

PONTIFICIA UNIVERSIDAD CATÓLICA DEL PERÚ
FACULTAD DE CIENCIAS E INGENIERÍA

ALGORITMIA Y ESTRUCTURA DE DATOS
2da. práctica (tipo B)
(Segundo Semestre 2025)

Duración: 1h 50 min.

- **No puede utilizar apuntes, solo hojas sueltas en blanco.**
- En cada función el alumno deberá incluir, a modo de comentario, la forma de solución que utiliza para resolver el problema. De no incluirse dicho comentario, el alumno perderá el derecho a reclamo en esa pregunta.
- No puede emplear plantillas o funciones no vistas en los cursos de programación de la especialidad.
- Los programas deben ser desarrollados en el lenguaje C++. Si la implementación es diferente a la estrategia indicada o no la incluye, la pregunta no será corregida.
- Un programa que no muestre resultados coherentes y/o útiles será corregido sobre el 50% del puntaje asignado a dicha pregunta.
- Debe utilizar comentarios para explicar la lógica seguida en el programa elaborado. El orden será parte de la evaluación.
- **Solo está permitido acceder a la plataforma de PAIDEIA, cualquier tipo de navegación, búsqueda o uso de herramientas de comunicación se considera plagio por tal motivo se anulará la evaluación y se procederá con las medidas disciplinarias dispuestas por la FCI.**
- Para esta evaluación solo se permite el uso de las librerías `iostream`, `iostream`, `climits`, `cstring`, `cmath` o `fstream`
- Su trabajo deberá ser subido a PAIDEIA.
- **Es obligatorio usar como compilador CLion.**
- Los archivos deben llevar como nombre su código de la siguiente forma `codigo_LAB2_P#&` (donde # representa el número de la pregunta a resolver y & la alternativa)

Pregunta 1 (10 puntos)

La gerencia deportiva de un equipo de futbol realiza la contratación de jugadores para la temporada 2026 y cada jugador contratado los ingresa a una lista simplemente enlazada. Los datos que guarda un jugador son: Número de Camiseta, Nombre, Posición (por ejemplo: portero, mediocampo, delantero, defensa, etc.).

Ahora, el club, contrata al director técnico, el cuál viene con varias formaciones que desea probar dentro del equipo. Estas formaciones involucran el orden de las posiciones de los jugadores dentro del campo. Por ejemplo, el entrenador podría formar a los jugadores de la siguiente manera: Portero, defensa, mediocampo, delantero.

La gerencia deportiva, lo contrata a Ud. para que pueda implementar un programa que reorganice a los jugadores, dentro de la lista original, siguiendo el orden de las posiciones brindadas por el entrenador. En caso existan varios jugadores dentro de la misma posición, debe colocarlos en el orden que fueron contratados.

Por ejemplo, si el equipo solo contratará jugadores de las posiciones: Portero, MedioCampo, Defensa y Delantero, una posible entrada podría ser:

```
(7, "Martínez", "Delantero")
(1, "Ramírez", "Portero")
(5, "Pérez", "Defensa")
(8, "Torres", "Mediocampo")
(9, "López", "Delantero")
(3, "Gómez", "Defensa")
```

Y la formación que usa el entrenador es: Portero, Defensa, Mediocampo y Delantero.

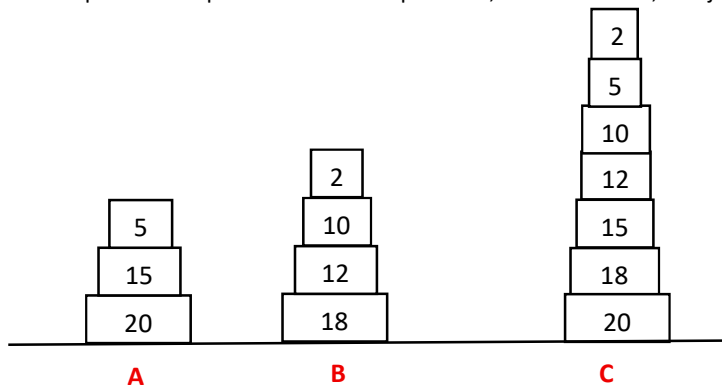
Entonces la salida sería:

(1, "Ramírez", "Portero")
(5, "Pérez", "Defensa")
(3, "Gómez", "Defensa")
(8, "Torres", "Mediocampo")
(7, "Martínez", "Delantero")
(9, "López", "Delantero")

- Se le pide definir las estructuras necesarias y el ingreso de datos para la solución del problema. Recuerde que las posiciones pueden variar y el orden en la formación también. (2 puntos).
- Se le pide implementar una función **reordenaFormacion** que permita reordenar la lista de jugadores en base al orden definido por el entrenador. Para este reordenamiento no puede utilizar ningún TAD adicional ni memoria adicional, ni arreglos auxiliares. (8 puntos).

Pregunta 2 (10 puntos)

En empresa de distribución de productos adquiere un robot apilador 2025, especializado en mover paquetes y apilarlos considerando su fragilidad, en otras palabras, sin colocar un paquete de menor peso bajo un paquete de mayor peso, ya que se rompe. El robot realiza la operación de colocar los paquetes de una pila a otra empleando como máximo una pila auxiliar. Luego de un tiempo nace una nueva necesidad la cual es fusionar dos pilas en una tercera pila en otras palabras mover los paquetes de una pila A y B colocándolos en una pila C. Debido a la tecnología base del robot solo puede emplear como máximo una pila auxiliar para realizar esta operación, a continuación, un ejemplo:



Se le pide:

Desarrolle un algoritmo en C++ que fusione las pilas A y B en una pila C considerando las restricciones de fragilidad, empleando como máximo una pila auxiliar. Recuerde que no puede utilizar ninguna estructura o TAD adicional. De no cumplir estas restricciones su respuesta no tendrá valor.

Al finalizar el laboratorio, comprima la carpeta de su proyecto empleando el programa Zip que viene por defecto en el Windows, **no se aceptarán los trabajos compactados con otros programas como RAR, WinRAR, 7zip o similares**. Luego súbalo a la tarea programa en Paideia para este laboratorio.

Profesores del curso:

Ana Roncal
Fernando Huamán
David Allasi
Rony Cueva
Erasmus Gomez

San Miguel, 11 de octubre del 2025