

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

ALGORITMIA

Laboratorio 4

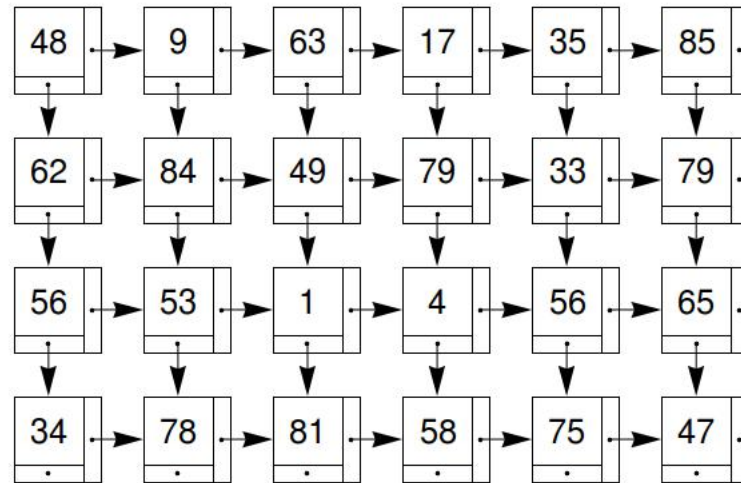
2015-2

Indicaciones generales:

- Duración: 2h 50 min.
 - Al inicio de cada programa, el alumno deberá incluir, a modo de comentario, la estrategia que utilizará para resolver el problema. De no incluirse dicho comentario, el alumno perderá el derecho a reclamo en esa pregunta.
 - Si la implementación es significativamente diferente a la estrategia indicada o no la incluye, la pregunta será corregida sobre el 50 % del puntaje asignado y sin derecho a reclamo.
 - Un programa que no muestre resultados coherentes y/o útiles será corregido sobre el 60 % 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.
 - Su trabajo deberá ser subido a PAIDEIA en el espacio indicado por los Jefes de Práctica.
-

Pregunta 1 (12 puntos)

Se desea implementar una estructura de datos (Matrix) similar a una matriz de enteros, pero donde cada elemento es un nodo similar al de una lista enlazada. Cada nodo contiene dos referencias a nodos siguientes (derecha y abajo), como se muestra a continuación.



Defina los tipos de datos necesarios (2 puntos) e implemente las funciones:

- void Matrix_print(Matrix matrix) (1 punto)
Imprime los elementos de *matrix*, de manera que la *i*-ésima línea impresa contiene los elementos de la *i*-ésima fila de *matrix*.
- void Matrix_addColumn(Matrix* matrix, List column) (1.5 puntos)
Agrega los elementos de la lista simplemente enlazada *column* como una columna nueva de *matrix*. La cantidad de elementos de *column* debe coincidir con la cantidad de filas de *matrix* a menos que *matrix* esté vacía.
- void Matrix_addRow(Matrix* matrix, List row) (1.5 puntos)
Agrega los elementos de la lista simplemente enlazada *row* como una fila nueva de *matrix*. La cantidad de elementos de *row* debe coincidir con la cantidad de columnas de *matrix* a menos que *matrix* esté vacía.
- void Matrix_deleteColumn(Matrix* matrix, int i) (1.5 puntos)
Elimina la *i*-ésima columna de *matrix*.
- void Matrix_deleteRow(Matrix* matrix, int i) (1.5 puntos)
Elimina la *i*-ésima fila de *matrix*.
- void Matrix_transpose(Matrix matrix, Matrix* transpose) (2 puntos)
Crea la matriz transpuesta de *matrix* y guarda el resultado en *transpose*.
- void main (1 punto)
Su función *main* debe incluir llamadas a las funciones solicitadas anteriormente de manera que se pueda comprobar que las funciones implementadas están funcionando correctamente.

Pregunta 2 (8 puntos)

En esta pregunta deberá implementar un programa que permita comprobar si una frase es PALÍNDROMA. Una frase es palíndroma si se puede leer igual de izquierda a derecha o de derecha a izquierda.

Algunos ejemplos:

A man, a plan, a canal... Panama

Madam I'm Adam

Ah, Satanás, amas a Natasha?

La ruta nos aportó otro paso natural

Condiciones:

- Solo puede utilizar una Pila y una Cola como herramientas para la ejecución de su programa.
- NO deberá considerar signos de puntuación, tildes, diferencia en mayúsculas o minúsculas ni espacios en blanco al momento de realizar su comparación.
- Use el archivo p2l4.txt proporcionado al inicio del laboratorio.
- Por cada caso de prueba deberá devolver Falso o Verdadero.

Profesores del curso: Fernando Alva
Robert Ormeño

Pando, 27 de octubre del 2015