igual a 32 749, encuentre todas sus raíces primitivas.

OBJETIVO DIDÁCTICO GENERAL

Poner a prueba, en un algoritmo no trivial, la comprensión del estudiante de los mecanismos de control de programación en su implementación en lenguaje ensamblador (bifurcación condicional, ciclos y llamadas a módulos). Adicionalmente, se introduce a los estudiantes en un concepto de teoría de números que es muy útil en otras áreas de la computación, por ejemplo, en criptografía.

UN PAR DE PUNTOS TEÓRICOS

Definición: sea p un número primo, se dice que r es una *raíz primitiva* de p si para todo $0 < n < p$ existe un entero a tal que $r^a \bmod p = n$.

Lema: sean a, b, n enteros, entonces $(ab) \bmod n = ((a \bmod n)(b \bmod n)) \bmod n$

Este lema es especialmente útil para calcular los módulos de potencias enteras grandes de un número.

DESCRIPCIÓN DETALLADA

El proyecto consiste en diseñar y escribir un programa 80x86-64 que reciba un número primo del usuario y encuentre sus raíces primitivas. El programa no necesita verificar si el entero dado es primo. Se puede utilizar cualquier algoritmo para encontrar las raíces, pero el más directo, aunque un poco ineficiente quizá, es el siguiente:

> Dado p, tome uno por uno los números $1 < n < p$, y calcule, para $0 < a < p$, los términos $n^a \bmod p$. Si en el proceso alguno de los números se repite entonces ese n se descarta como raíz primitiva.

Es obligatorio el uso de llamadas a módulos, con sus correspondientes stack frames.

Se puede usar un vector para almacenar las raíces primitivas que se vaya encontrando, o una lista dinámica.

Ya sea para usar el vector o la lista, deberán hacer uso de parámetros por referencia, esto es, no se permitirá el uso directo de la estructura.

Las raíces primitivas encontradas se mostrarán en la ventana de comandos, separadas cada una por un espacio en blanco.

EVALUACIÓN

Un programa que no ensamble o que no corra recibirá automáticamente un cero de calificación.

Resultados correctos (5 pruebas)..... 30%

Uso de stack frames 25%

Uso de parámetros por referencia 25%

Entrada/Salida mediante procedimientos externos 20%

Fecha de entrega: 18 de noviembre de 2025

El trabajo será en equipos de hasta 4 personas.