


## Rúbrica de Evaluación


### Actividad 2.8 – Simulador de Instrucciones en Java

 Criterios para evaluar el desarrollo, lógica, presentación y creatividad del simulador tipo ensamblador.




## Criterios generales

| Criterio                   | Peso orientativo |
|----------------------------|------------------|
| Lógica y funcionamiento    | 40%              |
| Uso de bucles y control    | 20%              |
| Diseño modular (funciones) | 15%              |
| Claridad en la salida      | 15%              |
| Ampliación / Creatividad   | 10%              |




## Simulación de instrucciones básicas

| Nivel  | Descripción  |
|--|--|
|  Bajo     | No se ejecutan correctamente todas las instrucciones   |
| <br>Medio | Instrucciones básicas ( <code>LOAD</code> , <code>ADD</code> , <code>PRINT</code> ) funcionan  |
|  Alto     | Todas las instrucciones ( <code>LOAD</code> , <code>ADD</code> , <code>JMP</code> , <code>PRINT</code> , <code>HALT</code> ) implementadas correctamente |

## Uso de bucles y lógica de control




| Nivel  | Descripción   |
|--|---|
|  Bajo     | No se usan bucles o hay errores lógicos graves  |
| <br>Medio | Se usa un bucle funcional pero con lógica limitada  |
|  Alto    | Bucle correctamente implementado con control de flujo (ej. <code>JMP</code> ) bien gestionado |

## Estructura y modularidad del código




| Nivel   | Descripción  |
|---|--|
|  Bajo  | Todo el código está en <code>main</code> , sin separación lógica |
|  Medio | Algunas partes modularizadas en funciones simples                |
|  Alto  | Uso coherente de funciones con nombres claros y reutilización    |






## Presentación y salida del programa

| Nivel   | Descripción  |
|---|--|
|  Bajo  | La salida es confusa o no muestra resultados esperados                         |
|  Medio | La salida funciona pero es poco clara  |
|  Alto  | La salida es clara, estructurada y facilita la lectura del flujo del simulador |

## Tabla de ejecución




| Nivel  | Descripción  |
|--|--|
|  Bajo     | No se incluye ninguna tabla ni análisis de ejecución   |
| <br>Medio | Tabla incompleta o con errores   |
|  Alto    | Tabla bien estructurada con <code>PC</code> , instrucción y valores del acumulador antes y después |

## Extensión del lenguaje (opcional)

| Nivel  | Descripción   |
|--|---|
|  Bajo     | Solo las instrucciones básicas  |
| <br>Medio | Añade una instrucción extra (ej. <code>SUB</code> , <code>IFZERO</code> )   |
|  Alto    | Añade varias instrucciones nuevas bien integradas y documentadas<br>( <code>LOADM</code> , <code>STORE</code> , etc.) |






## Gestión de errores y control

| Nivel  | Descripción   |
|--|---|
|  Bajo     | El programa falla ante entradas incorrectas ( <code>JMP</code> fuera de rango)            |
| <br>Medio | Algunos errores están gestionados con controles básicos                                   |
|  Alto    | Todos los errores esperables están controlados y el programa responde con mensajes claros |



## Creatividad y ampliación

| Nivel  | Descripción  |
|--|--|
|  Bajo     | Sin elementos adicionales  |
| <br>Medio | Añade alguna mejora (menú, interacción, visualización)                                     |
|  Alto    | Ampliaciones relevantes: memoria simulada, GUI, animaciones, editor de instrucciones, etc. |



## Reflexión o autoevaluación (opcional)

- ✓ ¿Has probado tu programa con distintos conjuntos de instrucciones?
- ✓ ¿Qué errores has encontrado y corregido?
- ✓ ¿Qué añadirías si tuvieras más tiempo?



Esta parte no es obligatoria, pero da valor añadido al trabajo entregado.

## Conclusión

Un simulador completo debe demostrar:

- ✓ Ejecución correcta y lógica de instrucciones
- ✓ Estructura clara y modular
- ✓ Uso adecuado de bucles y estructuras de control
- ✓ Capacidad de representar paso a paso lo que ocurre en la "CPU simulada"

 ¡Evalúate, revisa y entrega tu mejor versión!