



UNIVERSIDAD  
DE SANTIAGO  
DE CHILE

# Máquinas de Registros

Clase 06

Métodos de Programación  
2-2021



# CONTENIDO

Introducción



Máquinas de Registros



Ejercicios



# INTRODUCCIÓN [1/2]

Introducción

## Objetivo:

Buscar un modelo simple de ejecución.

Máquinas de  
Registros

## ¿Por qué?

Para comprender la naturaleza del cómputo y cómo se implementa en la realidad.

Ejercicios

## Contexto

Los procesadores reales se encuentran basados en máquinas de Stack y Registros.

No solo los procesadores sino que también las Máquinas Virtuales (e.g. Java, Python) que permiten portabilidad de código.

Existen otras máquinas que son más avanzadas, como las máquinas de Turing, que veremos más adelante.



# INTRODUCCIÓN [2/2]

Introducción

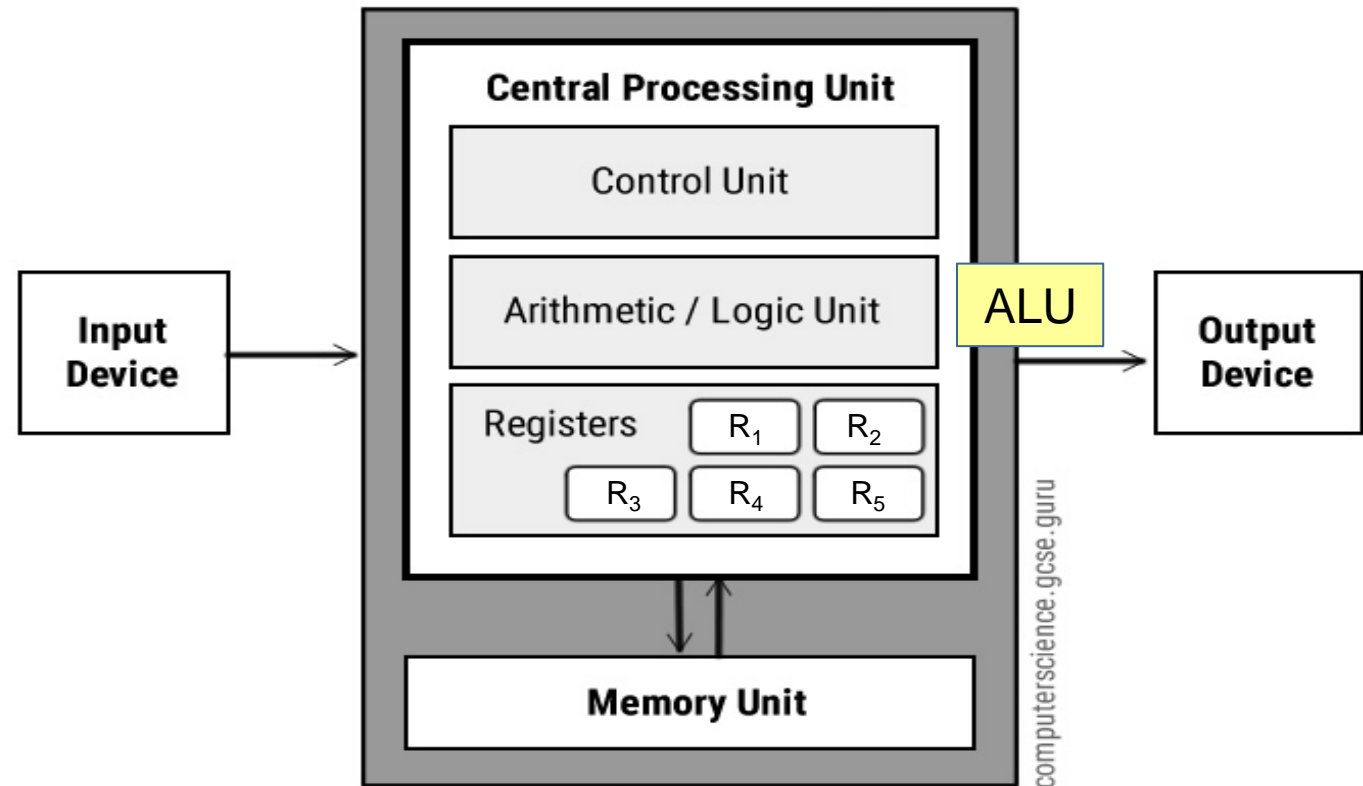
Máquinas de  
Registros

Ejercicios

## Máquinas de Registros:

Diseño abstracto de un microprocesador.

## Máquina de Von Neumann





# MÁQUINAS DE REGISTROS [1/2]

Introducción

Máquinas de  
Registros

Ejercicios

## Composición

- Un conjunto de registros (memoria estática)  $R_1, \dots, R_n$ . Cada registro  $R_i$  puede almacenar un número.
- Un programa que corresponde a un listado de instrucciones  $L_1, \dots, L_m$  etiquetadas (etiqueta:operación). En este caso  $L_i$  estaría etiquetada por  $i$ .
- Puntero ( $i$ ) a la instrucción para ejecutar.
- Instrucciones:
  - Salto a instrucciones
  - Saltos condicionales
  - Lectura/escritura de registros.



# MÁQUINAS DE REGISTROS [1/2]

Introducción

Máquinas de  
Registros

Ejercicios

## Composición

- Formalmente un conjunto de instrucciones de la máquina mínimo puede ser:
  - $R_x^+ \rightarrow i$  : Ejecuta  $R_x = R_x + 1$  y salta a la instrucción  $L_i$ .
  - $R_x^- \rightarrow i, j$  : Si  $R_x > 0$  entonces  $R_x = R_x - 1$  y salta a la instrucción  $L_i$ . En el caso  $R_x = 0$  entonces salta a la instrucción  $L_j$ .
  - HALT: Terminar



# MÁQUINAS DE REGISTROS [1/2]

Introducción

Máquinas de  
Registros

Ejercicios

## Ejemplo de ejercicio:

Sumar dos números,  $R0 = R1 + R2$

Instrucciones:

0:  $R-1 \rightarrow 1,2$

1:  $R+0 \rightarrow 0$

2:  $R-2 \rightarrow 3,4$

3:  $R+0 \rightarrow 2$

4: HALT

$R_0$	$R_1$	$R_2$	i
0	1	2	0
0	0	2	1
1	0	2	0
1	0	2	2
1	0	1	3
2	0	1	2
2	0	0	3
3	0	0	2
3	0	0	4



# MÁQUINAS DE REGISTROS [1/2]

Introducción

Máquinas de  
Registros

Ejercicios

## Ejemplo de ejercicio:

Sumar dos números,  $R0 = R1 + R2$

Instrucciones:

0:  $R-1 \rightarrow 1,2$

1:  $R+0 \rightarrow 0$

2:  $R-2 \rightarrow 3,4$

3:  $R+0 \rightarrow 2$

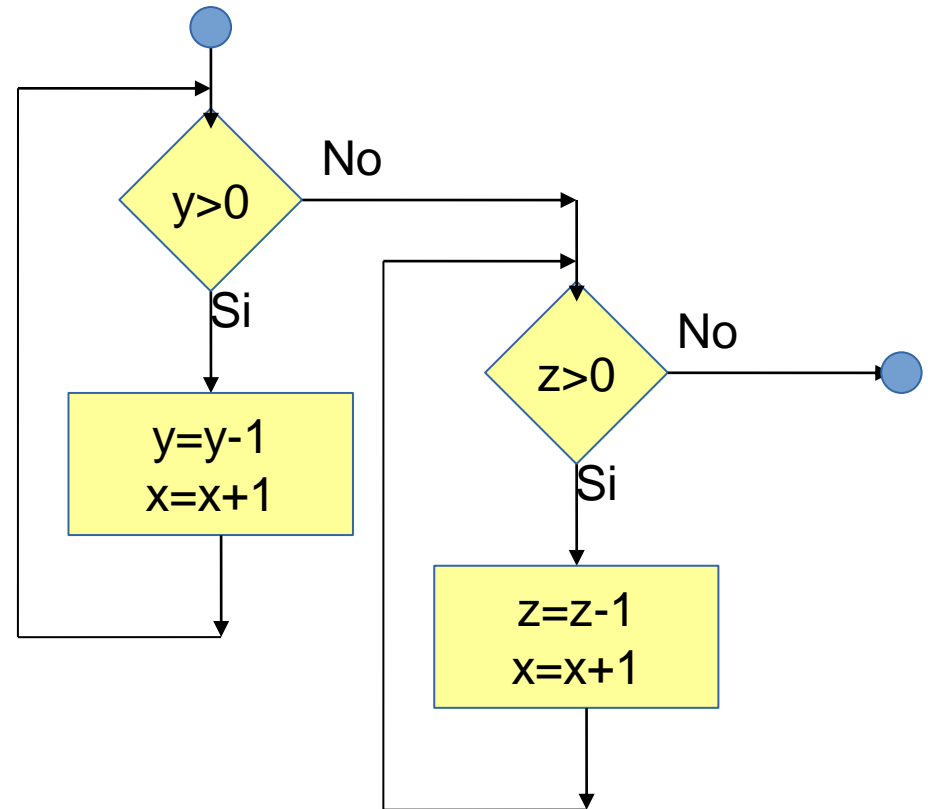
4: HALT

Considerando que:

$X = R_0$

$Y = R_1$

$Z = R_2$







# MÁQUINAS DE REGISTROS [1/2]

Introducción

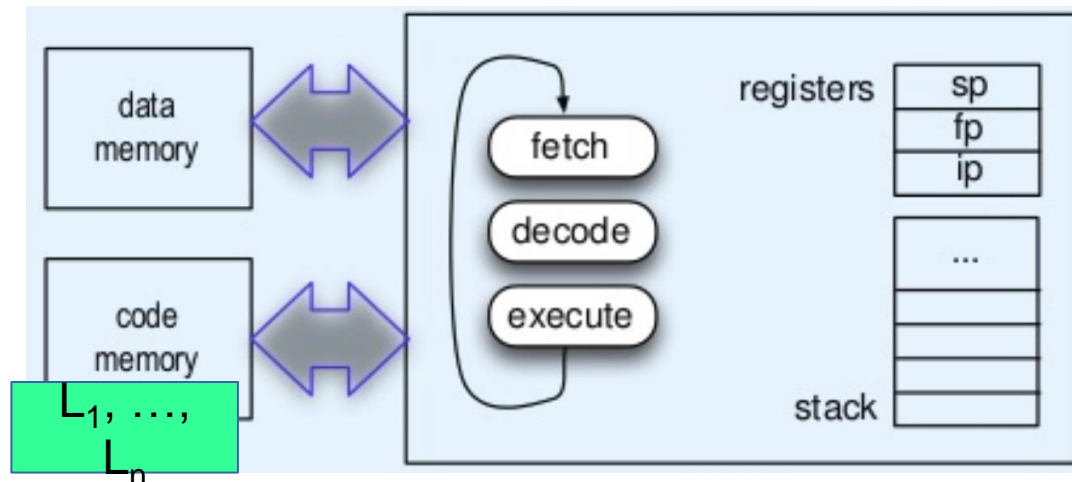
Máquinas de  
Registros

Ejercicios

## Cómo funciona:

Ciclo infinito:

- **Fetch** (buscar programa)
- **Decode** (descodificar programa)
- **Ejecutar instrucciones**: Avanza en el programa instrucción a instrucción cambiando el puntero  $i=i+1$ . En el caso de haber una instrucción de salto se modifica acorde a la instrucción.





# MÁQUINAS DE REGISTROS [1/2]

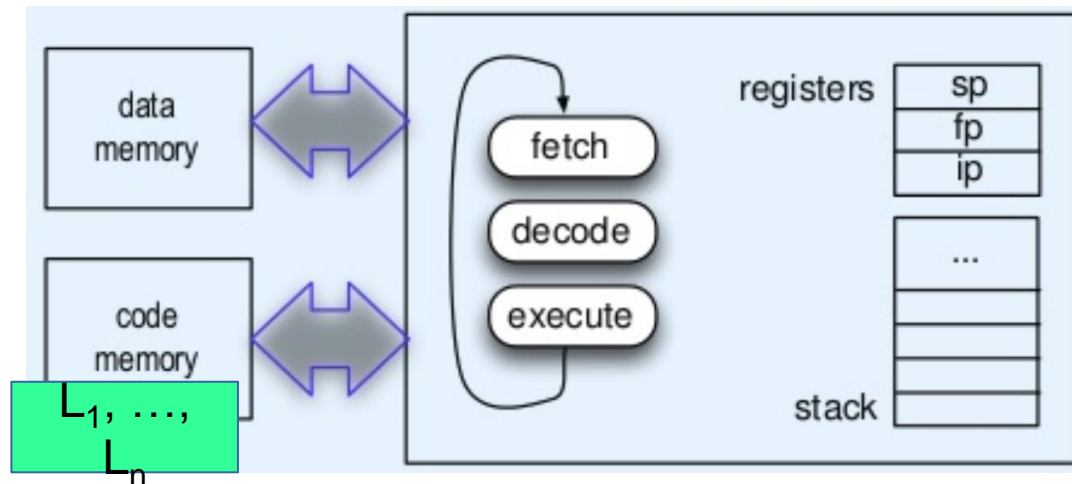
Introducción

Máquinas de  
Registros

Ejercicios

## Cómo funciona:

- Puede contener además un stack.
- Tiene acceso a memoria RAM: datos de usuario y memoria de programa.
- Emula en forma genérica un computador por lo que también se le denomina “Máquina Virtual”.





# MÁQUINAS DE REGISTROS [1/2]

Introducción

Máquinas de  
Registros

Ejercicios

## Otro conjunto de instrucciones:

**PUSH, POP:** idem máquina de stack.

**ADD:** suma en el stack

**MUL:** multiplicación en el stack

**LT:** menor que

**EQ:** igual que

**JMP <<addr>>:** saltar a la instrucción “addr”

**CJMP <<addr>>:** saltar a dirección addr si es mayor que 0.

**HALT:** terminar

Ejemplo: (<https://github.com/codyebberon/vm>)



# MÁQUINAS DE REGISTROS [1/2]

Introducción

Máquinas de  
Registros

Ejercicios

## Máquina entregada:

- Está compuesto de dos archivos:
  - Registros.c: Es el archivo principal (y el que se debe compilar). Este lee el archivo de instrucciones y realiza el cálculo de la máquina de registros.
  - Las instrucciones de la máquina de registros (Registros.in): Es un archivo de texto plano compuesto por las instrucciones básicas enseñadas. Su estructura es la siguiente:
    - La primera línea indica la cantidad de registros a utilizar.
    - La segunda línea posee la etiqueta de la primera instrucción a ejecutar seguido del valor inicial de los N registros a utilizar (la numeración de estos comienza en 0).
    - La tercera línea indica la cantidad de instrucciones (La numeración comienza en 0).
    - Las instrucciones de la máquina a implementar, donde:
      - **SUMA:**  $Lw\ R+ x \rightarrow Ly$  (w= Etiqueta Actual; x= Registro; y= Etiqueta siguiente)
      - **RESTA:**  $Lw\ R- x \rightarrow Ly\ Lz$  (w= Etiqueta Actual; x= Registro; y= Etiqueta siguiente si se logra hacer la operación; z= Etiqueta siguiente si no se logra hacer la operación)
      - **HALT:**  $Lw\ HALT$  (w= Etiqueta Actual)



# MÁQUINAS DE REGISTROS [1/2]

Introducción

Máquinas de  
Registros

Ejercicios

## Máquina entregada:

- Ejemplo de programa:

3

0 0 1 2

5

L0 R- 1 -> L1 L2

L1 R+ 0 -> L0

L2 R- 2 -> L3 L4

L3 R+ 0 -> L2

L4 HALT



# MÁQUINAS DE REGISTROS [1/2]

Introducción

Máquinas de  
Registros

Ejercicios

## Máquina entregada:

- Ejemplo de programa:

3

0 0 1 2

valores 0, 1 y 2

5

la 4)

L0 R- 1 -> L1 L2

la instrucción 2.

L1 R+ 0 -> L0

L2 R- 2 -> L3 L4

la instrucción 4.

L3 R+ 0 -> L2

L4 HALT

→ Se utilizan 3 registros

→ Se comienza en la instrucción 0, y los registros poseen

→ Se compone solo de 5 instrucciones (de la 0 a

→ La instrucción 0 resta 1 al valor guardado en el registro 1, si se puede va a la instrucción 1, en caso contrario a

→ La instrucción 1 suma un 1 al registro 0.

→ La instrucción 2 resta 1 al valor guardado en el registro 2, si se puede va a la instrucción 3, en caso contrario a

→ La instrucción 3 suma un 1 al registro 0.

→ La instrucción 4 suma un 1 al registro 0.



# EJERCICIOS PROPUESTOS

Introducción

Máquinas de  
Registros

Ejercicios

1. La resta truncada de dos números (es 0 si  $z > y$ )  $x = y - z$
1. El máximo de dos números  $x = \max(y, z)$ .
1. La multiplicación de dos números  $x = y * z$
1. La parte entera de la división de dos números  $x = \text{int}(y/z)$
1. El resto de la división de dos números.
1. La potencia y de 2,  $x = 2^y$
1. El logaritmo en base 2 de un número (0 si es 0).