



UNIVERSIDAD DE MÁLAGA  
DPTO. DE LENGUAJES Y C. DE LA COMPUTACION  
E.T.S. DE INGENIERIA INFORMATICA

## FUNDAMENTOS DE LA PROGRAMACIÓN

### PRIMERA RELACIÓN DE PROBLEMAS

1. Dado **unsigned** num, escribe una expresión booleana que determine si num es un número par.
2. Dado **unsigned** num, escribe una expresión booleana que determine si num es un número de tres dígitos.
3. Dado **unsigned** num, escribe una expresión booleana que determine si num es un divisor de 100.
4. Escribe una expresión booleana que represente la pertenencia:  $x \in \{3, 4, 5, 6, 7\}$ .
5. Escribe una expresión booleana que represente la pertenencia:  $x \in \{1, 2, 3, 7, 8, 9\}$ .
6. Escribe una expresión booleana que represente las pertenencias:  $x \in \{3, 4, 6, 8, 9\}$ ,  $y \in \{6, 7, 8, 3\}$
7. Dados **unsigned** x, y, escribe una expresión booleana que represente que ni x ni y son mayores que 10.
8. Dado **char** c, escribe una expresión booleana que determine si c es una letra mayúscula.
9. Dado **char** c, escribe una expresión booleana que determine si c es una letra.
10. Mejora el siguiente fragmento de código, siendo b una expresión lógica y T, S y U tres sentencias cualesquiera que no afectan a b:

```
if (b) {  
    T;  
    S;  
} else {  
    T;  
    U;  
}
```

11. Mejora el siguiente fragmento de código, siendo b una expresión lógica y x una variable booleana:

```
if (b) {  
    x = true;  
} else {  
    x = false;  
}
```

12. Simplificar el siguiente segmento de algoritmo de forma que se necesiten menos comparaciones:

```

    if (edad >= 65) {
        cout << "Jubilado";
    } else {
        if (edad < 18) {
            cout << "Menor de edad";
        } else {
            if ((edad >= 18) && (edad < 65)) {
                cout << "Activo";
            }
        }
    }
}

```

13. Diseñar un algoritmo que lea por teclado dos números enteros y determine si uno es divisor del otro.
14. Diseñar un algoritmo para determinar si un número entero (leído por teclado) es múltiplo de alguno de los siguientes números (o de varios de ellos): 3, 4, 5.
15. Diseñar un algoritmo para convertir un ángulo expresado en grados / minutos / segundos a radianes.
16. Escribir un algoritmo para calcular el precio de una partida de artículos aplicando un descuento creciente con la cantidad comprada según la siguiente tabla (la cantidad comprada se lee por teclado):

Número de unidades	Precio unitario
1	100
2	95
3	90
4 o más	85

17. ¿Cuántas veces se ejecutará la Acción I ?

```

cont=0;
while (cont < 10) {
    cont = cont + 1;
    if (cont == 10) {
        Acción I
    }
}

```

18. ¿Cuál es el valor de la variable booleana test después de la ejecución de los siguientes bucles?

a.

```

bool test;
unsigned N;

test = true;
for (unsigned contador= 1;contador<=N;contador++) {
    test = !test;
}

```

b.

```

bool test;
unsigned N;

test = N %2 ==0;
for (unsigned contador= 1;contador<=N;contador++) {
    test = !test;
}

```

}

19. ¿Qué rango de valores enteros pueden tomar X e Y para garantizar que el siguiente algoritmo iterativo termina?

```

A = X;
B = Y;
do {
    while (A > B) {
        A = A - B;
    }
    while (B > A) {
        B = B - A;
    }
} while (A != B);

```

20. Confecciona un bucle que lea de teclado un texto carácter a carácter hasta localizar un punto, y que al final dé como salida el número de comas encontradas, y el número de caracteres leídos.
21. Diseña un algoritmo que lea un número entero  $n$  por teclado (distinto de 0). Después el usuario introducirá por teclado una secuencia de números enteros terminada en 0. El algoritmo debe determinar si el número  $n$  aparece o no en la secuencia.
22. Diseña un algoritmo que determine si la cadena `abc` aparece en una sucesión de caracteres cuyo final viene dado por un punto.
23. Diseña un algoritmo que lea un número natural  $n$  por teclado (mayor que 0) y calcule el  $n$ -ésimo número de la serie de Fibonacci. Los dos primeros números de esta serie son el cero y el uno, y a partir de éstos cada número se calcula realizando la suma de los dos anteriores.
24. Escribe un algoritmo que encuentre el mayor, el menor y la media aritmética de una colección de  $N$  números leídos por el teclado donde  $N$  es el primero de los números.
25. Escribe un algoritmo que lea una lista de números enteros terminada en 0, y que encuentre y escriba en la pantalla la posición de la primera y de la última ocurrencia del número 12 dentro de la lista. Si el número 12 no está en la lista, el algoritmo debería escribir 0. Por ejemplo, si el octavo número de la lista es el único 12, entonces 8 sería la primera y la última posición de las ocurrencias de 12.
26. Desarrollar un algoritmo para el siguiente juego:  
El usuario introduce un límite inferior, un límite superior y piensa un número en ese rango. El ordenador tiene que acertarlo. Para ello el ordenador propone un número y el usuario responde con  $>$ ,  $<$  o  $=$  (correspondiente a acertado y el programa acaba). Si la respuesta es  $>$  o  $<$ , el ordenador pondrá otro número hasta que lo acierte.
27. Desarrollar un algoritmo para el siguiente juego:  
El usuario introduce un número natural  $n$  que representa a  $n$  objetos. La máquina decide quien empieza y alternativamente, el usuario y la máquina retiran cada uno 1 2 o 3 objetos (es decir, elige restar 1, 2 o 3 a  $n$ ). El que retira el último objeto pierde.  
En las condiciones del problema, es posible desarrollar un algoritmo para que siempre gane la máquina. Construir tal algoritmo.