

Problemas a resolver para generar habilidad algorítmica y programación.

1. Dado un arreglo de 3x3 el cual se encuentra lleno, determinar si una persona a ganado un juego o no. Las 'X' se representan con un uno, las 'O' con un 0 y los espacios en blanco con -1

$$\begin{pmatrix} 1 & 0 & 0 \\ 1 & 0 & 1 \\ 1 & 0 & 0 \end{pmatrix} \begin{pmatrix} 1 & 1 & -1 \\ 0 & 0 & 0 \\ -1 & -1 & -1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 1 & 1 & 0 \end{pmatrix}$$

En los primeros dos casos el programa debe decir que ganó la X y la O respectivamente. En el tercer caso debe decir que nadie ganó.

2. Dado un tablero de ajedrez de 4 x 4 dado, verificar si un caballo puede comerse a alguna de las figuras dadas e indicar cual. Un 0 representa un espacio en blanco, un -1 el caballo, un 1 un peón, un 2 una torre, un 3 un alfil, un 4 una Reyna y un 5 un Rey. Se ignoran las figuras del mismo equipo. El movimiento de un caballo es en forma de L, el movimiento es vertical-horizontal u horizontal-vertical. El primer movimiento avanza dos casillas y el segundo siempre es una casilla. El movimiento horizontal puede ser de izquierda o derecha y el vertical de arriba abajo.

$$\begin{pmatrix} 0 & 0 & 2 & 0 \\ 1 & 0 & 0 & 0 \\ 1 & -1 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 3 \\ 1 & -1 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} 0 & 0 & 0 & 5 \\ 2 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 4 & 0 & 0 & 0 \end{pmatrix}$$

Si se dan las siguientes entradas el programa debe decir en el primer caso que el caballo puede comerse a una torre, en el segundo caso que puede comerse a un peón y a un alfil, y en el tercer caso que no puede comerse a nadie.

3. El juego de acertijos de figuras se gana si se ordenan cada uno de las casillas de las figuras en el orden correcto. Cada figura tiene un número que lo representa, la casilla vacía se representa con 0. Realizar un programa que determine si un usuario a resuelto el ejercicio. La matriz de juego se da al programa (4x4).

$$\begin{pmatrix} 0 & 4 & 8 & 12 \\ 1 & 5 & 9 & 13 \\ 2 & 6 & 10 & 14 \\ 3 & 7 & 11 & 15 \end{pmatrix} \begin{pmatrix} 1 & 4 & 11 & 12 \\ 3 & 15 & 10 & 8 \\ 7 & 9 & 6 & 2 \\ 13 & 7 & 5 & 0 \end{pmatrix}$$

Para el primer caso el programa deberá indicar que el usuario ganó. Para los demás casos debe indicar que no ganó.

4. El juego del buscaminas es un juego muy popular. Dicho juego se juega introduciendo una posición que no sea en donde se encuentran las minas. Se gana el juego cuando se cubren todas las posiciones sin tocar ninguna mina. Sea un campo minado de tamaño 4x4 con las minas ya puestas, realizar un programa que permita a un usuario jugar el buscaminas. Las posiciones se introducen a través del teclado mediante coordenadas, donde la primera posición indica la fila y la segunda indica las columnas. Sea la matriz que representa el campo minado la siguiente:

$$\begin{pmatrix} 0 & 0 & 0 & 0 \\ -1 & 0 & 0 & -1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 \end{pmatrix}$$

Las minas están representadas con -1, las casillas sin visitar con 0 y las visitadas con uno. El usuario introduce la primera coordenada: [2,3] que indica la casilla que se ubica en la tercera fila y cuarta columna (recordar que los subíndices inician en 0 para la primera posición) por lo que la matriz quedaría de la siguiente forma:

$$\begin{pmatrix} 0 & 0 & 0 & 0 \\ -1 & 0 & 0 & -1 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & -1 & 0 \end{pmatrix}$$

Si el usuario introdujera una coordenada: [3,2] el usuario perdería el juego por que tocó una mina. Para ganar el usuario debe tener una matriz como se muestra a continuación. La validación se debe realizar en cada movimiento del usuario.

$$\begin{pmatrix} 1 & 1 & 1 & 1 \\ -1 & 1 & 1 & -1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & 1 \end{pmatrix}$$

5. El memorama es un juego de gran habilidad mental, consiste en ver por unos momentos una matriz que contiene figuras dispuestas en distinto orden. Las figuras están en pares. Se considera un tablero de 4x4. Realizar un programa que muestre al usuario una matriz dada de figuras por 3 segundos, la oculte y pide al usuario que trate de acordarse del orden de las figuras. El programa debe indicar cuantas figuras coincidieron, si las 8 figuras coincidieron se debe indicar que el usuario es un master, si hizo de 6 a 7 figuras debe indicar que su capacidad está arriba del promedio de 5 a 3, que su capacidad está en el promedio, 1 y 2 que está debajo del promedio y sino le atinó a ninguna debe indicar que realmente es un milagro que pueda prender la computadora. Suponga la siguiente matriz de juego:

$$\begin{pmatrix} 1 & 5 & 4 & 8 \\ 2 & 7 & 8 & 5 \\ 3 & 4 & 6 & 3 \\ 6 & 7 & 2 & 1 \end{pmatrix}$$

Si el usuario introdujera las siguientes matrices:

$$\begin{pmatrix} 1 & 5 & 3 & 8 \\ 2 & 7 & 8 & 5 \\ 4 & 4 & 6 & 3 \\ 2 & 6 & 7 & 1 \end{pmatrix} \begin{pmatrix} 1 & 5 & 4 & 8 \\ 2 & 7 & 8 & 5 \\ 3 & 4 & 6 & 3 \\ 6 & 7 & 2 & 1 \end{pmatrix}$$

El programa debe indicar para la primera matriz : 3 aciertos, “tu capacidad está en el promedio”; para la segunda matriz debe indicar: 8 aciertos, “Eres un master”.

6. El juego de Tetris es muy popular. El juego consiste en ir poniendo piezas para completar líneas. Cada que se completa una línea la cual puede ser sencilla, doble, triple o cuádruple se debe recorrer las figuras que no forman parte de la línea una posición. Hacer un programa que dada una matriz de 8x8 elementos, indique cuantos tetris se hicieron y devuelva las nuevas posiciones en el arreglo. Sea la matriz:

$$\begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 0 & 1 & 0 & 1 & 1 \end{pmatrix}$$

Las casillas marcadas con 0 indican espacios vacíos, los 1 indican lugares llenos. El programa debe mostrar la siguiente salida: “Tetris de dos líneas” y la matriz queda de la siguiente forma:

$$\begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 1 & 1 \end{pmatrix}$$

Los unos no se bajan en caso de que queden arriba de un 0.

7. La mayoría de las imágenes se llegan a representar con matrices de colores, en donde cada color está representando por un número e indica un punto en la pantalla. Un efecto de los programas visuales es el video inverso, el consiste en intercambiar los colores por su complemento. Sea un mapa de bits de 4x4 a 16 colores dado, encontrar su mapa de bits en video inverso. Si la imagen es la siguiente, la imagen en video inverso es la siguiente:

$$\begin{pmatrix} 1 & 5 & 4 & 12 \\ 2 & 1 & 8 & 5 \\ 3 & 15 & 6 & 3 \\ 6 & 1 & 12 & 1 \end{pmatrix}$$

La matriz de video inverso es la siguiente:

$$\begin{pmatrix} 14 & 10 & 11 & 3 \\ 13 & 14 & 7 & 10 \\ 12 & 0 & 9 & 12 \\ 9 & 14 & 3 & 14 \end{pmatrix}$$

Al tener 16 valores se tienen números de 0-15, al ser 15 el número más grande su complemento es 0, ya que no necesita de ningún valor para ser el más grande, el complemento de 4 es 11, ya que requiere de 11 números para alcanzar el máximo.

8. La transpuesta de una matriz consiste en convertir las filas de una matriz por columnas de la otra matriz (se obtiene el mismo resultado si las columnas de la matriz original pasan a ser las filas de la matriz nueva). Hacer un programa que dada una matriz obtenga su transpuesta. Si la matriz es:

$$\begin{pmatrix} 1 & 5 & 4 & 8 \\ 2 & 7 & 8 & 5 \\ 3 & 4 & 6 & 3 \\ 6 & 7 & 2 & 1 \end{pmatrix}$$

La transpuesta de dicha matriz es:

$$\begin{pmatrix} 1 & 2 & 3 & 6 \\ 5 & 7 & 4 & 7 \\ 4 & 8 & 6 & 2 \\ 8 & 5 & 3 & 1 \end{pmatrix}$$

La transpuesta de una matriz en una imagen nos sirve para representar un giro de 90 grados hacia la derecha (sentido de las manecillas de reloj). Esto nos permite rotar una imagen.

9. La matriz identidad es aquella cuyos elementos en la diagonal principal son unos y los demás elementos son 0. Hacer un programa que dado una matriz de 4x4 determine si se trata de una matriz identidad. Sean las siguientes matrices:

$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 5 & 4 & 8 \\ 2 & 7 & 8 & 5 \\ 3 & 4 & 6 & 3 \\ 6 & 7 & 2 & 1 \end{pmatrix} \begin{pmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{pmatrix}$$

El programa debe indicar que la primera matriz es transpuesta y que las restantes no lo son.

10. Otros de los efectos visuales de las herramientas gráficas consiste en reflejar una imagen (ponerla frente a un espejo) este efecto se logra intercambiando las columnas del mapa de bits. La primera columna pasa a ser la última y la última la primera intercambiando los demás valores. Hacer un programa que permita reflejar una imagen horizontalmente. Se la imagen:

$$\begin{pmatrix} 1 & 5 & 4 & 8 \\ 2 & 7 & 6 & 5 \\ 3 & 4 & 6 & 3 \\ 6 & 7 & 2 & 1 \end{pmatrix}$$

La imagen reflejada debe quedar de la siguiente forma:

$$\begin{pmatrix} 8 & 4 & 5 & 1 \\ 5 & 6 & 7 & 2 \\ 3 & 6 & 4 & 3 \\ 1 & 2 & 7 & 6 \end{pmatrix}$$

11. Si se tiene una imagen y se invierten los renglones que conforman el mapa de bits se obtiene una figura de cabeza. Hacer un programa que dada una imagen de entrada representada como una matriz, obtenga la matriz de cabeza que representa la nueva imagen. Si la matriz de entrada es:

$$\begin{pmatrix} 1 & 5 & 4 & 8 \\ 2 & 7 & 6 & 5 \\ 3 & 4 & 6 & 3 \\ 6 & 7 & 2 & 1 \end{pmatrix}$$

La imagen de cabeza debe quedar de la siguiente forma:

$$\begin{pmatrix} 6 & 7 & 2 & 1 \\ 3 & 4 & 6 & 3 \\ 2 & 7 & 6 & 5 \\ 1 & 5 & 4 & 8 \end{pmatrix}$$

12. Una matriz triangular es aquella cuyo todos los elementos de la matriz arriba o abajo de sus diagonales principal o secundaria son del mismo valor. Realizar un programa que permita determinar si una matriz es triangular o no. Sean las siguientes matrices cuadradas de 4x4:

$$\begin{pmatrix} 1 & 5 & 4 & 8 \\ 1 & 1 & 8 & 5 \\ 1 & 1 & 1 & 3 \\ 1 & 1 & 1 & 1 \end{pmatrix} \begin{pmatrix} 1 & 5 & 4 & 8 \\ 2 & 7 & 8 & 5 \\ 3 & 4 & 6 & 3 \\ 6 & 7 & 2 & 1 \end{pmatrix} \begin{pmatrix} 5 & 5 & 5 & 5 \\ 2 & 5 & 5 & 5 \\ 3 & 4 & 5 & 5 \\ 6 & 7 & 2 & 5 \end{pmatrix} \begin{pmatrix} 1 & 5 & 4 & 0 \\ 2 & 7 & 0 & 0 \\ 3 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}$$

El programa debe indicar que las matrices 1,3 y 4 son matrices triangulares y la matriz 2 no lo es.

13. Para obtener una franquicia de comida rápida no debe de existir más de 4 franquicias (incluyendo la nuestra) en la región. Las franquicias están representadas en una matriz de 4x4 la cual indica cada una de las zonas que en está divida una región. Realizar un programa que dado un mapa representado por una matriz de 4x4 con la ubicación de las franquicias y de la nueva franquicia determine si es viable la construcción de dicha franquicia. Sean los siguientes mapas:

$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & -1 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & -1 & 1 & 0 \\ 1 & 0 & 1 & 0 \end{pmatrix}$$

La primera matriz es válida ya que solo hay dos franquicias (una franquicia adyacente y la nuestra), la segunda también es válida por que hay 4 franquicias con la nuestra; mientras que el tercer mapa es inválido ya que hay 5 franquicias adyacentes.

14. Un número en octal es aquel que tiene como base al número 8, tiene 8 dígitos: del 0 al 7. Para convertir un número decimal a octal o a cualquier base se debe dividir el número entre esa base. El residuo se va almacenando en un arreglo y la parte entera de la división se vuelve a dividir entre la base hasta que la parte entera sea 0 almacenando los residuos en un arreglo. Una vez terminado el proceso se lee el arreglo al revés y ese es el número en la nueva base. Realizar un programa que permita convertir cualquier número en decimal a octal. Por ejemplo, el usuario introduce el número decimal 259 y se quiere convertir a octal. El proceso quedaría de la siguiente forma:

$$259/8 = 32 \quad 259\%8 = 3 \quad \text{Arreglo: } 3$$

$$32/8 = 4 \quad 32\%8 = 0 \quad \text{Arreglo: } 3 \ 0$$

$$4/8 = 0 \quad 4\%8 = 4 \quad \text{Arreglo: } 304 \quad \text{Número en octal: } 403$$

15. La mayoría de los PDA's y algunos otros dispositivos móviles por su diminuto tamaño no cuentan con un teclado. Para introducir información al dispositivo se necesitan de lápiz para escribir. Muchos dispositivos utilizan tecnologías de escritura predefinidas como: Grafiti. Dichos sistemas representan los caracteres en forma especial, por ejemplo la A. Se representaría en una imagen de mapa de bits de 6x6 de la siguiente forma:

$$\begin{pmatrix} 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 \end{pmatrix}$$

Se debe realizar un programa que indique si se trata de una letra A o no. La escritura de la letra del usuario se da como entrada. La representación de los puntos de entrada se da con un -1 a diferencia del patrón que se da con un 1, los puntos vacíos se dan con un 0. Para comprobar si la escritura es similar se hace una suma de matrices, si la matriz resultante es toda 0, el carácter se dibujo de manera precisa. Desafortunadamente ninguna tiene una caligrafía de 10. El sistema dara por bueno si la matriz resultante contiene cuando muchos dos 1. De lo contrario la dará por incorrecta. Ejemplo, si las matrices de las letras de los usuarios fueran:

$$\begin{pmatrix} 0 & -1 & -1 & 0 \\ 0 & -1 & -1 & 0 \\ -1 & 0 & 0 & -1 \\ -1 & 0 & 0 & -1 \end{pmatrix} \begin{pmatrix} 0 & -1 & 0 & 0 \\ 0 & -1 & -1 & 0 \\ 0 & 0 & 0 & -1 \\ -1 & 0 & 0 & -1 \end{pmatrix} \begin{pmatrix} -1 & -1 & 0 & 0 \\ -1 & -1 & -1 & 0 \\ -1 & -1 & 0 & 0 \\ -1 & 0 & 0 & 0 \end{pmatrix}$$

Las matrices resultantes quedarían:

$$\begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} -1 & 0 & 1 & 0 \\ -1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 1 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Por lo que las primeras dos escrituras se darían como As válidas. Ya que no tienen ningún 1 y dos 1 respectivamente. La tercera se da como inválida ya que tiene tres 1.

16. La multiplicación de matrices es una de las operaciones más utilizada en arreglos de dos dimensiones ($C = A * B$, donde A, B y C son matrices se necesita que A sea de tamaño ixk y B de tamaño kxj para que C sea de tamaño ixj). La multiplicación se realiza tomando como pivote el renglón de la matriz A que indica la Matriz C contra todos los elementos de la columna indicada por la matriz C. La formula sería:

$$C_{ij} = \sum_{k=1}^n A_{ik} * B_{kj}$$

Realizar un programa que permita multiplicar dos matrices dadas (se asume que son de tamaño correcto). Sean A y B las siguientes matrices:

$$\begin{pmatrix} 1 & 4 & 1 & 0 \\ 0 & 0 & 1 & 2 \\ 3 & 5 & 1 & 2 \end{pmatrix} \begin{pmatrix} 1 & 0 \\ 0 & 1 \\ 3 & 2 \\ -1 & 0 \end{pmatrix}$$

La matriz resultante C debe ser de tamaño 3x2, los procesos intermedios se calculan si:

$$\begin{pmatrix} 1*1+4*0+1*3+0*-1 & 1*0+4*1+1*2+0*0 \\ 0*1+0*0+1*3+2*-1 & 0*0+0*1+1*2+2*0 \\ 3*1+5*0+1*3+2*-1 & 3*0+5*1+1*2+2*0 \end{pmatrix} \begin{pmatrix} 1+0+3+0 & 0+4+2+0 \\ 0+0+3-2 & 0+0+2+0 \\ 3+0+3-2 & 0+5+2+0 \end{pmatrix}$$

$$\begin{pmatrix} 4 & 6 \\ 1 & 2 \\ 4 & 7 \end{pmatrix}$$

17. Dado un arreglo de 4x4 encontrar el elemento mayor y menor del arreglo. Sea la siguiente matriz:

$$\begin{pmatrix} -1 & 0 & 21 & 9 \\ 18 & 25 & 0 & 0 \\ 0 & 14 & -10 & 7 \\ 11 & 0 & 2 & 5 \end{pmatrix}$$

El elemento mayor es 25 y el menor es el 10

18. Dado un arreglo de mxn elementos encontrar la media y la moda por medio de un programa. La media es el promedio aritmético de cada uno de los elementos del arreglo, la moda es el número que más veces se repite. Por ejemplo si se da la siguiente matriz de 4x4:

$$\begin{pmatrix} 1 & 5 & 4 & 7 \\ 0 & 1 & 3 & 6 \\ 5 & 3 & 1 & 2 \\ 0 & 2 & 2 & 1 \end{pmatrix}$$

La media es: $(1+5+4+7+0+1+3+6+5+3+1+2+0+2+2+1)/16=43/16 = 2.68$.

Para calcular la moda se necesita saber la frecuencia de un número, esta se obtiene listando cada número y contándolo las veces que aparezca:

Num Freq

- 1 4 Para realizar la tabla, se va listando cada número que aparece con la tabla.
- 5 2 si el número no está en la tabla, se anota el número y se procede a buscarlo
- 4 1 alrededor de toda la matriz el número de ocasiones que se tiene.
- 7 1 Si el número ya está en la tabla se ignora.

0	2	El total de la frecuencia debe ser igual al número de elementos del arreglo.
3	2	La moda es el elemento que tiene mayor frecuencia, en este caso la moda es el 1
6	1	
2	3	
Total	16	

19. En muchos casos es necesario ordenar los elementos de un arreglo. Esta ordenación puede ser descendente (de mayor a menor) o ascendente (de menor a mayor). La manera más fácil de hacerlo es a través de la creación de otro arreglo y copiando los valores en el nuevo arreglo ya ordenados. Para ordenarlos se busca el elemento mayor del arreglo original, se copia su valor en el nuevo arreglo y se marca con un número inválido. El proceso continua hasta que el arreglo original queda todo marcado. Realizar un programa que ordene un vector de 5 elementos. Si el vector fuese:

A(23 11 15 37 8)

El arreglo auxiliar sería B(-1 -1 -1 -1 -1) donde -1 indica una celda vacía.

La primera iteración del programa encontraría que el número mayor es 37, por lo que lo marcamos con -1 y copiamos su valor en el arreglo auxiliar:

A (23 11 15 -1 8), B (37 -1 -1 -1 -1)

Para la siguiente iteración el número mayor es 23 y se repite el proceso hasta que el arreglo A esté vacío y B lleno: A(-1 -1 -1 -1 -1), B (37 23 15 11 8)

20. Otra forma de ordenar un arreglo es a través del método de la burbuja en donde se van comparando elementos adyacentes y se intercambian en caso de ser mayor o menor dependiendo del caso. Realizar un programa que pueda ordenar un vector de 5 elementos de forma descendente. Sea el siguiente arreglo:

A(5, 18, 11, 23, 9), en la primera pasada se toma como base el primer elemento del arreglo y se compara con el valor siguiente, en este caso 18. Como se está ordenando de manera descendente primero van los elementos más grandes por lo que se intercambian los elementos quedando así: A(18, 5, 11, 23, 9) se toma como base el segundo elemento y se compara con el que sigue quedando: A(18, 11, 5, 23, 9), luego A(18, 11, 23, 5, 9) y finalmente A(18, 11, 23, 9, 5), por lo que en la primera pasada el elemento mas pequeño cayó hasta el fondo. Para la segunda pasada se sigue el mismo procedimiento pero el algoritmo para en la penúltima posición ya que la última ya está ordenada. Las iteraciones en la segunda pasada son: A(18, 11, 23, 9, 5), A(18, 23, 11, 9, 5),

A(18, 23, 11, 9, 5). Para la tercera repetición el límite es el antepenúltimo lugar ya que los de abajo ya están ordenados. Al final el resultado debe quedar: A(23, 18, 11, 9, 5).

21. Otra forma de ordenar un arreglo es a través del proceso de inserción. En este algoritmo se escoge el mayor de los n elementos de un arreglo y se inserta en la posición que le corresponde. Dicha posición va aumentando hasta que se llegue al final del arreglo ordenado. Sea un vector A de elementos: A(5,18, 11, 23, 9).

El proceso de ordenamiento es el siguiente: se encuentra el mayor de los n elementos y se coloca en el primer lugar intercambiando su contenido con el elemento de dicha posición, el mayor es 23 por lo que se intercambia con 5 quedando: A(23, 18, 11, 5, 9). Para la segunda pasada, se encuentra el número mayor a partir de la posición dos ya que la primera posición ya está ordenada, el elemento mayor fue 18 el cual ya se encontraba en su posición, el algoritmo se repite hasta que se ordena el último elemento: A(23,18, 11, 9, 5).

22. Otra manera de ordenar un arreglo es a través del método de selección el cual consiste en tomar un elemento como pivote comparándolo con los demás, si es mayor se intercambia hasta ordenarlo. Hacer un programa que ordene un arreglo por inserción. El vector es de 5 elementos y ya tiene valores. Sea A(5, 18, 11, 23, 9), en la primera pasada se toma el primer número y se compara con todos: 5-18, como es más grande el 18 se intercambia por lo que el vector queda A(18, 5, 11, 23, 9), A(18, 5, 11, 23, 9), A(23, 5, 11, 18, 9) quedando finalmente A(23, 5, 11, 18, 9). Nótese que en la primera pasada el elemento mayor queda en la cima. En la segunda pasada, el elemento que se toma de pivote es el segundo elemento y se compara con los elementos restantes: A(23, 5, 11, 18, 9), A (23, 11, 5, 18, 9), A(23, 18, 5, 11, 9), A(23, 18, 5, 11, 9). El proceso se repite hasta que el pivote sea el último elemento, quedando el arreglo ordenado: A(23, 18, 11, 9, 5).

23. El máximo común divisor de dos números es el número más grande que puede dividir a dos números. Por ejemplo: el $\text{mcd}(6,15)=3$, $\text{mcd}(10, 21)=1$. Cuando el mcd de dos números es uno se dice que esos dos números son primos entre sí. La forma de calcularlo es encontrar el número más pequeño de dos números, y dividir los números entre dicho número, hasta que sea divisible (desde número menor hasta 1 en el peor de los casos). Realizar un programa que permita leer dos números y encuentre el mcd de dichos números. Si el usuario teclea 15 y 6. El mcd se calcula así: se encuentra el número más pequeño: 6 y se empieza a dividir $6\%6=0$ y $15\%6=3$ por lo que no divide a los dos, luego el divisor se decrementa en 1, por lo que ahora las divisiones son: $6\%5=1$ y $15\%5=0$, sigue sin dividir a los dos; luego se procede a $6\%4=2$ y $15\%4=3$, luego $6\%3=0$ y $15\%3=0$, dividiendo a ambos números por lo que el mcd es 3.

24. Una forma más rápida para calcular el mcd de dos números es utilizando el algoritmo de Euclides. El cual consiste en dividir los dos números m y n , donde $m < n$. Mientras $m > 0$ se guarda el valor de m en una variable temporal, a m se le asigna el valor de dividir $n \% m$, a n se le asigna la variable temporal. Al final del ciclo se devuelve el valor de n el cual representa el mcd. Realizar un programa que implemente el algoritmo de Euclides para calcular el mcd. Ejemplo: si el usuario introduce los números 15 y 6, m corresponde al más pequeño por lo que $m=6$ y por lo tanto $n=15$. Como $m > 0$ entramos a un ciclo en donde asignamos el valor de m a una variable auxiliar aux , por lo que $aux=6$, después $m= n \% m$, por lo que $m=15 \% 6$, por lo que $m=3$, después se asigna a n el valor de aux , por lo que $n=6$. Después se vuelve a realizar esas acciones dado que $m > 0$ y se repiten los pasos anteriores, se guarda el valor de m en aux , por lo que $aux=3$, $m=n \% m$, por lo que $m=6 \% 3$, $m=0$, se copia el valor de aux a n , por lo que $n=3$. Se comprueba la condición del ciclo nuevamente, como m no es mayor que 0, el ciclo termina y se devuelve n como el mcd, en este caso $n=3$, el $mcd(15,6)$ es 3.

25. Se tiene un número de 4 dígitos que representa un código de un producto. Dicho código de un producto se representa con 5 dígitos donde los primeros 4 dígitos es el código del producto y el dígito extra se calcula de la suma interna de dichos números. Sea por ejemplo el código de un producto el número 4257. El número se separa en cada uno de sus dígitos respectivos y se suman por lo que el resultado queda: $4+2+5+7=18$. 18 es un número de dos dígitos por lo que se vuelve a repetir el proceso de suma, quedando: $1+8=9$. Como 9 es un número de un dígito, ese número complementa el código de un producto. Realizar un programa que obtenga dicho número de control.

26. Realizar un programa que dado un número de bits determine el número de KB (Kilo Bytes), MB y GB. Un byte son 8 bits. Un KB son 1024 bytes, un MB son 1024 KB, un GB son 1024 MB. Por ejemplo si el usuario introduce la cantidad: 12,896,521 bits el programa debe devolver:

1,612,065 (1612065.18) bytes

1,574 (1574.28) KB

1 (1.53) MB

0 (0.0015) GB

En donde se deja la parte entera de la división.