

**PONTIFICIA UNIVERSIDAD CATÓLICA DEL PERÚ**  
**FACULTAD DE CIENCIAS E INGENIERÍA**

**ALGORITMIA**

**Primer Examen**

**(Segundo Semestre 2023)**

Duración: 2h 50 min.

- En cada función el alumno deberá incluir, a modo de comentario, la estrategia o 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 STL, Plantillas o funciones no vistas en los cursos 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.
- Se utilizarán herramientas para la detección de plagios, por tal motivo si se encuentran soluciones similares, se anulará la evaluación a todos los implicados y se procederá con las medidas disciplinarias dispuestas por la FCI.
- **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.**
- **Los programas deben ser desarrollados utilizando nombres para las funciones y variables en español, al igual que los comentarios. El uso de otro idioma anula su respuesta.**
- Para esta evaluación solo se permite el uso de las librerías **iostream, iomanip, climits cmath, fstream y cstring**
- Su trabajo deberá ser subido a PAIDEIA.
- **Es obligatorio usar como compilador NetBeans. Solo pueden usar una biblioteca para cada tipo de TAD, cualquier duplicidad o unión de bibliotecas invalidará su respuesta.**
- Los archivos deben llevar como nombre su código de la siguiente forma `codigo_EX1_P#` (donde # representa el número de la pregunta a resolver)

---

**Resuelva solo 2 de las siguientes preguntas:**

**Pregunta 1 (10 puntos)**

El maestro Splinter les ha dado una misión importante a las 4 Tortuninjas (Leonardo, Rafael, Donatello y Miguel Ángel). La misión consiste en que deben salvar unos camiones que llevan artefactos pequeños pero muy peligrosos que podrían atentar contra la vida en Nueva York. La cantidad de camiones es variable y solos se sabe que son más de 4 y menos que 9. Además, la cantidad de artefactos que lleva cada camión también es variable, así como el peso de cada artefacto.

Cada camión debe ser manejado por una Lista, por lo que los artefactos de cada camión serán nodos de esta lista. Los artefactos se deben ir ingresando al final de la lista de cada camión. Cada TortuNinja tiene una capacidad máxima de peso que puede cargar.

La misión consiste en que las 4 tortuNinjas deben salvar a **todos** los camiones porque si algún artefacto se pierde, la ciudad de Nueva York será destruida. Se sabe que todas las TortuNinjas deben rescatar **si o si al menos un camión y que una o más tortuNinjas podrían rescatar como máximo 2 camiones, siempre y cuando la suma de los pesos de todos los artefactos**

**que se encuentran en dichos camiones no supere su capacidad máxima de peso que puede cargar.**

Si alguna TortuNinja debe salvar a más de un camión, debe unir los artefactos de ambos camiones en uno solo. Para realizar la unión debe ordenar los pesos de los artefactos de uno de los camiones de menor a mayor y el otro de mayor a menor e intercalar los artefactos de ambos camiones en el camión ordenado de menor a mayor.

Al finalizar, debe imprimir los artefactos del camión (unificado en algunos casos) que salvará cada TortuNinja.

Por ejemplo:

Ingrese la cantidad de camiones: 5

Ingrese la capacidad máxima de carga de Leonardo: 34

Ingrese la capacidad máxima de carga de Rafael: 24

Ingrese la capacidad máxima de carga de Donatelo: 12

Ingrese la capacidad máxima de carga de Miguel Ángel: 30

**Ingrese los pesos del Bus Nro. 1: 8 9 0**

Ingrese los pesos del Bus Nro. 2: 15 9 0

Ingrese los pesos del Bus Nro. 3: 3 6 0

Ingrese los pesos del Bus Nro. 4: 1 6 8 4 6 0

**Ingrese los pesos del Bus Nro. 5: 5 6 3 0**

Resultado:

Las TortuNinjas completaron su misión.

Los camiones a salvar son:

Leonardo: 8 6 9 5 3

Rafael: 15 9

Donatelo: 3 6

Miguel Ángel: 1 6 8 4 6 0

Como se puede apreciar, Leonardo salvó a los camiones 1 y 5, por lo tanto, estos camiones se fusionaron, de acuerdo con lo descrito en el caso y dio como resultado el camión: 8 6 9 5 3, porque el camión 1 se mantuvo así porque ya está ordenado de menor a mayor y el camión 5 se ordenó de mayor a menor antes de fusionar.

- Implemente las estructuras necesarias, así como el ingreso de datos, para soportar los camiones y artefactos de acuerdo con lo descrito en el enunciado. Recuerde que la cantidad de camiones y los pesos de cada artefacto es variable. También recuerde que la capacidad máxima de carga de cada TortuNinja es variable. Si asume una cantidad fija de camiones en el ingreso de datos no se corregirá toda la pregunta. (2 puntos).
- Implemente una función que permita determinar si existe o no una posible solución para que las TortuNinjas puedan terminar exitosamente la misión encargada por el maestro Splinter. (3 puntos).
- Implemente una función que permita imprimir la lista final del camión que cada TortuNinja salvará. Debe tener en cuenta que, si una TortuNinja va a salvar a más de un camión, debe pasar todos los artefactos a solo uno de los camiones que salvará, siguiendo lo descrito en el enunciado, y debe imprimir solo ese camión final. Si desea ordenar alguna lista, debe utilizar el método de la burbuja para dicha acción, si no lo utiliza, se le colocará 0 en todo el punto c. Para esta función no puede utilizar memoria extra por ningún motivo, tampoco TADS adicionales. Si utiliza alguno de ellos se invalidará toda la pregunta. (5 puntos)

## Pregunta 2 (10 puntos)

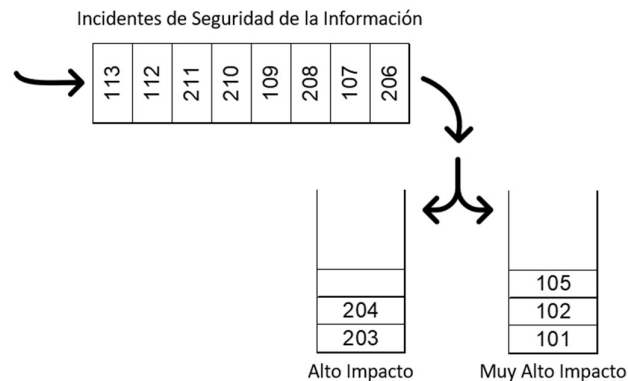
La Gestión de Incidentes Seguridad de la Información es un proceso que se enfoca en identificar, atender, responder y manejar los incidentes de seguridad que ocurren en una empresa. Uno de los factores críticos en la gestión de incidentes es el tiempo de atención, por lo que la premisa principal es: “resolver el incidente en el menor tiempo posible”. Debido al incremento de incidentes de seguridad de la información a nivel mundial, la empresa ha establecido replantear su metodología para gestionar la atención a los incidentes de seguridad. Un extracto de la metodología es el siguiente:

1. El incidente es registrado con un código de 3 dígitos:

**2 0 3**

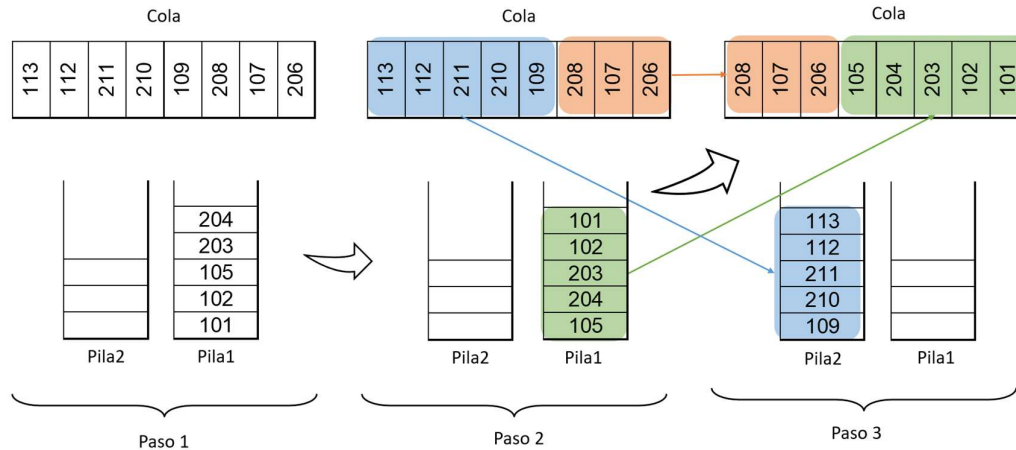
1 – Muy alto impacto	01 – Incidente N°01
2 – Alto impacto	02 – Incidente N°02
...	...
	99 – Incidente N°99

2. Al inicio de la jornada, el registro de los incidentes se realiza en **una cola** y luego pasan a **dos pilas** para ser atendidas (una para las de Alto Impacto y otra para las de Muy Alto Impacto)



Nótese que la **Cola** tiene una cantidad **n espacios** definidos. Las Pilas no tienen espacios limitados.

3. Cuando una de las pilas contiene 3 incidentes, se **reinicia la cola** de la siguiente manera:



Paso 1: Se pasan los nodos de Pila2 a Pila1.

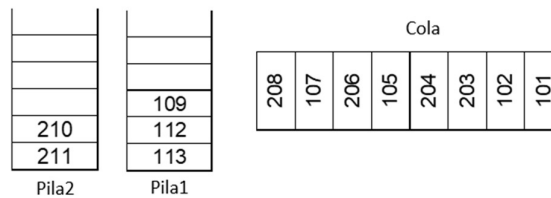
Paso 2: Se ordena Pila1.

Paso 3:

- **En azul:** Se desencola y encola  $\text{longitud Cola} - \text{longitud Pila1}$  espacios, seguidamente se desencola  $\text{longitud Pila1}$  espacios y se apilan en Pila1.
- **En verde:** Se encola  $\text{longitud Pila1}$  espacios desde Pila1.
- **En anaranjado:** Se desencola y encola  $\text{longitud Cola} - \text{longitud Pila1}$  espacios.

En términos prácticos, se busca volver a tener los incidentes según orden de llegada, pero como la cantidad excede la capacidad de la cola, se utiliza una de las pilas como “espacios” temporales para los incidentes restantes.

4. Finalmente, se **reclasifican las pilas** según el nivel de impacto de los incidentes:



Como se observa las pilas están ordenadas por código, listas para ser desapiladas y entrar a la cola una vez se vayan liberando espacios.

Se le solicita que implemente un algoritmo que permita simular el extracto de la metodología presentada que tiene la empresa para gestionar la atención a los incidentes de seguridad de la información:

- Desarrollar una función **reinicia\_cola()** que satisfaga TODO lo que indica el punto 3 de la metodología. La función recibe como parámetros: la Cola, la cantidad espacios permitidos de la cola, la Pila1 y la Pila2. Para esta función no puede usar memoria extra, recursión u otras estructuras. (6 puntos).
- Desarrollar una función **reclasifica\_pilas()** que satisfaga TODO lo que indica el punto 4 de la metodología. La función recibe como parámetros la Pila1 y la Pila2. Puede usar una cola

auxiliar temporal. No puede usar memoria extra, recursión u otras estructuras auxiliares. Solo puede usar variables simples. (3.0 puntos).

- c) Desarrolle la función **main()** donde se declaren: la Cola, la cantidad espacios permitidos de la cola y las 2 pilas. Debe inicializar las estructuras de datos (puede usar los datos mostrados en el punto 2 de la metodología). Debe invocar a las funciones solicitadas en a) y b) así como mostrar las 3 estructuras antes y después de cada acción de la metodología. (1 punto).

Ejemplo:

```

X
INICIAL:
Cola: 206 107 208 109 210 211 112 113
Pila 1: 105 102 101
Pila 2: 204 203

REINICIA COLA:
Cola: 101 102 203 204 105 206 107 208
Pila 1: La cola está vacía
Pila 2: 113 112 211 210 109

RECLASIFICA PILAS:
Cola: 101 102 203 204 105 206 107 208
Pila 1: 210 211
Pila 2: 109 112 113
```

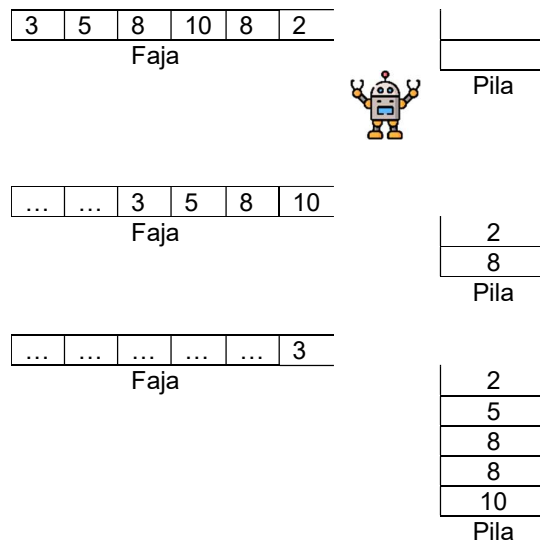
### Pregunta 3 (10 puntos)

Una empresa adquiere un nuevo robot apilador modelo HEX 1.0, esta unidad se encargará de mover los productos que vienen por una faja transportadora (opera como cola) hacia una pila donde debe acumular todos los productos, para tal tarea el robot tiene 2 brazos mecanizados, uno de ellos se encargará de recibir el producto que ha salido de la cola y el otro servirá para realizar los movimientos necesarios para mantener la pila ordenada. La pila siempre estará ordenada colocando los productos con mayor peso en la parte inferior y los productos de menor peso en la parte más alta ya que, debido a su fragilidad, si se coloca un producto con menor peso bajo un producto de mayor peso, el más ligero se puede quebrar. **Por tal motivo en ningún momento puede haber un producto de mayor peso encima de uno con menor peso, tampoco se pueden volver a encolar los productos que salen de la cola, ya que es demasiado larga.**

Aunque esta tarea puede ser difícil para cualquier robot, nuestra nueva unidad HEX 1.0, cuenta con muchas herramientas, como la posibilidad de utilizar estrategias de recursión y de iteración sin problema, para cualquier tarea que necesite.

Para realizar la tarea de mantener siempre la pila ordenada por peso, el robot puede usar 2 pilas auxiliares, donde puede colocar los productos temporalmente antes de llevarlos a la pila principal. Desde luego la restricción de fragilidad de los productos debe ser considerado tanto para la pila

principal, como para sus pilas auxiliares, **por lo que en ningún momento puede haber un producto de mayor peso sobre un producto de menor peso**. A continuación, mostramos un ejemplo de una línea y como los productos se van acumulando en la pila:



Los valores en cada nodo de la cola y la pila estarán dados solo por el peso. Recuerde que se tratan de objetos reales, por tal motivo no puede duplicar nodos o usar estructuras adicionales no indicadas en el proceso.

- a) Implemente un programa para el robot apilador, que sirva para colocar los elementos de la faja transportadora en la pila principal, considerando siempre la restricción de fragilidad al realizar los movimientos. Para esta tarea debe utilizar 2 pilas auxiliares y variables de tipo entero. (5 puntos).

Luego de un tiempo la empresa se da cuenta que contar solo con 2 pilas auxiliares genera mayor complejidad en el trabajo del robot HEX, por tal motivo realiza una inversión y adquiere una pila auxiliar adicional.

- b) Implemente un programa para el robot apilador, que sirva para colocar los elementos de la faja transportadora en la pila principal, considerando siempre la restricción de fragilidad al realizar los movimientos. Para esta tarea debe utilizar 3 pilas auxiliares y variables de tipo entero. (5 puntos).

**Para esta pregunta no puede usar ningún arreglo, matriz o TAD auxiliar adicional a la indicada, solo variables simples. No es necesario el ingreso de datos.**

Profesores del curso:

David Allasi  
Fernando Huamán  
Rony Cueva

San Miguel, 14 de octubre del 2023