# 

Asignatura:

**ESTRUCTURA DE DATOS**

**NRC: 29902**

Proyecto Grupal:

**Sistema de control de tareas y gestión de software**

INTEGRANTES:

* Collazos Gracía Jose
* Escobar Ruiz Kevin
* Solis Hinostroza Cynthia Rossy
* Oroya Molina Luis Fernando

**PERÚ**

**2025**

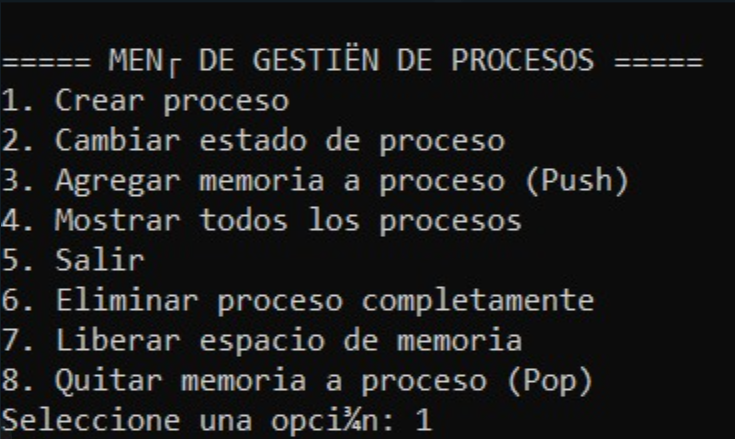
Manual del Sistema de Gestión de Procesos en C++

# 1. Introducción

Este manual describe el funcionamiento detallado de un sistema de gestión de procesos desarrollado en lenguaje C++. El sistema simula un entorno donde múltiples procesos pueden crearse, gestionarse y eliminarse, incluyendo el uso de memoria por parte de cada uno. Este sistema puede ser útil en ambientes académicos o de simulación para entender cómo operan los procesos dentro de un sistema operativo.

# 2. Estructura del Sistema

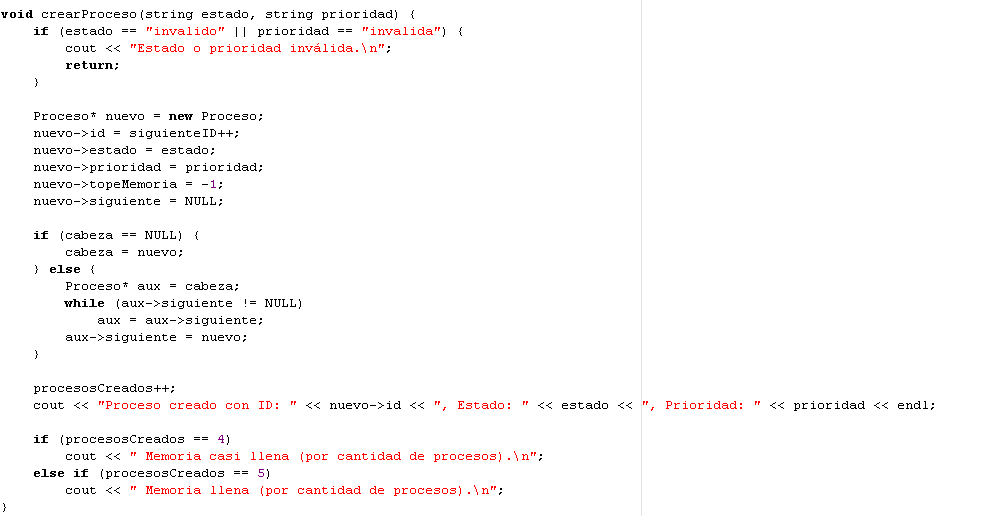
El sistema utiliza una estructura de lista enlazada para almacenar los procesos. Cada proceso es una instancia de la estructura `Proceso`, que contiene un identificador único, estado, prioridad, una pila de memoria (simulada mediante un arreglo), y un puntero al siguiente proceso en la lista.



# 3. Funcionalidades del Sistema

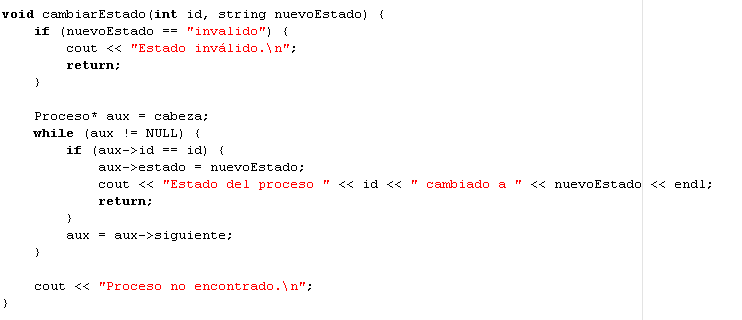
## 3.1 Crear proceso

* Permite al usuario crear un nuevo proceso, asignándole un estado (listo, ejecutando, bloqueado) y una prioridad (alta, media, baja). Cada nuevo proceso recibe automáticamente un ID único. El sistema alerta cuando se han creado 4 o 5 procesos para indicar saturación.



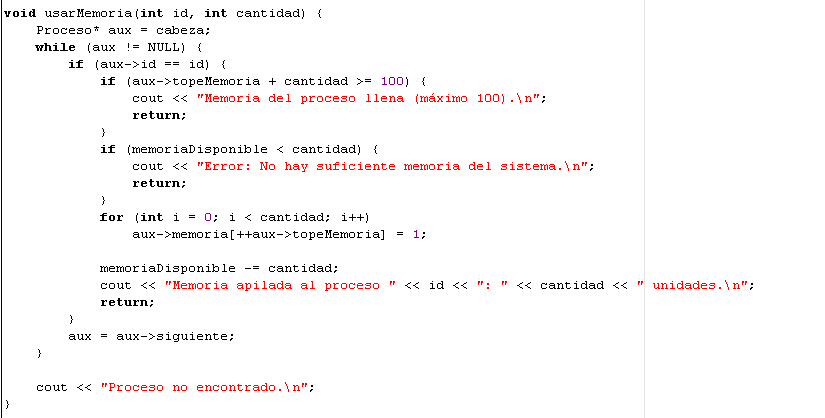
## 3.2 Cambiar estado del proceso

* Permite modificar el estado de un proceso existente, identificado por su ID. Es útil para simular cambios de contexto en sistemas operativos.



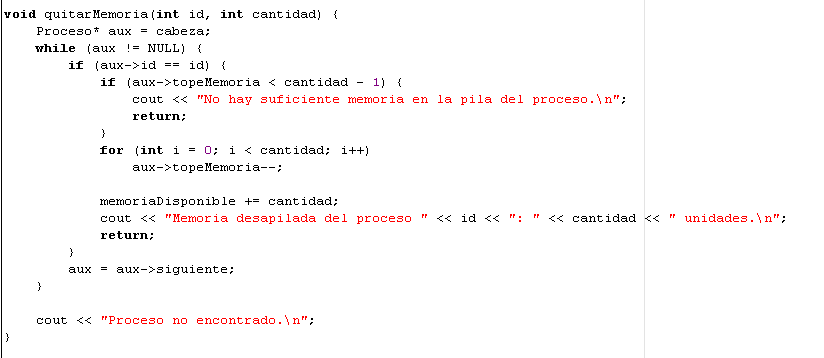
## 3.3 Agregar memoria (Push)

* Simula la asignación de memoria al proceso usando una estructura de pila. Se verifica que la cantidad de memoria asignada no exceda el límite del proceso ni del sistema.



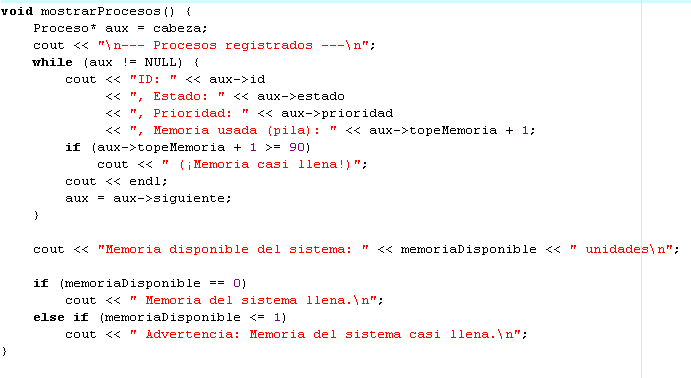
## 3.4 Quitar memoria (Pop)

* Permite reducir la memoria usada por un proceso, simulando el desapilamiento. Se verifica que la memoria a liberar no exceda la actualmente utilizada.



## 3.5 Mostrar procesos

* Muestra todos los procesos registrados, incluyendo su ID, estado, prioridad y memoria usada. También indica cuánta memoria del sistema queda disponible.



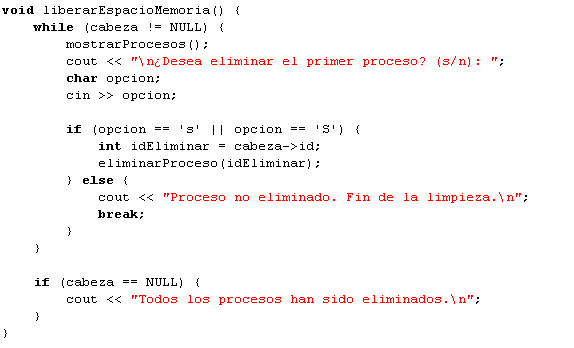
## 3.6 Eliminar proceso

* Elimina un proceso completamente, liberando toda su memoria y actualizando el estado del sistema. Es útil para mantener la eficiencia del sistema.



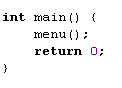
## 3.7 Liberar espacio de memoria

* Revisa uno a uno los procesos registrados y da la opción de eliminarlos para liberar memoria del sistema. Es útil en escenarios de emergencia donde la memoria está casi o totalmente ocupada.



## 3.8 Salida del sistema

Finaliza el programa, permitiendo al usuario cerrar la simulación de manera segura.



# 4. Recomendaciones de Uso

 **Utilizar estados y prioridades válidas al crear procesos:**  
Para asegurar que el sistema funcione correctamente, es fundamental que los usuarios ingresen únicamente los estados permitidos (listo, ejecutando, bloqueado) y las prioridades definidas (alta, media, baja). Esto evita errores en la gestión y facilita el seguimiento de los procesos.

 **Supervisar constantemente la memoria disponible del sistema:**  
El sistema tiene un límite total de memoria que puede asignar a todos los procesos. Es importante monitorear este recurso para evitar fallos o bloqueos. La aplicación muestra alertas cuando la memoria se acerca a su límite, lo que ayuda a tomar decisiones oportunas.

 **Evitar saturar la pila de memoria en cada proceso:**  
Cada proceso tiene una pila de memoria limitada (hasta 100 unidades en este caso). Se debe evitar apilar demasiada memoria para no provocar errores o pérdida de datos. La función usarMemoria previene asignaciones que excedan este límite, pero el usuario debe ser consciente de no sobrecargar los procesos.

 **Eliminar procesos inactivos o innecesarios para liberar memoria:**  
Cuando un proceso ya no es necesario o está inactivo, es recomendable eliminarlo para liberar recursos. Esto permite que otros procesos tengan acceso a la memoria disponible y mejora la eficiencia general del sistema.

 **Usar la funcionalidad de liberación de espacio cuando se aproxime al límite del sistema:**  
El sistema incluye una opción para liberar espacio de memoria eliminando procesos desde el inicio de la lista. Esta herramienta es útil para evitar que la memoria se llene completamente y se produzcan errores inesperados o ralentizaciones.

 **Realizar un monitoreo periódico de los estados de los procesos:**  
Cambiar el estado de un proceso puede simular cambios reales en sistemas operativos, como pasar de “listo” a “ejecutando”. Esto ayuda a gestionar mejor la ejecución y prioridad de las tareas en un entorno multitarea.

 **Validar entradas del usuario para evitar errores:**  
Es importante que los usuarios introduzcan datos correctos, ya que el sistema depende de la integridad de esta información para funcionar adecuadamente. En caso de entradas incorrectas, el sistema ofrece mensajes de error y solicita corrección.

# 5. Conclusión

El sistema desarrollado ofrece una plataforma básica pero robusta para simular la gestión de procesos y la administración de memoria en un sistema operativo. A través de una estructura modular basada en listas enlazadas y el uso de pilas para la asignación de memoria, se logra un modelo funcional que refleja aspectos esenciales de la gestión real de procesos.

Gracias a este enfoque, el sistema puede ser utilizado como una herramienta educativa que ayuda a comprender conceptos clave como la creación y eliminación de procesos, la asignación dinámica de memoria, y la gestión de estados y prioridades. Además, el diseño abierto y claro facilita futuras mejoras, como la implementación de algoritmos de planificación de procesos, manejo de interrupciones, y soporte para diferentes tipos de memoria.

Este proyecto también resalta la importancia de mantener un control constante sobre los recursos del sistema para evitar sobrecargas y garantizar un funcionamiento eficiente. La experiencia obtenida puede servir como base para desarrollar sistemas más complejos y realistas en ambientes profesionales o académicos.

En resumen, esta simulación no solo cumple su función educativa, sino que sienta las bases para entender mejor cómo los sistemas operativos manejan los procesos y la memoria, preparándonos para retos tecnológicos más avanzados en el futuro.