Problemas de Algorítmica (E.T.S.I.T.)

Javier Miranda, Francisco Guerra

20 de octubre de 1999

Resumen

Este documento proporciona una colección de problemas de la asignatura de Algorítmica, que se imparte en el primer curso de la Escuela Técnica Superior de Ingenieros de Telecomunicación.

Índice General

1 Identificadores y tipos de datos.

1. Indica qué identificadores son correctos en Ada.

```
MI_COCHE     Casa_3      "Contador"    Numero1

indice 1     Elemento&3  Procedure     %Total

34_Final     Codigo-1    Total_Suma     'x'
```

2. Indica de qué tipo de dato es cada uno de los siguientes valores Ada:

```
167    167.0   'x'  "True"  16.4e5  16e5

  7    '7'     "7"  False   1_000   0.0_5
```

2 Expresiones

1. Supongamos que en un programa hemos hecho las siguientes declaraciones:

```
I: Integer := 2;
J: Integer := 3;
X: Float   := 4.0;
Y: Float   := 5.0;
```

Evalúa las siguientes expresiones e indica el tipo de dato de cada resultado.

```
(a) I + J          (b) I + 5          (c) 2 + 3

(d)  X - 1.5       (e) 2.0 * 2.5      (f) Y / X

(g) J / I          (h) 14 rem 4       (i) J mod I

(j) X ** I         (k) Y ** (-1)      (l) I ** J

(m) I + J * 2      (n) X * Y ** 2     (o) abs X-Y

(p) X / Y * 2.0    (q) 2.0 * FLOAT(J)
```

2. Evalúa las siguientes expresiones lógicas:

```
(a) True and 10 > 8
(b) 5.0 >= 10.3 or 'a' > 'b'
(c) 3 not in 1..7
(d) I /= 0 and 14/I > 3  -- Supongamos que I vale 3
(e) 3 > 3 or "hola" /= "HOLA"
```

3. Escribe un programa que reciba desde teclado la siguiente información de un coche: Número de registro, kilometraje al principio y al final del año pasado, y total de litros de gasolina consumidos ese año. A partir de esta información el programa debe generar un informe con la siguiente información:

   - Número de registro:
   - Kilómetros recorridos el año pasado:
   - Combustible consumido:
   - Litros por kilómetro:

4. Escribe un programa que lea del teclado una cantidad en pesetas y nos diga el equivalente en euros.

5. Escribe un programa que lea del teclado una cantidad en euros y nos diga el equivalente en pesetas.

6. Escribe un programa que lea del teclado un peso en libras y onzas y nos diga el equivalente en kg. NOTA: 1 libra son 0.4536 Kg y hay 16 onzas en cada libra.

7. Escribe un programa que lea del teclado un peso en Kg y nos diga el equivalente en libras y onzas.

8. En Europa el consumo de un coche se mide en litros por kilómetro, mientras que en Inglaterra se mide en millas por galón. Escribe un programa que reciba una cantidad en litros por kilómetro y genere su equivalente en millas por galón. NOTA: 1 milla son 1109 Km; 1 galón son 3,785 litros.

9. Escribe un programa que calcule el volumen y el área de una esfera.

$$Volumen = \frac{4\pi r^3}{3} \qquad Area = 4\pi r^2$$

10. Escribe un programa que lea del teclado la longitud de dos lados de un triángulo rectángulo y nos diga el valor de la hipotenusa.

    NOTA: Para poder utilizar la función que calcula la raíz cuadrada de un número, que se llama SQRT, tenemos que utilizar un paquete genérico que se llama Numerics.Generic_Elementary_Function. La forma de utilizarlo es similar a Integer_Io. Por ejemplo:

```
with Ada.Numerics.Generic_Elementary_Functions;
procedure Ejemplo is
    package Math is new
            Ada.Numerics.Generic_Elementary_Functions(Float);
    use Math;
begin
    -- Ya puedo utilizar SQRT, SIN, COS, etc.
end Ejemplo;
```

11. Escribe un programa que reciba desde teclado los coeficientes de una ecuación de segundo grado y calcule las raíces.

12. Escribe un programa que reciba desde teclado las coordenadas de dos puntos (X_1, Y_1), (X_2, Y_2) y calcule la distancia entre ellos.

$$Distancia = \sqrt{(X_1 - X_2)^2 + (Y_1 - Y_2)^2}$$

3 Sentencias

3.1 if

1. Escribe un programa que lea dos números desde el teclado y nos diga cual de los números es el menor.

2. Escribe un programa que lea del teclado dos caracteres y nos los escriba ordenados (siguiendo el orden del estándar ASCII).

3. Escribe un programa que lea del teclado tres nombres y nos los escriba ordenados de menor a mayor (siguiendo el orden del estándar ASCII).

4. Escribe un programa que nos pida el precio de un artículo de una tienda y el número de artículos de ese tipo que hemos vendido a un cliente. Si el total de la factura es mayor a 1000 ptas, el programa hará un descuento de un diez por ciento a la factura.

5. Escribe un programa que lea de teclado una cantidad en pesetas y nos diga cuantos billetes de 10.000, 5.000, 1.000 y monedas de 500, 100, 25, 5 y 1 peseta debemos dar para pagar esa cantidad.

6. Utilizando la sentencia *if*, escribe un programa que lea una nota (un número entre 1 y 10) y genere en pantalla la nota correspondiente: suspenso, aprobado, bien, notable, sobresaliente.

3.2 case

1. Utilizando la sentencia *case*, escribe un programa que lea un número entre 1 y 12 que representa uno de los meses del año y genere en pantalla un mensaje que nos diga la estación del año asociada.

   NOTA: Para el programa, la primavera abarca los meses de Marzo, Abril y Mayo; el verano, los meses de Junio, Julio y Agosto; el otoño, los meses de Septiembre, Octubre y Noviembre; y el invierno, los meses de Diciembre, Enero y Febrero.

```
Ejemplo:

    Dime un mes (1..12): 12
    Es invierno
```

2. Utiliza la sentencia *case* para escribir un programa que lea una nota (un número entre 1 y 10) y genere en pantalla la nota correspondiente: suspenso, aprobado, bien, notable, sobresaliente.

3. Escribe una calculadora de sobremesa que sume, reste, multiplique y divida números de forma acumulativa. Cuando comienza su ejecución la calculadora contiene el valor 0.

```
Ejemplo de uso de la calculadora:

       Resultado = 0
      ----------------
       Operacion: +
       Dato     : 12
       Resultado = 12
      ----------------
       Operacion: -
       Dato     : 4
       Resultado = 8
      ----------------
       Operacion: *
       Dato     : 3
       Resultado = 24
      ----------------
       Operacion: /
       Dato     : 4
       Resultado = 6
```

3.3 loop

1. Escribe un programa que, mediante dos bucles anidados, genere en pantalla la siguiente figura:

```
+ + + + +
+ + + +
+ + +
+ +
+
```

2. Escribe un programa que nos muestre en pantalla las tablas de multiplicar.

3. Escribe un programa que, utilizando bucles, genere en pantalla la siguiente tabla de multiplicar. NOTA: El límite superior se especifica desde teclado.

```
1   2   3   4   5
2   4   6   8   10
3   6   9   12  15
4   8   12  16  20
5   10  15  20  25
```

4. Escribe un programa que nos muestre en pantalla una tabla con los cuadrados de los doce primeros números.

5. Escribe un programa que muestre en pantalla una tabla con los valores de los cuadrados y cubos de todos los números comprendidos entre N1 y N2. Los valores de N1 y N2 los lee el programa desde el teclado.

6. Escribe un programa que calcule la suma de todos los números enteros positivos comprendidos entre 1 y N. El valor de N lo lee el programa desde el teclado.

7. Escribe un programa que, mediante bucles, calcule la media arit'metica de todos los números comprendidos entre 1 y N. El valor de N lo lee el programa desde el teclado.

8. Escribe un programa que calcule la suma de todos los números impares menores que N. El valor de N lo lee el programa desde el teclado. NOTA: Escribe tres versiones del programa; una utilizando sólo bucles *loop*, otro utilizando sólo bucles *while* y otra utilizando sólo bucles *for*.

9. Escribe un programa que calcule el factorial de cualquier número entero positivo.

10. Escribe un programa que nos diga si un número cualquiera es primo o no.

11. Modifica el programa anterior para que calcule todos los números primos que hay entre 1 y el número que digamos nosotros por teclado.

12. Dada la siguiente función matemática: $f(x) = 3x^3 - 5x^2 + 2x - 20$.

    (a) Genera una tabla con todos los valores de $f(x)$ existentes en el intervalo entero -10..10.

    (b) Genera la tabla con los valores de $f(x)$ asociados a los valores de x comprendidos en el rango de números reales -2..2 en incrementos de 0,1.

13. Escribe un programa que lea N números desde teclado y nos diga cual de ellos es el menor y su posición.

14. Un banco nos da un interés del 3,5 por ciento al año. Supongamos que ponemos X pesetas fijas en el banco. Escribe un programa que calcule cuantos años tendremos que esperar hasta que la cuenta tenga 100.000 ptas más que cuando la creamos.

15. Escribe un programa que calcule cuál es el número entero más pequeño que comple que:

    $$\sum_{i=1}^{k} i^2 > n$$

    NOTA: El valor de n se lee desde el teclado.

16. El ayuntamiento de una ciudad ha hecho un estudio de su población y ha obtenido los siguientes resultados:

- En 1994 tenían 26.000 habitantes.
- Cada año nace un 0,7 por ciento de personas y muere un 0,6 por ciento.
- Cada año hay una inmigración de 300 personas y una emigración de 325 personas.

Escribe un programa que calcule el número de habitantes que habrá en cualquier año.

17. El alquiler de un coche nos cuesta 7.000 ptas por día más 50 ptas por cada kilómetro recorrido. Escribe un programa que reciba desde teclado un número de días y un número de kilómetros y nos diga cuanto nos va a costar el alquiler del coche.

18. Escribe un programa que reciba una hora desde el teclado (horas, minutos y segundos) y nos escriba en pantalla el valor de la hora en el siguiente segundo.

```
Ejemplo:
    Hora: 0:3:59
    Siguiente segundo: 0:4:00

    Hora: 2:59:59
    Siguiente segundo: 3:00:00
```

19. Modifica el programa anterior para que el programa se convierta en un reloj. NOTA: Para que el programa Ada espere un segundo debemos utilizar la sentencia *delay 1.0;*.

4 Procedimientos y funciones

1. Escribe una función que evalúe el signo de un número. La función debe retornar el valor 1 si el número es mayor que cero, 0 si el número es igual a cero y -1 si el número es menor que cero.

2. Escribe un programa que contenga:

   (a) Una función que calcule el máximo de dos números enteros.

   (b) Una función que recibe un carácter y nos diga si es una letra o no.

   (c) Una función que recibe un carácter y nos diga si es una letra o un número.

3. Escribe una función que reciba una letra mayúscula y devuelva su equivalente en minúscula.

4. Escribe la función *mcd()* que calcula el máximo común divisor de dos números enteros positivos utilizando el algoritmo de Euclides. El algoritmo de Euclides es el siguiente:

   (a) Realiza la división entera.
   (b) Si el resto es cero, el *mcd* es el divisor.
   (c) En caso contrario, copia el valor del divisor en el dividendo, el valor del resto en el divisor, y vuelve al primer paso.

5. Escribe una función que, mediante bucles, calcule la potencia n de un número (X^n).

6. Escribe una función que, mediante bucles, calcule el factorial de un número ($X!$).

7. Cuando un lenguaje de programación no tiene las funciones matemáticas, podemos utilizar el desarrollo en serie de McLaurin para calcularlas. Por ejemplo, la función *seno()* podemos evaluarla calculando la siguiente serie:

$$sen(X) = X - \frac{X^3}{3!} + \frac{X^5}{5!} - \frac{X^7}{7!} + \frac{X^9}{9!} + ...$$

   Utilizando las funciones de los dos problemas anteriores (X^n y $X!$), escribe una función que calcule *sen(X)* realizando el desarrollo en serie de McLaurin hasta que el valor de $\frac{X^n}{n!}$ sea menor que 10^{-5}.

8. Modifica el programa anterior para que también evalúe e^x realizando la siguiente serie mientras no se alcance una precisión de 10^{-4}.

$$E(X) = 1 + X + \frac{X^2}{2!} + ... + \frac{X^n}{n!}$$

9. Escribe una función que calcule la suma de los 10 elementos de un vector. NOTA: La función debe trabajar correctamente independientemente del rango del índice del vector.

10. Escribe una función que calcule la suma de todos los elementos de un vector, independientemente del tamaño del vector y del rango del índice del vector.

11. Escribe una función que recibe un vector con números enteros y nos diga cual es la posición del menor.

12. Utilizando la función anterior, escribe un programa que reciba N números, los inserte en una tabla y escriba la tabla en pantalla ordenada de menor a mayor.

13. Escribe una función que sume dos vectores de igual tamaño y rangos cualesquiera, dando como resultado otro vector de igual tamaño.

14. Escribe una calculadora que, mediante funciones y procedimientos, permita sumar, restar, multiplicar y dividir números romanos.

15. Escribe una calculadora que permita sumar, restar y multiplicar vectores.

16. La amplitud de un vector $(V_1, V_2, ..., V_n)$ se calcula aplicando la siguiente fórmula:

$$L = \sqrt{V_1^2 + V_2^2 + ... + V_n^2}$$

Escribe una función que calcule la amplitud de un vector de números reales.

    (a) Considera que el vector sólo contiene cuatro elementos.

    (b) Considera que el vector tiene un número arbitrario de elementos.

17. Decimos que dos vectores U y V son ortogonales cuando se verifica que:

$$\sum_{i=1}^{n} u_i * v_i = 0$$

Escribe una función que nos sirva para saber si dos vectores (de cualquier tamaño) son ortogonales.

18. Escribe una función que compruebe si el nombre de un identificador sigue las reglas de Ada.

19. Utilizando la función anterior, escribe un programa que lea de teclado una línea de texto y nos diga cuantas palabras son identificadores Ada válidos.

20. Definimos la rotación de un vector a la derecha como una operación que mueve cada elemento una posición a la derecha y el último lo coloca en la primera posición. Escribe una función que rote un vector un número determinado de posiciones a la derecha.

21. En estadística, la mediana es el valor central de un conjunto de valores ordenados. En caso de que el número de valores sea par, la mediana es el valor medio de los dos valores centrales. Escribe una función que reciba un número arbitrario (y no ordenado) de valores de entrada y retorne la mediana.

22. En estadística, la moda es el valor que más veces se repite en un conjunto de datos. Escribe una función que reciba un número arbitrario de valores de entrada y retorne la moda.

23. Escribe las declaraciones Ada necesarias para representar la siguiente tabla que indica si un país es vecino de otro:

	Belgica	Italia	Francia
Belgica	no	si	no
Francia	si	no	si
Italia	no	si	no

Añade algunos países más y escribe la función *Número_De_Vecinos()* que recibe como parámetro un país y una tabla similar a la descrita anteriormente y calcula el número de vecinos de dicho país.

24. Escribe un programa que lea un mensaje y nos codifique y decodifique mensajes en Morse. La tabla de codificación Morse es la siguiente:

```
A .-      H ....   O ---    U ..-
B -...    I ..     P .--.   V ...-
C -.-.    J .---   Q --.-   W .--
D -..     K -.-    R .-.    X -..-
E .       L .-..   S ...    Y -.--
F ..-.    M --     T -      Z --..
G --.     N -.
```

NOTA: La codificación la debe realizar el procedimiento *Codificar* y la decodificación el procedimiento *Decodificar*.

25. Escribe un programa que, mediante funciones, nos permita sumar, restar, multiplicar y dividir dos fracciones. NOTA: El resultado debe mostrarse en pantalla también en forma de fracción.

4.1 Recursividad

1. Una empresa nos hace la siguiente oferta de trabajo:

   - El primer año nos paga 120.000 ptas.
   - El resto de los años tendremos un aumento de sueldo anual de 4 por ciento sobre lo que estemos cobrando.

   Escribe una función recursiva que calcule el sueldo que tendremos dentro de X años.

2. Una forma de calcular el máximo común divisor de dos números enteros positivos es utilizar la siguiente definición recursiva:

   - $mcd(m, n) = m$, si $m = n$.
   - $mcd(m, n) = mcd(m - n, n)$, si $m > n$.
   - $mcd(m, n) = mcd(m, n - m)$ en el resto de los casos.

   Escribe la función recursiva $mcd()$ que calcula el máximo común divisor de dos números aplicando esta definición.

3. Escribe un procedimiento recursivo que escriba en pantalla todos los dígitos de un número entero positivo en orden inverso. NOTA: Utiliza la división por 10 para extraer cada uno de los dígitos.

```
Ejemplo  : 1234
Resultado: 4321
```

4. Dado un número natural N y una base de numeración B inferior a N, escribe un procedimiento recursivo que escriba en pantalla N en base B.

```
Ejemplo:
     Numero (N): 15
     Base   (B): 2
     Resultado : 1111
```

5. Escribe un procedimiento recursivo que multiplique dos números naturales utilizando la siguiente fórmula:

   - $a * b = a$, si $b = 1$
   - $a * b = a * (b - 1) + a$, si $b > 1$

6. Los coeficientes de un binomio se pueden definir de la siguiente forma:

$$\binom{n}{0} = 1$$

$$\binom{n}{n} = 1$$

$$\binom{n}{k} = \binom{n-1}{k-1} + \binom{n-1}{k} 0 < k < n$$

   Escribe una función recursiva que calcule el valor de: $\binom{i}{j}$.

5 Estructuras de datos

5.1 Vectores

1. Escribe un programa que lea desde teclado 20 números y los vuelva a escribir en pantalla en orden inverso.

2. Escribe un programa que lea desde teclado 20 números y los vuelva a escribir en pantalla sin repetir los números que estén duplicados. NOTA: No es necesario que los números estén ordenados para que el programa funcione correctamente.

3. Escribe un programa que reciba desde teclado un cierto número de números reales y escriba en pantalla:

   (a) El número menor.

   (b) El número mayor.

   (c) El valor medio de todos los números.

   Supongamos que este programa lo necesitan tres personas diferentes, y cada una quiere leer los datos de una forma diferente:

- La primera persona quiere que siempre se lean 10 números.
- La segunda persona quiere que no se fije esta cantidad. Ella quiere decir al programa (cada vez que lo ejecute) el número de datos que va a introducir.
- La tercera persona dice que ella no sabe cuantos datos va a introducir. Quiere que el programa le pregunte si quedan más datos.

4. Declara un enumerado que contenga los días de la semana. A partir de esta declaración, crea la tabla *Mañana* que utilizaremos para saber cuál es el día siguiente a cualquier día de la semana.

5. Declara un array de enteros (F) que tenga un índice entre 0 y N. Escribe un programa que almacene en cada elemento del array los números de Fibonacci. Los números de Fibonacci se calculan de la siguiente forma:

$$F_0 = 0, \quad F_1 = 1, \quad F_i = F_{i-1} + F_{i-2}$$

6. Declara una tabla que sirva para traducir números romanos en números enteros positivos. Utilizando esta tabla, escribe un programa que lea un número romano y lo traduzca a un entero positivo.

7. Escribe un programa que reciba un número entero positivo y genere el número romano equivalente.

8. Escribe un programa que calcule y almacene en una tabla los 50 primeros números primos. Para calcular si un número es primo lo dividimos por todos los que tenemos en la tabla; si es primo, lo insertamos en la tabla.

5.2 Cadenas de caracteres

1. Escribe un procedimiento que cuente el número de espacios en blanco que hay en una línea de texto.

```
Ejemplo:                1         2         3
               1234567890123456789012345678901234
    TEXTO    : Estoy   aprendiendo a   programar.
    RESULTADO: 7.
```

2. Escribe un programa que lea un texto y nos indique:
   - Cuántas letras contiene (mayúsculas y minúsculas).
   - Cuántos números contiene.
   - Cuántos caracteres especiales (caracteres que no son letras ni números).

3. Escribe un programa que lea tres líneas de texto y cuente el número de vocales.

4. Escribe un procedimiento que sustituya los espacios en blanco consecutivos por un único espacio.

```
Ejemplo:

    TEXTO    : Estoy   aprendiendo a    programar.
    RESULTADO: Estoy aprendiendo a programar.
```

5. Escribe un procedimiento que centre en pantalla cualquier frase que reciba como parámetro de entrada.

6. Escribe un programa que reciba desde teclado una línea con un máximo de 80 caracteres y nos escriba en las líneas siguientes cada una de las frases que hemos puesto. NOTA: Suponemos que el separador de frases es el punto.

7. Escribe un programa que lea una fecha en formato americano (mes/día/año) y nos la genere en el formato utilizado en España (día/mes/año) y en el formato estándar ISO (año-mes-día).

```
Ejemplo:

    Introduce la fecha en formato americano: 10/25/98

    25/10/98
    1998-10-25
```

8. Escribe una función que busque una palabra dentro de una frase y nos indique la posición donde la encontró. Tanto la palabra como la frase la introducimos nosotros desde el teclado.

```
Ejemplo:

    FRASE   : Quiero buscar una palabra en esta frase.
    PALABRA : una
    POSICION: 15
```

9. Modifica el programa anterior para que nos diga cuantas veces aparece una determinada palabra dentro de una frase.

10. Escribe un programa que busque una palabra dentro de una frase y la sustituya por otra palabra.

```
Ejemplo:

    FRASE    : Esta es una posible frase.
    BUSCA    : una
    SUSTITUYE: otra
    RESULTADO: Esta es otra posible frase.
```

11. Escribe un procedimiento que, añadiendo espacios, justifique un texto a un tamaño que especificamos mediante un parámetro de entrada.

    NOTA: Para justificar debemos calcular el número de espacios que debemos insertar y seguidamente recorrer el texto de izquierda a derecha (o de derecha izquierda) buscando los espacios en blanco y añadiendo un nuevo espacio hasta completar dicha cantidad.

```
Ejemplo:

    TEXTO    : Este es el texto a justificar.
    TAMANO   : 40
                         1         2         3         4
               1234567890123456789012345678901234567890
    RESULTADO: Este    es   el   texto   a   justificar
```

12. El método simple de cifrado de información consiste en sustituir cada letra del abecedario por otra que está un 3 posiciones por delante de ella. Por ejemplo, la A se sustituye por la C, la B se sustituye por la D, la C por la E y así sucesivamente.

```
Ejemplo:

    Texto original   : JAVIER
    Texto codificado: LCXKGT
```

    (a) Escribe un programa que lea un mensaje con un tamaño máximo de 60 caracteres, lo cifre utilizando el método César, y escriba en pantalla el resultado. NOTA: Suponemos que todas las letras del mensaje están siempre en mayúscula; si hay alguna en minúscula se deja sin codificar.
    (b) Escribe un programa que lea un mensaje cifrado por el método César, lo descifre y genere en pantalla el mensaje original.
    (c) Escribe un programa que lea un mensaje cifrado cuyo desplazamiento desconocemos, y escriba en pantalla todos los posibles mensajes originales.

13. Un texto palíndromo es un texto que se lee igual de izquierda a derecha que de derecha a izquierda. Escribe un programa que lea un texto y nos diga si es palíndromo.

14. Escribe un programa que calcule la frecuencia de todos los caracteres de que consta un mensaje.

15. Los elementos químicos se representan mediante una o dos letras: la primera siempre en mayúscula y la segunda en minúscula (por ejemplo C, Hg, Au). Escribe un programa que calcule el peso molecular de una fórmula química.

    NOTAS:

    - Los valores de los pesos moleculares están en cualquier libro de química.

- Como no podemos poner en el teclado subíndices, utilizaremos la representación H2O para representar H_2O(2 moléculas de Hidrógeno y una de Oxígeno).

5.3 Matrices

1. Escribe un programa que nos diga la posición (I,J) del número más grande que hay en una matriz.
2. Escribe una función que calcule la suma de todos los elementos de la diagonal principal de cualquier matriz cuadrada. Si la matriz no es cuadrada, la función retorna siempre -1.
3. Escribe una función que calcule la suma de todos los elementos de la diagonal secundaria de cualquier matriz cuadrada. Si la matriz no es cuadrada, la función retorna siempre -1.
4. Escribe un procedimiento que ponga a cero todos los elementos positivos de una matriz cuadrada de cualquier dimensión.
5. Escribe una función que calcule la suma de todos los elementos del borde de cualquier matriz de cualquier dimensión.
6. Escribe un procedimiento que intercambie dos filas de una matriz de cualquier dimensión.
7. Un tablero de ajedrez tiene 64 cuadros. Las columnas se nombran mediante las letras a,b,c,d,e,f,g y h, y las filas mediante los números del 1 al 8. Las piezas de ajedrez son: reina, rey, alfil, caballo, torre y peón. Las piezas son blancas o negras. Escribe una declaración Ada que describa la posición inicial del tablero, que es la siguiente:

```
                    NEGRAS
     ----------------------------------------
  8  | T* | C* | A* | D* | R* | A* | C* | T* |
     ----------------------------------------
  7  | P* | P* | P* | P* | P* | P* | P* | P* |
     ----------------------------------------
  6  |    |    |    |    |    |    |    |    |   T: Torre
     ----------------------------------------   C: Caballo
  5  |    |    |    |    |    |    |    |    |   A: Alfil
     ----------------------------------------   D: Dama
  4  |    |    |    |    |    |    |    |    |   R: Rey
     ----------------------------------------   P: Peon
  3  |    |    |    |    |    |    |    |    |
     ----------------------------------------   *: Ficha negra
  2  | P  | P  | P  | P  | P  | P  | P  | P  |
     ----------------------------------------
  1  | T  | C  | A  | D  | R  | A  | C  | T  |
     ----------------------------------------
       a    b    c    d    e    f    g    h
                   BLANCAS
```

8. Realiza la declaración de una matriz tridimensional de números enteros. Utilizando esta declaración, escribe un programa que calcule el número de elementos positivos que contiene una matriz tridimensional llena de números enteros.

9. Escribe un programa que lea desde teclado un número N, cree una matriz cuadrada de NxN, pida desde teclado los elementos de la matriz, calcule la suma de cada una de sus filas y columnas, y almacene el resultado en dos vectores (uno con la suma de cada filas y otro con la suma de cada columnas). Finalmente el programa debe escribir en pantalla el contenido de estos dos vectores.

10. Supongamos que nuestro lenguaje de programación no es capaz de trabajar con matrices de rango (1..N, 1..M). Escribe un programa que, mediante funciones, nos permita trabajar con un un vector como si fuese una matriz. Por ejemplo, cuando queremos acceder al elemento (4,4), accedemos al elemento 44 del vector; cuando queremos acceder al elemento (2,3), accedemos al elemento 23, y así sucesivamente.

11. Generiliza la solución del problema anterior para que se pueda trabajar con matrices de cualquier rango.

12. Escribe una función que nos diga si una matriz es simétrica.

13. Escribe las funciones *Fila()* y *Columna()* que nos permiten obtener los elementos de una determinada fila o columna de cualquier matriz de números enteros positivos.

14. Escribe una función que calcule el producto escalar de dos vectores.

15. Utilizando las funciones de los dos problemas anteriores, escribe un programa que realice la multiplicación de dos matrices cualesquiera.

5.4 Registros

1. Escribe la declaración de un registro que sirva para almacenar la información de un número complejo. Utilizando esta declaración escribe:

    - Un procedimiento que sirva para leer números complejos desde teclado.
    - Una función que sirva para sumar números complejos.
    - Una función que que sirva para restar números complejos.
    - El programa principal para utilizar esta calculadora de números complejos.

2. Escribe un programa que nos sirva como agenda de teléfonos. Cada elemento de la agenda debe contener al menos el nombre, el teléfono y la dirección de una persona.

3. Una empresa nos ha encargado un programa para llevar la gestión de su inventario. Por cada artículo necesitan que el programa recuerde:

    - La identificación del artículo (un código de cuatro caracteres).
    - Una breve descripción de hasta 60 caracteres.
    - El número de artículos existentes en el almacén.
    - El precio de venta.

    Escribe un programa que lee esta información desde teclado y a continuación espera a que le demos una orden. Las órdenes posibles son:

    - INFO xxxx: Escribir toda la información del artículo xxxx.
    - VENDER xxxx: Registrar que se ha vendido un artículo del producto xxxx.
    - COMPRAR xxxx: Registrar que se ha comprado un artículo del producto xxxx.

    NOTA: Cada una de estas operaciones la debe realizar un procedimiento diferente.

6 Atributos

1. Escribe un programa que muestre en pantalla cual es el entero más pequeño y más grande que es capaz de manejar nuestro compilador de Ada. Haz lo mismo con los números reales.

2. Escribe un programa que muestre en pantalla la tabla de caracteres ASCII de los caracteres que hay entre la posición 65 y la posición 127.

3. Escribe un programa que lea del teclado un caracter y calcule el siguiente caracter (siguiendo el orden de la tabla ASCII).
4. Escribe un programa que lea del teclado dos caracteres y escriba en pantalla todos los caracteres ASCII que están entre ellos (siguiendo el orden de la tabla ASCII).
5. Escribe un programa que lea de teclado una letra en minúscula y, utilizando atributos, la convierta a mayúscula y la escriba en pantalla.
6. Supongamos que la variable T contiene un carácter entre '0'y '9'. Escribe una sentencia que realice la conversión de T a un entero en el rango 0..9 y lo asigne a una variable de tipo entero.

7 Ficheros

7.1 Ficheros de texto

1. Escribe un programa que cuente el número de líneas, palabras y caracteres que hay en cualquier fichero de texto.
2. Escribe un programa que nos pase a mayúsculas todo el contenido de un fichero de texto.
3. Escribe un programa que calcule la frecuencia de todas las palabras que hay en un fichero de texto.
4. Escribe un programa que muestre en pantalla el contenido de un fichero. Cada vez que se llene la pantalla (se escriben en pantalla 23 líneas) el programa debe esperar a que se pulse una tecla.
5. Escribe un programa que nos pida una palabra y el nombre de un fichero de texto y nos escriba en pantalla todas las líneas del fichero que contienen esa palabra.

7.2 Ficheros de acceso secuencial

1. Para llevar el control de las notas de un examen nos han encargado escribir un programa que almacene en un fichero secuencial la siguiente información:

   - Código del alumno.
   - La nota obtenida en cada una de las 10 preguntas.

   Además, debemos escribir otro programa que nos diga el código del alumno que sacó la mejor nota en el examen.

2. Supongamos que tenemos el fichero PERSONAS.DAT con el nombre y la edad de un conjunto de personas. Se pide:

    - Escribir un programa que copie el contenido del fichero en otro fichero llamado COPIA.DAT.
    - Modificar el programa anterior para que nos permita insertar un elemento en el fichero de salida. El programa nos debe preguntar si queremos insertar el elemento por el principio, por el final o en una determinada posición del fichero de salida.
    - Modificar el programa para que el programa genere el fichero de salida EDADES.DAT, que solamente contiene las edades de todas las personas.
    - Escribir un programa que lea el fichero PERSONAS.DAT y le sume 1 a la edad de todas las personas.

3. Escribe un programa que copie cualquier fichero en otro.

4. Escribe un programa que calcule la longitud de un fichero.

5. Supongamos un fichero secuencial de números enteros. Escribe un programa que inserte ordenadamente un número en el fichero.

```
      1 3 12 14          -------->   1 3 7 9 12 14
      ---------                      -------------
   Fichero Inicial                     RESULTADO
```

6. Escribe un programa que funcione como una agenda electrónica. Cada ficha debe contener la siguiente información:

    - Nombre.
    - Peso.
    - Teléfono.

   El programa debe contener los procedimientos y funciones que consideres necesarios para buscar la información de una persona, aadir la información de una persona, buscar la información de una persona (para mostrarla en pantalla) y eliminar su información del fichero.

7. Escribe un programa que, a partir de la información contenida en el fichero anterior, nos permita crear otro fichero que contiene la misma información y además la edad de cada persona.

8. Escribe un programa que, a partir del fichero generado en el apartado anterior, nos genere dos ficheros: uno con los jugadores menores de 25 años y otro con los jugadores mayores de 18 años.

7.3 Ficheros de acceso directo

1. Escribe el programa juego del *el ahorcado*. El programa tiene un fichero de acceso directo con un conjunto de palabras. El jugador debe adivinar una de las palabras que está almacenada en un fichero. Para ello dice una letra y el programa le enseña en pantalla los caracteres que ha acertado. El juego termina cuando el jugador acierta la palabra completa o falla un determinado número de intentos.

```
Ejemplo:
   ****
   Adivina una letra: O
   O*O*
   Adivina un letra: M
   <<<< NO HAY NINGUNA M. TE PERMITO 3 FALLOS MAS >>>>
   Adivina una letra: L
   OLO*
   Adivina una letra: R
   OLOR
   >>>>>> GANASTE
```