|  |  |
| --- | --- |
|  |  |

|  |  |
| --- | --- |
|  | 1. Dado un número introducido por teclado, comprobar si el número es divisible por 3. Un número es divisible por 3 cuando la suma de sus dígitos lo es. Realizar comprobación de entrada y un menú para poder comprobar varios números y una opción para poder salir del programa. |
|  |

|  |  |
| --- | --- |
|  | 1. Dados dos números introducidos por teclado, averiguar el máximo común divisor de esos dos números e imprimirlo en pantalla. |
|  |

|  |  |
| --- | --- |
|  | 1. Combinaciones sin repetición. Se llama combinación sin repetición de p elementos, con p < n, elegidos entre los n, a cualquier subconjunto de p elementos distintos del conjunto.   El número de combinaciones sin repetición de p elementos elegidos entre los n se denota habitualmente:  ( n! representa el factorial de un número: n! = n\*(n-1)\*(n-2)\* … 2\*1 )  Realizar un programa para calcular el número de combinaciones sin repetición de p elementos cogidos de entre n con 2 <= p < n < 8. Los números se pedirán en el momento de la ejecución y se introducirán por teclado. |
|  |

|  |  |
| --- | --- |
|  | 1. Dadas dos matrices de dimensiones (2 x 2), multiplicar dichas matrices y mostrar por pantalla el resultado. |
|  |

|  |  |
| --- | --- |
|  | 1. Tenemos una cantidad muy elevada de ficheros dentro de un directorio Windows relativos a operaciones en departamentos en una empresa en fechas determinadas. El nombre de los ficheros tiene el siguiente formato:   *nombreDepartamento\_mes-dia-2020.doc*  El nombre del directorio es documentación2020 y está en el directorio raíz.  Se pide realizar un programa para reorganizar dichos ficheros en función del contenido de su nombre en diferentes carpetas por departamento, mes y dia.  Es decir, se obtendrá una estructura de directorios tal y como se muestra a continuación para cada combinación de departamento, mes y dia distinto que se detecte:  nombreDepartamento1|  |  |mes1 |  | | dia1 |  | | |documento1.doc  | | | ….  | | | documentoN.doc  | | dia2 |  | | |documento1.doc  | | | ….  | | | documentoN.doc  | | …. |  | | |documento1.doc  | | | ….  | | | documentoN.doc  | | diaN|  | | |documento1.doc  | | | ….  | | | documentoN.doc  Al final tendremos también un fichero de dato serializado con objetos de información que contendrá los movimientos (origen, destino, fecha y hora del movimiento). Sabiendo que la fecha y la hora se puede obtener con la clase Date() y métodos asociados. |
|  |

Principio del formulario