



Profesora: Mónica Villanueva

Ayudante: Nicolás Gutiérrez

## Enunciado 2 : Programación Dinámica.

La empresa minera *Clover ores* ha construido recientemente una nueva cantera en una zona remota de un país. En esta cantera se extraen diferentes minerales, pero uno de los principales es el oro.

El material extraído es llevado en láminas rectangulares al centro de clasificación y filtrado. Hace poco la máquina que filtra los materiales se ha dañado por lo que es necesario una nueva. Ante este problema la empresa le solicita que cree un algoritmo para una nueva máquina que clasifique la lámina de minerales que se le ingresa.

La lámina posee cuadros donde hay o no oro, por lo que la máquina debe cortar esta lámina de tal forma de dejar todo el oro junto y los cuadros que no tengan oro juntos.

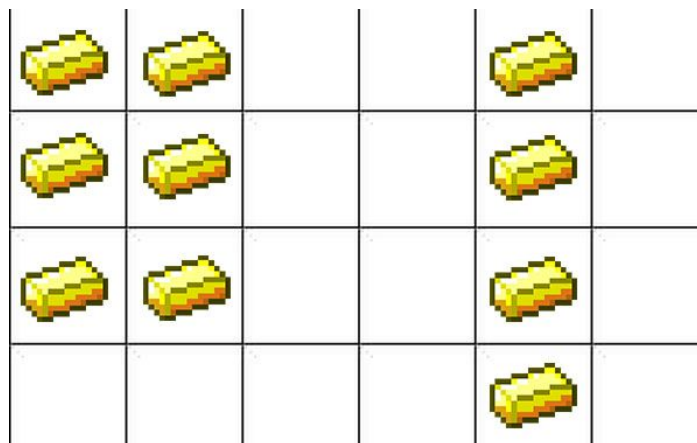


Figura 1: Ejemplo lámina de minerales.

La lámina rectangular puede ser cortada en forma horizontal y vertical, los cortes que se realicen deben dejar figuras cuadradas y rectangulares.

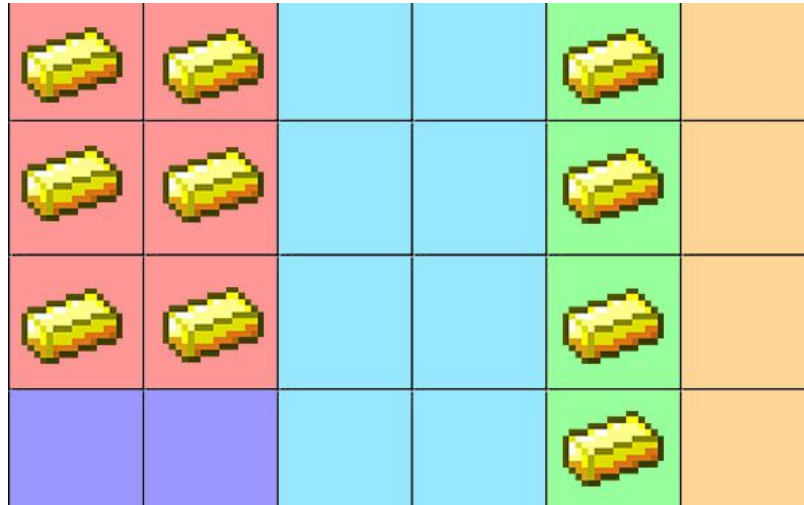


Figura 2: Ejemplo de corte.

La empresa ha solicitado que el programa sea desarrollado por programación dinámica

### **Funcionalidades:**

Construya un programa en C que logre entregar cuantos cortes se deben realizar sobre una lámina de minerales con tal de garantizar que los cuadros dentro de una sección cortada sean del mismo tipo. (cada sección debe tener solo cuadros de oro o solo cuadros sin oro)

Su código debe contener la función **dinamica (entradas)** .

La entrada de la función anterior queda a su criterio, pero debe comentar que es lo que entra en la función. Por ejemplo:

```
/*  
  ENTRADAS: dos enteros que representan la cantidad de ciudades  
             y la cantidad de recursos respectivamente.  
  
  SALIDA: entrega como salida un puntero al nodo inicial de la  
           lista dónde está la ruta óptima.  
*/  
nodo* dinamica (int A, int B) {...}
```



Su código debe incluir la función ***printCurrent (entradas)*** que imprime por pantalla el estado actual del nodo que está siendo visitado (o generado). Esta función es solo para el control del flujo de la ejecución, le ayudará a la revisión y la ubicación dentro de su código. La entrada de la función anterior queda a su criterio, pero debe comentar que es lo que entra en la función. Para mayor facilidad se le facilitara parte del código

```
printCurrent(...) {  
    }
```

Para la parte de ejecución del programa este debe ser del tipo:

```
$ ./ejecutable archivo_entrada archivo_salida
```

o en windows:

```
$ ejecutable.exe archivo_entrada archivo_salida
```

### ***Entrada:***

Como entrada se debe tener un archivo “*entrada.in*” el cual contiene; una matriz que representa una lámina donde el carácter “#” representa el oro y el carácter “.” representa un cuadro sin oro.

### ***Formato de la entrada:***

Entrada1.in	Entrada2.in	Entrada3.in
5 5 .##.. .#### ##... ..... ..###	1 6 ...###	2 2 #. .#



La primera línea al tamaño de la matriz en formato NxM. Las siguientes líneas son las líneas de datos que tiene la matriz

**Salida:**

El programa debe crear un archivo de salida llamado "salida.out" que muestra cuántos cortes se debe realizar a la lámina con tal de dejar todas las secciones cortadas con el mismo tipo

Formato de salida:

Salida1.out	Salida2.out	Salida3.out
Se necesitan 7 cortes	Se necesita 1 corte	Se necesitan 3 cortes

**Entregas:**

La parte presencial se debe entregar el día **13/01/2019 || 23:55** por link en UsachVirtual. El presencial debe incluir idea, pseudocódigo y traza, se debe entregar en formato **PDF**.

Para este laboratorio, el informe no es exigencia (no es necesario presentarlo).

El código debe ser entregados el día **25/01/2019** a las **23:55** en el link de UsachVirtual.

El formato de entrega debe ser: rutCompleto\_n° lab.zip.

Ejemplo: 12345678K\_3.zip

El descuento por atraso de la entrega del código es : -0.3 décimas de la nota cada 30 min