

PONTIFICIA UNIVERSIDAD CATÓLICA DEL PERÚ

FACULTAD DE CIENCIAS E INGENIERÍA

ALGORITMIA

Laboratorio 3

Primer Semestre de 2020

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.
 - Los programas deben ser desarrollados en Ansi 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.
 - Su trabajo deberá ser subido a PAIDEIA en el espacio indicado por los jefes de práctica.
-

Pregunta 1 (10 puntos)

GTIN-13 (Global Trade Item Number) es el sistema estándar y uniforme para la codificación e identificación de productos que llegan al consumidor final en todo el mundo, siendo el Código de Barras su representación más conocida que permite la lectura electrónica en un punto de venta. GTIN-13 tiene un total de 13 dígitos, en el que los tres primeros corresponden al código del país (775 en el caso de Perú), los 4 siguientes corresponden al código de empresa, los 5 siguientes corresponden al código del producto y el último dígito corresponde al dígito de control que es calculado. El supermercado Plaza Métrica utiliza este sistema para la codificación de sus productos, y se puede ver un ejemplo de código GTIN-13 para un producto del supermercado en la Fig. 1.



Fig. 1: Ejemplo de código de barra GTIN - 13

(Fuente: https://gs1pe.org/sites/default/files/cod_gitin_13.jpg)

Adicional al código descrito, Plaza Métrica también utiliza un código extendido de 32 dígitos por producto para la gestión de inventario de la tienda al cual denomina trama. La trama de datos de este sistema tiene la siguiente estructura:

`<código_GTIN-13><precio_en_soles><stock><stock_mínimo><descuento>`

Donde:

- `código_GTIN-13`: corresponde a los 13 dígitos para el código GTIN-13 del producto.
- `precio_en_soles`: corresponde a los 7 dígitos para el precio original en soles del producto.
- `stock`: corresponde a los 5 dígitos para el stock disponible del producto.
- `stock_mínimo`: corresponde a los 4 dígitos para el valor considerado como stock mínimo para ese producto.
- `descuento`: corresponde a los 3 dígitos para el porcentaje de descuento.

Como ejemplo, analicemos la siguiente trama:

```
77512345678920039990000040005060
```

La trama presentada contiene la siguiente información:

```
GTIN-13: 7751234567892
Precio: 0039990 (S/. 399.90)
Stock: 00004 (4 unidades)
Stock mínimo: 0005 (5 unidades)
Descuento: 060 (60% de descuento)
```

Para impulsar la rotación de venta de determinados productos no perecibles, el supermercado ha tomado la medida de poner a remate aquellos productos que están con un stock por debajo de su stock mínimo. La ubicación de los productos a rematar será en un pasillo de solo ofertas, colocando en una pantalla digital, delante del pasillo, la información de los productos que están en oferta, mostrándose en el mismo orden cómo están ordenados en el pasillo, además, se muestra en la pantalla el título de “Hasta XX% de descuento”, donde XX corresponde al mayor porcentaje de descuento que puede tener uno de los productos del pasillo.

Para ordenar los productos que irán en el pasillo de ofertas se usará el siguiente criterio: el orden es de modo descendente en base al precio final de venta del producto. El precio final de venta del producto se obtiene aplicando el porcentaje de descuento al precio original. Veamos el siguiente ejemplo:

Si la cantidad de productos (n) de la tienda es 6, entonces se registrará, en un arreglo de tramas (matriz bidimensional de dígitos), 6 tramas de 32 dígitos cada uno.

```
/* Matriz ingresada sin ordenar */
Matriz_inventario = {
    77512345678920015990000040005060,
    77512341234560009690000400050030,
    77512349876540009990000300039035,
    77512346969690001990003000100010,
    77512346669990014990000110010020,
    77512349696960006990000200026025
}
```

Los productos que irán al pasillo de ofertas serán los productos del primer grupo luego de haber reagrupado la matriz en dos grupos lógicos, aquellos cuyo stock está por debajo del mínimo y aquellos cuyo stock es igual o mayor a su stock mínimo. **Usted puede definir el criterio de cómo reagrupar, considerando que la estrategia a usar es “divide y vencerás” aplicando los algoritmos de MergeSort, QuickSort y Binary Search.** Para el ejemplo, luego de haber reagrupado la matriz, el primer grupo estaría formado por los 4 primeros productos y el segundo grupo por los otros 2 restantes:

```
/* Matriz reagrupada en dos grupos */
Matriz_inventario = {
    77512341234560009690000400050030,
    77512349876540009990000300039035,
    77512349696960006990000200026025,
    77512345678920015990000040005060,
    77512346669990014990000110010020,
    77512346969690001990003000100010
}
```

Luego se debe de ordenar los productos que irán en el pasillo de ofertas, es decir, solo los del primer grupo. Tenga en cuenta que el criterio para el ordenamiento es en modo descendente en base al precio

final de venta del producto que se obtiene aplicando el porcentaje de descuento al precio original. **La estrategia a usar para el ordenamiento es “divide y vencerás”**. Para el ejemplo, luego de ordenar los datos del primer grupo quedaría de la siguiente manera:

```
/* Matriz ordenada por el precio final para el primer grupo */
Matriz_inventario = { 77512341234560009690000400050030,
                      77512349876540009990000300039035,
                      77512345678920015990000040005060,
                      77512349696960006990000200026025,
                      77512346669990014990000110010020,
                      77512346969690001990003000100010
                    }
```

Finalmente, se mostraría en la pantalla el mayor descuento y los productos de oferta ordenados con sus respectivos descuentos y precios finales:

```
Hasta 60% de descuento
Los productos del pasillo de ofertas son:
7751234123456 | 30% descuento | P.F: S/. 67.83
7751234987654 | 35% descuento | P.F: S/. 64.935
7751234567892 | 60% descuento | P.F: S/. 63.96
7751234969696 | 25% descuento | P.F: S/. 52.425
```

NOTA: el algoritmo debe funcionar para cualquier cantidad de productos registrados en el sistema del supermercado.

- Desarrolle el ingreso adecuado de los datos vía teclado que corresponde al llenado de la base de datos de productos del supermercado. Primero indique la cantidad de productos que va a registrar, luego ingrese, para cada producto, su información codificada propia del supermercado. (1 punto)
- Desarrolle una función que utilice la estrategia “divide y vencerás” para reagrupar la matriz en dos grupos, los que irán al pasillo de ofertas y los que no. Recuerde que el primer grupo corresponde a los productos cuyo stock está por debajo de su stock mínimo, y el segundo grupo corresponde a los productos cuyo stock está igual o por encima de su stock mínimo. La función debe también retornar la cantidad de productos que irán al pasillo de ofertas. (4.5 puntos)
- Desarrolle una función que utilice la estrategia “divide y vencerás”, para ordenar los productos que irán al pasillo de productos en oferta, considerando que el ordenamiento es de mayor a menor por el precio final luego de haber aplicado el descuento a cada producto. (2.5 puntos)
- En el programa principal se debe mostrar el mayor descuento en porcentaje aplicado a los productos del pasillo y el listado de productos que irán al pasillo de ofertas. La búsqueda del mayor descuento debe ser usando “divide y vencerás”. Recuerde que los productos están ordenados de mayor a menor por el precio final luego de haber aplicado el descuento. (2 puntos).

Pregunta 2 (10 puntos)

Una empresa dedicada a la consultoría informática, brinda servicios de tercerización de especialistas a las organizaciones que lo requieran (clientes). Actualmente la empresa cuenta con una planilla de N consultores, los cuales están divididos de acuerdo a su experiencia, conocimientos y compromiso al realizar los proyectos asignados. Estos grupos son Consultores Junior (1), SemiSenior (2) y Senior (3). Por tal motivo cuando un cliente les solicita uno o más especialistas, se necesita saber con cuantos consultores se cuenta de acuerdo al nivel solicitado, y el costo de los profesionales. Los datos que se consideran por cada consultor son: código, años de experiencia, nivel y sueldo (cantidades enteras).

Cuando un cliente solicita un consultor, debe indicar el nivel y los años de experiencia. La empresa busca entre sus consultores la cantidad de profesionales que cumplan con este requerimiento. Además con el fin de brindarles mejores opciones, en esta elección se considera a los consultores de niveles superiores, pero sin tomar en cuenta sus años de experiencia. Finalmente los clientes indican la cantidad de dinero con que cuentan (P) y la empresa elige los consultores que le alcanzan con ese presupuesto, desde luego sin superarlo y buscando el ahorro.

Ejemplo del ingreso de datos de empresa:

N = 8

Código	Años de experiencia	Nivel	Sueldo (Miles\$)
1320	10	2	5
1317	12	3	8
1021	10	3	7
1909	3	2	6
1000	10	2	6
1989	2	1	3
1590	5	1	4
1489	4	1	4

Ejemplo del ingreso de datos del cliente:

Años de experiencia = 5 Nivel = 2 P = 12

Ejemplo de salida:

Hay 4 consultores para lo requerido.

Con 12k alcanzan los siguientes consultores: 1320 - 5k 1000 – 6k

- Desarrolle el ingreso adecuado de los datos vía teclado, tanto para la carga de los datos de los consultores, como los requerimientos del cliente (1.0 punto).
- Realice una función de ordenamiento que utilice la estrategia “divide y vencerás”, para organizar la información de la empresa, que ayude a resolver la pregunta siguiente (2.0 puntos)
- Desarrolle una función que utilice la estrategia “divide y vencerás”, que obtenga la cantidad consultores que pueden cumplir con los requerimientos del cliente, el algoritmo debe tener una complejidad $O(\log n)$ (5 puntos).
- Imprima los códigos de los consultores que el cliente puede contratar, con un presupuesto P, maximizando la cantidad de profesionales, para esta operación utilice la estrategia “divide y vencerás”. Puede omitir la estrategia para la impresión de datos (2 puntos).

Profesores del curso: Johan Baldeon
 Rony Cueva
 Ivan Sipiran

Pando, 18 de mayo del 2020