

Universidad de Guadalajara

Centro Universitario de Ciencias Exactas E Ingenierías
Ingeniería Informática



Código Python del Decodificador

Nombre del alumno(a)s:

López Hernández Emiliano Juan

Huerta Romo Adolfo

Materia:

Arquitectura de computadoras, 2025B

Profesor(a):

Lopez Arce Delgado Jorge Ernesto

ANÁLISIS DEL ALGORITMO IMPLEMENTADO

PROBLEMA RESUELTO

Problema del Cambio de Monedas:

- Dada una cantidad en centavos, calcular el número mínimo de monedas necesarias
- Denominaciones utilizadas: [25, 10, 5, 1] centavos

ESTRATEGIA VORAZ (GREEDY)

- En cada paso se elige la moneda más grande posible
- Decisiones locales óptimas sin considerar el futuro
- Siempre toma el máximo valor disponible

EJEMPLO DE EJECUCIÓN

Entrada: 47 centavos

Proceso:

1. $47 \geq 25 \rightarrow$ Sí \rightarrow Usa 1 moneda de 25 \rightarrow Quedan 22
2. $22 \geq 25 \rightarrow$ No
3. $22 \geq 10 \rightarrow$ Sí \rightarrow Usa 1 moneda de 10 \rightarrow Quedan 12
4. $12 \geq 10 \rightarrow$ Sí \rightarrow Usa 1 moneda de 10 \rightarrow Quedan 2
5. $2 \geq 10 \rightarrow$ No
6. $2 \geq 5 \rightarrow$ No
7. $2 \geq 1 \rightarrow$ Sí \rightarrow Usa 1 moneda de 1 \rightarrow Queda 1
8. $1 \geq 1 \rightarrow$ Sí \rightarrow Usa 1 moneda de 1 \rightarrow Queda 0

Salida: 5 monedas totales

- 1 moneda de 25¢
- 2 monedas de 10¢
- 0 monedas de 5¢
- 2 monedas de 1¢

ARQUITECTURA MIPS32

REGISTROS UTILIZADOS

- **\$18:** Cantidad de moneda inicial (47 centavos en el ejemplo)

CARACTERÍSTICAS TÉCNICAS

- **Longitud de instrucción:** 32 bits
- **Total de instrucciones generadas:** 42 instrucciones
- **Formato soportado:** Binario directo para carga en memoria

EVIDENCIA DE FUNCIONAMIENTO

ENTRADA DEL SISTEMA

El sistema recibe código ensamblador MIPS que implementa el algoritmo voraz de cambio de monedas.

SALIDA GENERADA

- Código binario de 32 bits por instrucción
- Formato compatible con memoria de instrucciones del procesador MIPS32
- Archivo de texto con todas las instrucciones en binario

CARACTERÍSTICAS DEL DECODIFICADOR

1. Soporte para múltiples formatos:
 - R-type (registro)
 - I-type (inmediato)
 - J-type (salto)
2. Instrucciones soportadas:
 - Operaciones aritméticas y lógicas
 - Accesos a memoria (lw, sw)
 - Saltos condicionales e incondicionales
3. Manejo de etiquetas para direccionamiento simbólico

ESTRUCTURA DEL PROYECTO

COMPONENTES PRINCIPALES

1. Algoritmo voraz para cambio de monedas
2. Decodificador MIPS32 de ensamblador a binario
3. Generador de código para memoria de instrucciones
4. Interfaz para carga y visualización de resultados

ARCHIVOS GENERADOS

instrucciones.txt: Contiene todas las instrucciones en formato binario

Código binario listo para ser cargado en la memoria del procesador MIPS

CONCLUSIÓN

El proyecto implementa exitosamente un decodificador MIPS32 que transforma código ensamblador en código binario ejecutable, aplicado específicamente al problema del cambio de monedas usando una estrategia voraz. El sistema genera 42 instrucciones en formato binario de 32 bits, demostrando el correcto funcionamiento del algoritmo y la conversión a lenguaje máquina.

CAPTURAS DE SU FUNCIONAMIENTO

