

PONTIFICIA UNIVERSIDAD CATÓLICA DEL PERÚ
FACULTAD DE CIENCIAS E INGENIERÍA

ALGORITMIA Y ESTRUCTURA DE DATOS
Segundo Examen
(Primer Semestre 2024)

Duración: 2h 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`, `iomanip`, `limits`, `cstring`, `cmath`, `vector`, `list`, `map` o `fstream`
- Su trabajo deberá ser subido a PAIDEIA.
- **Es obligatorio usar como compilador NetBeans.**
- Los archivos deben llevar como nombre su código de la siguiente forma `codigo_EX1_P#` (donde # representa el número de la pregunta a resolver)

Pregunta 1 (10 puntos)

Un tiempo después del ataque de SkyNerd, un grupo de especialistas en algoritmia desarrollan varios bloques de código que, en un orden establecido logran liberar a los servidores contagiados por este programa nocivo. Para poder limpiar la red es necesario enviar los bloques de código lo más rápido posible, ya que, si el tiempo computacional es alto, SkyNerd puede buscar la forma de bloquear la desinfección. Se sabe que los bloques de código se encuentran almacenados en una pila y cada bloque se representa como un número entero.

Por tal motivo se necesita que se desarrolle un algoritmo que clone lo más rápido posible la pila de bloques de código original. Finalmente muestre los datos de la pila clon. **Para que esta solución sea efectiva su complejidad debe ser $O(n)$.**

Para el desarrollo de esta pregunta solo puede usar 2 pilas de enteros: la pila original y la pila clon. Los datos de la pila origen deben ir en duro. No puede superar la complejidad en ningún momento o parte del desarrollo del programa. **Para el desarrollo puede emplear cualquiera de las estrategias algorítmicas vistas en el curso.**

Pregunta 2 (10 puntos)

Intensamente 2 es una película animada estadounidense producida por Pixar Animation Studios para Walt Disney Pictures. Esta película muestra las diversas emociones que presenta una niña

de 13 años llamada Riley, las cuáles son: Alegría, Tristeza, Temor, Furia, Desagrado, Ansiedad, Envidia, Vergüenza y Aburrimiento.

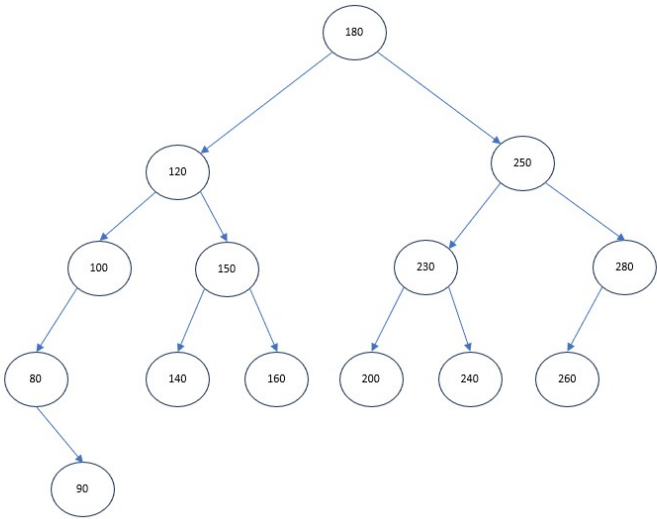
En esta ocasión vamos a trabajar estas emociones en un árbol binario de búsqueda de emociones, para ello se le ha asignado a cada emoción un peso o conjunto de pesos (en algunos casos) de acuerdo con la siguiente tabla:

Emoción	Peso1	Peso2	Peso 3
Alegría	180	200	
Tristeza	140	230	80
Temor	240	100	
Furia	280		
Desagrado	250		
Ansiedad	260	120	
Envidia	160	240	
Vergüenza	120	80	
Aburrimiento	150	230	90

Tabla 1. Pesos para las emociones

Como puede observar, algunos pesos se pueden repetir entre emociones.

La idea de manejar estos pesos y este árbol de emociones es poder tratar de representar los cambios de emociones que se pueden dar en Riley. Un posible árbol de emociones es el siguiente:



Ahora, vamos a tener un conjunto de cambios de emociones que tiene Riley y vamos a buscarlos en el árbol de emociones para ver si se encuentran.

Un ejemplo de este conjunto de cambios de emociones es el siguiente:

Nro. Combinación	Emoción 1	Emoción 2	Emoción 3
1	Ansiedad	Aburrimiento	Tristeza
2	Desagrado	Tristeza	Temor
3	Temor	Vergüenza	Aburrimiento

Tabla 2. Combinaciones de emociones.

Estos cambios de emociones se deben dar en el orden que aparecen las emociones en la tabla anterior, por ejemplo, los cambios en la combinación 1 debe ser en este orden, primero Alegría, luego Aburrimiento y finalmente Tristeza.

Para buscar los cambios de emociones en el árbol, se debe realizar desde el nivel 1 (raíz) hasta el último nivel. La búsqueda siempre debe ser por niveles, es decir, primer buscaré todas las posibles combinaciones que se pueden formar desde el nivel 1, luego desde el nivel 2 y así hasta el nivel donde se pueda buscar.

Por ejemplo, para este caso:

- El nivel 1 tiene el peso 180, por lo tanto, el peso 180 solo corresponde a la emoción Alegría (ver tabla 1), y si vemos las combinaciones de emociones, ninguna empieza con Alegría, por lo tanto, desde el Nivel 1 no se puede encontrar ninguna combinación.
- El nivel 2 tiene los pesos 120 (Ansiedad y Vergüenza) y 250 (Desagrado), vamos a ver si iniciando con esos pesos encontramos alguna combinación:
 - Para el peso 120 que puede ser Ansiedad y Vergüenza, vemos que la combinación de emociones 1 inicia en Ansiedad, por lo que esa combinación podría ser, para confirmarlo debemos verificar los hijos del peso 120 para ver si alguno de ellos corresponde a Aburrimiento. En este caso los hijos del peso 120 son 100 (Temor) y 150 (Aburrimiento), por lo que la combinación 1 se va confirmando, dado que el peso 150 es Aburrimiento. Para terminar de confirmar la combinación de emociones, reviso los hijos del peso 150 para ver si alguno es Tristeza. Los hijos del peso 150 son 140 (Tristeza) y 160 (Envidia). Con esto confirmamos que la combinación 1 si se encuentra porque el peso 140 es Tristeza.
 - Para el peso 250 que es Desagrado, vemos que la combinación 2 inicia en Desagrado, por lo que esa combinación podría ser, para confirmarlo debemos verificar los hijos del peso 250 para ver si alguno de ellos corresponde a Tristeza. En este caso los hijos del peso 250 son 230 (Tristeza y Aburrimiento) y 280 (Furia), por lo que la combinación 2 se va confirmando, dado que el peso 230 es Tristeza. Para terminar de confirmar la combinación de emociones, reviso los hijos del peso 230 para ver si alguno es Temor. Los hijos del peso 230 son 200 (Alegría) y 240 (Temor y Envidia). Con esto confirmamos que la combinación 2 si se encuentra porque el peso 240 es Temor.
 - Entonces, se concluye que iniciando en el nivel 2 se pueden encontrar 2 combinaciones.
- El nivel 3 tiene los pesos 100, 150, 230 y 280. Iniciando en este nivel se ve que solo se puede trabajar el peso 100 porque es el único que podría encontrar la combinación de 3 emociones, por lo que analizaremos solo ese.
 - Para el peso 100 que puede ser Temor, vemos que la combinación de emociones 3 inicia en Temor, por lo que esa combinación podría ser, para confirmarlo debemos verificar los hijos del peso 100 para ver si alguno de ellos corresponde a Vergüenza. En este caso los hijos del peso 100 solo es 80 (Vergüenza y Tristeza), por lo que la combinación 3 se va confirmando, dado que el peso 80 es Vergüenza. Para terminar de confirmar la combinación de emociones, reviso los hijos del peso 80 para ver si alguno es Aburrimiento. Los hijos del peso 80 solo es 90 (Aburrimiento). Con esto confirmamos que la combinación 3 si se encuentra porque el peso 90 es Aburrimiento.

- Entonces, se concluye que iniciando en el nivel 3 se puede encontrar 1 combinación.

Por lo tanto, la salida del programa será:

Nivel 1 – 0 combinaciones

Nivel 2 – 2 combinaciones

Nivel 3 – 1 combinación

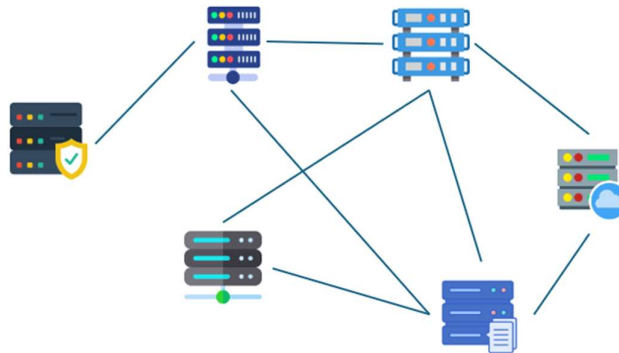
Se le pide:

- Elabore una función que permita indicar la cantidad de combinaciones que se pueden encontrar, siguiendo estrictamente las indicaciones dadas en el enunciado. El resultado debe mostrar el nivel donde inicia la combinación (8 puntos).
- Elabore una función main que lea los datos necesarios para resolver la pregunta y utilizando la función anterior pueda dar solución al problema (2 puntos).

Para el desarrollo de esta pregunta solo puede usar como estructuras auxiliares: listas, pilas o colas.

Pregunta 3 (10 puntos)

- (5.0 puntos)** Una empresa ha implementado una red interna de servidores que intercambian datos sensibles. Para asegurar la integridad y la confidencialidad de la información, se deben establecer rutas seguras y eficientes entre los servidores. Además de la latencia entre los servidores, cada conexión tiene un nivel de seguridad asociado, que varía en función de factores como la encriptación, autenticación y otras medidas de seguridad implementadas.



La empresa necesita un algoritmo que encuentre entre un servidor origen y un servidor destino la menor latencia posible, calculando además el nivel de seguridad de esa transferencia de información.

Utilice los datos de la siguiente tabla (matriz de adyacencia del grafo, la cual proporciona los valores de latencia entre cada servidor, matriz paralela que proporciona los niveles de seguridad de cada conexión).

	Matriz de adyacencia	Matriz Niveles seguridad	Salida
1	{0, 0, 0, 3, 0, 0}, {0, 0, 1, 0, 3, 0}, {0, 1, 0, 2, 5, 7}, {3, 0, 2, 0, 7, 0}, {0, 3, 5, 7, 0, 2}, {0, 0, 7, 0, 2, 0}	{0, 0, 0, 9, 0, 0}, {0, 0, 7, 0, 8, 0}, {0, 7, 0, 7, 5, 7}, {9, 0, 7, 0, 7, 0}, {0, 8, 5, 7, 0, 9}, {0, 0, 7, 0, 9, 0}	Servidor: 1 Latencia: 0 Seguridad Promedio: 0 Servidor: 2 Latencia: 6 Seguridad Promedio: 7.66667 Servidor: 3 Latencia: 5 Seguridad Promedio: 8 Servidor: 4 Latencia: 3 Seguridad Promedio: 9 Servidor: 5 Latencia: 9 Seguridad Promedio: 7.75 Servidor: 6 Latencia: 11 Seguridad Promedio: 8
2	{0, 0, 0, 3, 0, 0}, {0, 0, 3, 0, 3, 0}, {0, 3, 0, 2, 5, 7}, {3, 0, 2, 0, 7, 0}, {0, 3, 5, 7, 0, 2}, {0, 0, 7, 0, 2, 0}	{0, 0, 0, 9, 0, 0}, {0, 0, 7, 0, 8, 0}, {0, 7, 0, 7, 5, 7}, {9, 0, 7, 0, 7, 0}, {0, 8, 5, 7, 0, 9}, {0, 0, 7, 0, 9, 0}	Servidor: 1 Latencia: 0 Seguridad Promedio: 0 Servidor: 2 Latencia: 8 Seguridad Promedio: 7.66667 Servidor: 3 Latencia: 5 Seguridad Promedio: 8 Servidor: 4 Latencia: 3 Seguridad Promedio: 9 Servidor: 5 Latencia: 10 Seguridad Promedio: 8 Servidor: 6 Latencia: 12 Seguridad Promedio: 7.66667

NOTA: Desde el programa principal debe inicializar las matrices con los valores brindados en la tabla anterior y enviarlos como argumentos al invocar a la función que permita obtener lo indicado en la columna “Salida

- b. **(5 puntos)** Una empresa de tecnología, "TechAlgoritmia", ha decidido optimizar su planilla debido a la automatización avanzada y la inteligencia artificial, se necesita reducir el personal. La dirección ha determinado que se reducirá el personal de manera que cada k-ésimo empleado será despedido hasta que solo quede uno. Este último empleado será nombrado el nuevo director de innovación. Los empleados están inicialmente numerados del 1 al n.

Por ejemplo, suponga que tenemos una lista de 7 empleados numerados del 1 al 7:

1 2 3 4 5 6 7

Si decidiéramos eliminar un empleado de la lista de cada tres, el primer empleado que se eliminaría sería el número 3, dejando la lista:

1 2 4 5 6 7

El siguiente empleado que se eliminaría sería el número 6, lo que nos dejaría:

1 2 4 5 7

Se considera que los empleados pertenecen a un ciclo continuo, de modo que cuando alcanzamos el final de la lista continuamos contando por el principio. De este modo, el siguiente empleado que se eliminaría sería el número 2, quedando la lista:

1 4 5 7

El siguiente empleado que se eliminaría sería el número 7, dejando la lista:

1 4 5

El siguiente empleado que se eliminaría sería el número 5, quedándonos:

1 4

El penúltimo empleado en eliminarse sería el número 1, y con eso nos quedaría el número 4 como el último elemento de la lista.

Se pide implementar un algoritmo que resuelva el problema de reducción de personal. Se debe considerar el uso de una **lista circular** para resolver el problema, por lo cual se deben establecer condiciones adecuadas para evitar caer en ciclos infinitos, puede hacer uso de las funciones de lista y adecuarlas al problema. **El desarrollo de esta pregunta puede ser iterativo o recursivo, con una complejidad no mayor a $O(n)$.**

Caso de prueba

En una lista de 5 empleados, para $k = 2$, el elegido es: 3

En una lista de 7 empleados, para $k = 3$, el elegido es: 4

En una lista de 10 empleados, para $k = 7$, el elegido es: 9

Al finalizar el examen, 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 examen.

Profesores del curso:

Ana Roncal
Fernando Huamán
David Allasi
Rony Cueva

San Miguel, 6 de julio del 2024