PONTIFICIA UNIVERSIDAD CATÓLICA DEL PERÚ
FACULTAD DE CIENCIAS E INGENIERÍA

ALGORITMIA
Examen 1
(Segundo semestre 2016)

Indicaciones generales:

- Duración: 3h.
- Materiales o equipos a utilizar: Apuntes de clase personales escritos a mano.
- 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 o si la implementación es significativamente diferente de la estrategia indicada, el alumno no obtendrá el puntaje completo en dicha pregunta.
- 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 considerado dentro de la evaluación.
- Cada programa debe ser guardado en un archivo con el nombre *preg#codigo_de_alumno.c* y subido a PAIDEIA en el espacio indicado por los Jefes de Práctica.
- La presentación, la ortografía y la gramática de los trabajos influirá en la calificación.

Puntaje total: 20 puntos

PREGUNTA 1 (7 puntos)

La empresa Technological Pieces S.A. se especializa en tender cableado de fibra óptica. Technological Pieces S.A. atiende solicitudes de diversas organizaciones tales como municipalidades, universidades, colegios, etc. Cuando una organización llega con un requerimiento a la empresa, generalmente trae un número de kilómetros n, que es el número de kilómetros de tendido que desean.

Pero hay un detalle (siempre hay detalles!). Technological Pieces S.A. compra porciones de fibra óptica de 3 tamaños a, b, c (porque así le cuesta menos la importación). Además Technological Pieces S.A. cobra una cantidad de dinero proporcional a la cantidad de porciones de cable que usa para tender los n kilómetros de un requerimiento, por lo que es necesario conocer el máximo número de porciones de cable que puede obtenerse para cubrir n kilómetros.

Los tamaños de las porciones a, b, c también vienen en unidad de kilómetros y existe un número ilimitado de cables de cada tamaño.

Entrada

La primera línea contiene 4 números enteros n, a, b, c ($1 \leq n, a, b, c \leq 4000$) – la cantidad de kilómetros a cablear y las longitudes de las porciones a, b, c . Los números a, b, c pueden coincidir.

Salida

Imprimir un solo número – el máximo número posible de porciones de cable que cubren los n kilómetros de manera exacta (es decir no debe sobrar ningún kilómetro sin cubrir).

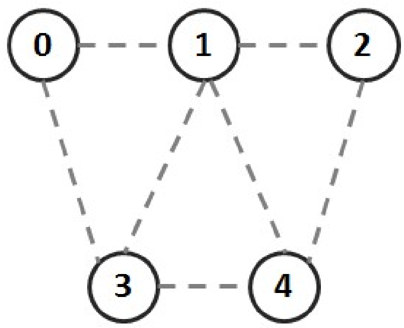
Ejemplos

Entrada	**Salida**
5 5 3 2	2
7 5 5 2	2

- En el primer caso, las porciones a usar sería 2 y 3
- En el segundo caso, las porciones a usar con 5 y 2

PREGUNTA 2 (7 puntos)

Un grafo es un tipo abstracto de datos que consiste en un conjunto de nodos (también llamados vértices) y un conjunto de arcos (aristas) que establecen relaciones entre los nodos. Una de las formas de representar un grafo es a través de una matriz de adyacencia. En el lado izquierdo de la Figura 1 se presenta un grafo con 5 vértices. La figura a la derecha es una matriz de adyacencia de tamaño 5 (número de vértices) que representa el grafo. Cada celda (i, j) puede contener 2 valores: 1 en caso exista una arista desde “i” hasta “j”, y 0 en caso no exista arista. Por ejemplo, la celda (0, 1) tiene como valor 1 ya que existe una arista desde el vértice 0 hasta el vértice 1. Ya en el caso de la celda (0, 2), puede notarse que el valor es 0, ya que no existe una arista que conecte el vértice 0 con el vértice 2.



	0	**1**	**2**	**3**	**4**
0	0	1	0	1	0
1	1	0	1	1	1
2	0	1	0	0	1
3	1	1	0	0	1
4	0	1	1	1	0

Figura 1. Grafo y matriz de adyacencia

Un camino hamiltoniano es un camino que visita todos los vértices de un grafo una vez. Un ciclo Hamiltoniano es un camino Hamiltoniano en el cual el último vértice visitado es adyacente al primer vértice visitado (existe una arista que los conecta). Dado un grafo, determinar si el mismo contiene un ciclo hamiltoniano o no. En el caso que contenga, imprimir el camino.

Entrada

La primera línea corresponde al número de casos a analizar. Para cada caso de prueba, la primera línea representa el número de vértices “n” del grafo. Las siguientes “n” líneas representan las filas de la matriz de adyacencia del grafo, donde cada fila está compuesta por "n" columnas.

Salida

En caso exista un ciclo hamiltoniano, este debe ser impreso. En caso no exista un ciclo hamiltoniano, se debe mostrar el mensaje *"No existe ciclo hamiltoniano"*.

Ejemplos

Entrada	**Salida**
2	
5 0 1 0 1 0 1 0 1 1 1 0 1 0 0 1 1 1 0 0 1 0 1 1 1 0	0 1 2 4 3 0
5 0 1 0 1 0 1 0 1 1 1 0 1 0 0 1 1 1 0 0 0 0 1 1 0 0	No existe ciclo hamiltoniano

PARTE ELECTIVA

Resuelva **UNA** de las preguntas que se presentan a continuación.

PREGUNTA 3 (6 puntos) – Programación Dinámica

Una secuencia de números es llamada zig-zag cuando las diferencias entre números sucesivos alternan estrictamente entre positivo y negativo. Las secuencias con una cantidad de elementos menor a 2 son consideradas secuencias zig-zag. Adicionalmente, la primera diferencia (si existiese) puede ser positiva o negativa. Por ejemplo, 1, 7, 4, 9, 2, 5 es una secuencia zig-zag porque las diferencias (6, -3, 5, -7, 3) son positivas y negativas alternadamente. En contraste, 1, 4, 7, 2, 5 y 1, 7, 4, 5, 5 no son secuencias zig-zag, la primera porque sus 2 primeras diferencias son positivas y la segunda porque la última diferencia es 0.

Dada una secuencia de enteros, retornar la longitud de la **sub-secuencia** zig-zag más larga. Una sub-secuencia es obtenida por la eliminación de un número de elementos (o ninguno) de la secuencia original, dejando los elementos restantes en su posición original.

Entrada

La primera línea contiene un entero n – el número de casos a evaluar. La primera línea de cada caso contiene un valor m $(1 \leq m \leq 50)$, que representa el número de elementos de la secuencia a leer. La segunda línea de cada caso contiene los elementos de la secuencia separados por espacios, donde cada elemento puede ir desde el número 1 hasta el 1000.

Salida

Imprimir un solo número entero – la longitud de la sub-secuencia zig-zag más larga.

Ejemplos

Entrada	**Salida**
3	
6 1 7 4 9 2 5	6
1 44	1
10 1 17 5 10 13 15 10 5 16 8	7

- En el primer caso, todas las diferencias son alternadas (positiva/negativa)
- En el segundo caso, dado que hay un solo elemento, este forma la sub-secuencia zig-zag
- En el tercer caso, algunas de las posibles soluciones podrían ser 1, 17, 5, 10, 5, 16, 8 ó 1, 17, 5, 13, 5, 16, 8.

PREGUNTA 4 (6 puntos) – Divide y vencerás

Félix tiene una nueva cerca al frente de su jardín. Por ahora la cerca luce un pálido color blanco y Félix ha decidido ponerle color a la cerca. La cerca se representa como n tablas verticales puestas en fila. No existen espacios entre tabla y tabla. Las tablas están enumeradas de izquierda a derecha, la i-ésima tabla tiene el ancho de un metro y una altura de a_i metros.

Félix ha comprado una brocha cuyo ancho tiene un metro. Él puede hacer trazos verticales u horizontales con la brocha. Durante un trazo, la superficie completa de la brocha está en contacto con la cerca todo el tiempo. ¿Cuál es el mínimo número de trazos que Félix debería hacer para pintar la cerca entera? Note que puede ser posible pintar la misma área de la cerca varias veces.

Entrada

La primera línea contiene un entero n $(1 \leq n \leq 5000)$ – el número de tablas de la cerca. La segunda línea contiene n números enteros separados por espacios $a_1, a_2, \ldots, a_n (1 \leq a_i \leq 10^9)$.

Salida

Imprimir un solo número entero – el mínimo número de trazos necesarios para pintar la cerca entera.

Ejemplos

Entrada	Salida
5 2 2 1 2 1	3
2 2 2	2
1 5	1

- En el primer caso, Félix necesita pintar la cerca en tres trazos con la brocha: el primer trazo pinta una altura de 1 horizontalmente a través de todas las tablas. El segundo trazo se hace a la altura 2 y pinta la primera y segunda tabla y el tercer trazo pinta la cuarta tabla.
- En el segundo caso, se puede pintar la cerca ya sea con dos trazos verticales o dos horizontales.
- En el tercer caso hay una sola tabla que puede ser pintada usando un solo trazo vertical.

Profesor del curso: Marco Sobrevilla

Iván Sipiran

San Miguel, 15 de octubre de 2016