



Fundamentos de Programación
Ingeniería Informática en Sistemas de Información
EJERCICIOS

1. Algoritmo que obtenga la hora, minutos y segundos de una hora expresada en segundos.
2. Algoritmo que convierta una temperatura expresada en Fahrenheit a Celsius.
3. Algoritmo que calcule la cifra i-ésima de un número natural n.
4. Algoritmo que dados tres números a, b y c, los presente en pantalla de menor a mayor.
5. Algoritmo que dados tres números a, b y c, representando los coeficientes reales de una ecuación de segundo grado, obtenga las soluciones reales (simples o dobles).
6. Algoritmo que dados tres números d, m y a, representando el día, mes y año de una fecha, indique si la fecha es válida, considerando los años bisiesto. (Son años bisiestos los múltiplos de 4, excepto los múltiplos de 100 que no son múltiplos de 400).
7. Algoritmo que devuelva la suma de los n primeros números naturales.
8. Algoritmo que devuelva la suma de los cuadrados de los n primeros números naturales.
9. Algoritmo que devuelva la suma de los cuadrados de los números comprendidos entre a y b.
10. Algoritmo que devuelva el número de cifras de un número n.
11. Algoritmo que calcule el número de Fibonacci de un natural n. (Ej: la serie de números de Fibonacci es 1, 1, 2, 3, 5, 8, 13, 21, 34, ..., donde un número es obtenido a partir de la suma de los dos precedentes, exceptos los dos primeros que valen 1; si n=6 entonces f=13).
12. Algoritmo que dado un número de entrada indicando el número de números que se van a leer desde el teclado, calcule la media geométrica de ellos y la presente en pantalla.
13. Algoritmo que calcule el factorial de un número n.
14. Algoritmo que calcule el producto de dos números a y b. (Sin utilizar el operador de multiplicación *).
15. Algoritmo que calcule la potencia de n-ésima de un número x. (Sin utilizar la función potencia de math.h).
16. Algoritmo que visualice el valor máximo y mínimo de un conjunto de valores introducidos por teclado. La lectura concluirá cuando se hayan leído un máximo de cien números o algún número negativo.
17. Algoritmo que presente en pantalla los primeros n múltiplos de tres.
18. Algoritmo que presente en pantalla los múltiplos de tres comprendidos entre a y b.
19. Algoritmo que indique si un número n es cuadrado perfecto (sin utilizar la operación raíz). Un número es cuadrado perfecto si existe un número natural cuyo cuadrado es igual a n.
20. Algoritmo que indique si un número n es primo.
21. Algoritmo que indique si un número es perfecto. (Un número es perfecto si es suma de sus divisores, comenzando por 1 y excluyendo el propio número).
22. Algoritmo que indique si un número n es capicúa (se lee igual de izquierda a derecha que viceversa).
23. Realizar un algoritmo que muestre por pantalla la cifra que resulta de aplicar a un número n el siguiente proceso: a) sumar el valor absoluto de todas las cifras del número; b) si el valor resultante tiene más de una cifra, volver a sumar todas sus cifras sucesivamente hasta obtener un valor de una única cifra.
24. Algoritmo que sume todos los elementos de un vector t de N reales.
25. Algoritmo que calcule el producto escalar de dos vectores t1 y t2 de N reales.
26. Algoritmo que calcule el máximo y el mínimo de un vector de N reales.
27. Algoritmo que invierta un vector t de N reales.
28. Algoritmo que sume los elementos de la fila k de una matriz m de NxN reales.
29. Algoritmo que localice un elemento x en un vector t de N reales, devolviendo la posición p y una variable lógica éxito. (Búsqueda lineal).
30. Algoritmo que indique si un vector t de N enteros está ordenado crecientemente.
31. Algoritmo que sume dos matrices m1 y m2, devolviendo el resultado en m3. m1 y m2 son matrices de NxN reales.
32. Algoritmo que localice un elemento x en una matriz t de N reales, devolviendo la posición (p,q) y una variable lógica éxito. (Búsqueda lineal).
33. Algoritmo que calcule la traspuesta de una matriz m de NxN reales respecto de la diagonal principal.
34. Algoritmo que cuente el número de apariciones de un número n en una matriz m de NxN reales.
35. Algoritmo que realice una rotación a la derecha en un vector t de N reales.
36. Algoritmo que sume los elementos de la columna k de una matriz m de NxN reales.
37. Algoritmo que sume los elementos múltiplos de tres de un vector t de N reales.
38. Algoritmo que sume los elementos que se encuentran en posiciones múltiplos de tres en un vector t de N reales..



39. Algoritmo que localice un elemento x en un vector t ordenado de N reales., devolviendo la posición p y una variable lógica *éxito*. (Búsqueda binaria).
40. Algoritmo que sume todos los elementos de una matriz m de $N \times N$ reales.
41. La diagonal principal de una matriz cuadrada $N \times N$ son los elementos que ocupan las posiciones $(1,1)$, $(2,2)$, $(3,3)$... (N,N) . Algoritmo que sume los elementos de la diagonal principal de una matriz m de $N \times N$ reales. (T1).
42. Se define la diagonal secundaria como los elementos que ocupan las posiciones $(1,N)$, $(2,N-1)$, $(3,N-2)$... $(N,1)$. Algoritmo que sume los elementos de la diagonal secundaria de una matriz m de $N \times N$ reales (T1).
43. Algoritmo que multiplique dos matrices $m1(M \times L)$ y $m2(L \times N)$, devolviendo el resultado en $m3(M \times N)$.
44. Algoritmo que ordene un vector t de N reales por el método de selección. (Se localiza el menor y se coloca en la primera posición; así, sucesivamente).
45. Se tiene una tabla bidimensional de $M \times N$ enteros comprendidos entre 1 y 10, ambos inclusive. Realizar un algoritmo que calcule el número que aparece con más frecuencia en la tabla.
46. Escriba una función tal que dada una matriz cuadrada devuelve cierto si ésta tiene dos elementos, uno en la diagonal principal y otro en la secundaria que sean iguales.
47. Algoritmo que calcule la traspuesta de una matriz m de $N \times N$ reales respecto de la diagonal secundaria.
48. Sea un vector de enteros, calcular la media de los valores absolutos de las diferencias entre dos cualesquiera elementos.
49. Algoritmo que ordene un vector t de N reales por el método de inserción.
50. Algoritmo que ordene un vector t de N reales por el método de la burbuja.
51. Dado un vector t de N reales, ¿en cuántas subsecuencias ordenadas crecientes se descompondría ese vector?
52. Algoritmo que realice k rotaciones a la derecha en una tabla.
53. Algoritmo que indique si existe un número n en una matriz m .
54. Algoritmo que sume todos los cuadrados perfectos que se encuentran en una tabla t .
55. Algoritmo que indique si existe un cuadrado perfecto en un vector.
56. Algoritmo que indique si existe una posición en el vector t tal que la suma de todos los elementos precedentes es igual a un número dado n .
57. Algoritmo que indique si existe alguna fila que suma s .
58. Algoritmo que indique si existe algún número de alguna fila que se encuentre en alguna otra.
59. Algoritmo que indique si existe en una matriz m de $N \times M$ enteros una fila que sume igual que una columna.
60. Algoritmo que indique si el producto de dos elementos de la diagonal principal de una matriz m de $N \times N$ reales se encuentra en alguna posición fuera de la diagonal principal.
61. Algoritmo que reciba una matriz m de $N \times N$ enteros rellenas con f (filas) y c (columnas), y retorne otra matriz que contenga en cada posición la media de los elementos válidos que lo rodean.
62. Se dice que una matriz es triangular superior cuando todos los elementos situados por debajo de la diagonal principal son iguales a 0. Diseñar una función que devuelva cierto o falso dependiendo de si la matriz m , de $N \times N$ enteros, que se pasa como parámetro es o no es triangular.
63. Escriba una función tal que dada una matriz cuadrada de $N \times N$ enteros, devuelva cierto si ésta tiene dos elementos, uno en la diagonal principal y otro en la secundaria que sean iguales. Por ejemplo,

1 2 3 4		1 2 3 4
2 3 7 5	devuelve falso	2 3 4 5 devuelve cierto
3 0 5 6		4 5 6 7
4 2 2 1		-1 4 5 5

64. Algoritmo que dada una matriz m de enteros, un número p (que representa a una fila) y un número q (que representa a una columna), devuelva el menor(p,q), es decir, devuelva la matriz que resulta de eliminar la fila p y la columna q .
65. Sea a un vector t de N enteros, ordenado ascendentemente, y un entero $k \geq 0$. Diseñar un algoritmo que calcule el número de pares (i,j) que verifiquen: $1 \leq i \leq j \leq N$ y $a(j) - a(i) \leq k$.
66. Diremos que un vector de enteros positivos es "coqueto" si el doble de la suma de cualesquiera tres números es mayor que el producto de cualesquiera dos números adyacentes. Realizar un algoritmo que indique si un vector de enteros t es o no coqueto.
67. El matemático Lagrange demostró que cualquier número positivo es suma de cuatro cuadrados. Escribir un programa que encuentre los cuatro cuadrados que componen un número n .
68. Diseñar una función que indique cuántos subvectores, de cualquier tamaño estrictamente positivo y menor o igual que N , formados por elementos consecutivos del vector t de N enteros, suman el mismo valor que algún elemento del vector.
69. Dado un número entero n devolver una vector de caracteres donde se expresa el número n en romano (sólo son posibles los caracteres I,V,X,L,C,D y M), sabiendo además que sólo las letras I,X y C pueden ir colocadas a la izquierda de otra de mayor valor. Es decir, I puede ir delante de V, X; X puede ir delante de C, L; y C puede ir delante de D, M. Además, las letras I, C, y M pueden aparecer repetidas juntas hasta tres veces y en total pueden aparecer cuatro veces (39=XXXIX; 49=XLIX). Los valores son I=1, V=5, X=10, L=50, C=100, D=500 y M=1000. Por comodidad sólo se permitirán números menores que MMMDCCCLXXXVIII.



70. Realizar un algoritmo que coloque en un vector de tamaño N los números del triángulo de Tartaglia hasta un nivel n determinado. Los valores pueden calcularse de dos formas: o bien utilizando los números combinatorios y la función factorial; o bien obteniendo los números a partir de los dos que se encuentran justo encima de éste, excepto los de las esquinas que valen siempre 1. Se piden dos algoritmos, cada uno siguiendo la política expresada en los dos métodos expuestos.
71. Algoritmo que indique si existe un subvector (de una fila) en alguna columna de una matriz m de NxN enteros.
72. Algoritmo que indique cuáles son las dos filas de una matriz cuyo producto escalar es mayor que el de las demás.
73. Llamaremos a una matriz "sumidero" si cualquier elemento que pertenezca a un rectángulo exterior es mayor que todos los elementos que pertenecen a un cuadrado más interior. (Imagínese la matriz como una serie de rectángulos concéntricos).
74. Sea una tabla bidimensional con las coordenadas de 100 puntos del plano. Se define centro del subespacio al punto cuya distancia media a los restantes es más pequeña. Realizar un algoritmo que devuelva la posición en la tabla del centro del subespacio.
75. Una matriz de MxN representa un laberinto de espejos. Cada posición de la matriz puede contener un 0, indicando que no hay espejo en dicha posición; un 1, indicando que existe un espejo situado a 45 grados respecto de la horizontal; o un -1, indicando que existe un espejo situado a 135 grados respecto de la horizontal. Los espejos reflejan la luz por las dos caras, de modo que un rayo incidente girará 90 grados cada vez que se encuentre un espejo. Se introduce un rayo de luz verticalmente hacia abajo por una de las columnas (c) de la fila superior de la matriz. Realizar un algoritmo que, partiendo de la columna por la que se lanza el rayo y la matriz que contiene la información de los espejos, devuelva las coordenadas (fila y columna) de la casilla por la que saldrá el rayo emergente.
76. Realizar un algoritmo que determine si existe en una matriz de NxN al menos dos submatrices de MxM, con $1 < M < N$, tales que las respectivas sumas de todos los elementos de cada submatriz sean iguales.
77. Dadas dos matrices de NxM reales, t1 y t2, escribir un algoritmo que indique si una matriz es imagen especular de la otra. Por ejemplo, dadas t1 y t2, dos matrices de 2x3, se afirma que una es imagen especular de la otra.

t1	t2
3 4 6	6 4 3
1 5 2	2 5 1

78. Escribir una función que rellene un vector t de N enteros con los primeros N números primos.

```
void rellenar_primos(long t[N])
```

79. Escribir un algoritmo que indique si un vector de N enteros contiene la secuencia de números de Fibonacci (1, 1, 2, 3, 5, 8, ...; en general, para $n > 1$, $f(n) = f(n-1) + f(n-2)$).

```
int es_fibonacci(long t[N])
```

80. Escriba un algoritmo tal que dado un fichero de enteros escriba las sumas parciales de los elementos del fichero que siendo iguales se encuentran en posiciones consecutivas. Por ejemplo, si los datos del fichero son

2 2 3 3 3 -1 -1 -1 5 6 6 2

el resultado del procedimiento debe ser imprimir los valores 4 9 -3 5 12 2.

Nota:

Se supone que el procedimiento recibe el fichero ya abierto y no vacío.