

1. Sea una escena como la de la figura. La escena tiene un cuadrado, un triángulo (no tiene que ser equilátero) y un círculo. Las dimensiones y las posiciones de estas figuras en la escena pueden cambiar. Se pide, crear una estructura que sea capaz de albergar toda la información necesaria para definir la escena.
2. Crear un programa que muestre por pantalla las tres posiciones en las que aparece el texto "tri" en el siguiente texto: "Tres tristes tigres comen trigo en un trigal"
3. A partir del texto "En un lugar de la Mancha, de cuyo nombre no quiero acordarme" crear un programa que muestre por pantalla dos textos: lo que hay antes de Mancha (*En un lugar de la*) y lo que hay después (*, de cuyo nombre no quiero acordarme*).
4. Pedir al usuario que introduzca un texto por teclado. Comprobar si el texto contiene la palabra "password" y en caso de que la contenga mostrar por pantalla **la palabra** que haya a continuación.
5. Realizar un programa que recorra un array de valores enteros y busque si contiene el número 7. En caso de que lo contenga imprimir por pantalla la primera posición en la que aparezca.
6. Realizar un programa que recorra un array de valores enteros y busque si contiene el número 7. En caso de que lo contenga imprimir por pantalla todas las posiciones en las que lo contiene.
7. Pedir al usuario que introduzca un texto por teclado y mostrar por pantalla el número de veces que aparece la vocal "a".
8. Pedir al usuario que introduzca un texto por teclado y mostrar todas las palabras que empiecen por "a".
9. Crear un array de números enteros e indicar cuántas veces aparece el número 7.
10. Crear un programa que calcule la media de un array de números enteros.
11. Crear un array de números enteros (entre el 1 y el 10) e indicar cuál es el número que más veces aparece (y cuántas veces aparece).
12. Crear un array de números enteros e indicar cuántas veces seguidas aparece como máximo el número 7. Es decir, para el array {1,2,7,7,4,3,3,7,7,7,2} el número máximo de veces seguidas que aparece el 7 es 3. Aparece 2 veces seguidas, y 3 veces seguidas, por lo que el máximo es 3.
13. Pedir al usuario que introduzca un texto por teclado y mostrar todas las palabras que contengan al menos una "a".
14. Usando `for elem:vec`. A partir de un vector de números enteros, crear otro que contenga sólo los números pares. Por ejemplo, a partir de {1,3,4,5,6,7} se crea -> {4,6}.
15. Usando `while`. A partir de un vector de números enteros, crear otro que contenga sólo los números pares. Por ejemplo, a partir de {1,3,4,5,6,7} se crea -> {4,6}.
16. Pedir un número al usuario y calcular el sumatorio desde 1 hasta dicho número.
17. Pedir tres números al usuario: *a*, *b* y *c*. Crear un vector que contenga todos los números múltiplos de *c* que hay entre *a* y *b* (incluidos).
18. A partir de un vector de `std::string` crear otro vector que contenga sólo los elementos que aparece después del `std::string` "guardar". Por ejemplo, para el vector {"perro", "gato", "guardar", "andres", "luis", "guardar", "botella"} crearía otro vector conteniendo {"andres", "botella"}.
19. Pedir al usuario un número y mostrar por pantalla si dicho número es primo o no (usando el bucle `while`).
20. Pedir al usuario que introduzca una cadena de texto por teclado, almacenarla en un `std::string`, crear a continuación un `std::vector` que contenga cada una de las palabras de la cadena por separado.
21. Crear un `std::array` de 10 elementos que contenga los 10 primeros números de la Serie de Fibonacci. Los números se deben calcular, no introducir uno a uno. Sólo se

pueden introducir manualmente los dos primeros.

22. Crear un programa que calcule los n primeros números de la Serie de Fibonacci. El programa debe pedir al usuarios cuántos números desea calcular, y continuación estos se deben calcular y almacenar en el tipo de datos oportuno.
23. Realizar un programa que pida al usuario un número y muestre por pantalla su factorial.
24. Realizar un programa que pida al usuario un número y almacene en un `std::vector` sus factores (todos aquellos números menores que él por los que puede ser dividido).
25. Realizar un programa que pida al usuario un número y muestre por pantalla si el número es primo o no.
26. A partir de un array de 100 números enteros aleatorios entre 0 y 20 (ver [cómo generar números aleatorios \(http://www.cplusplus.com/reference/cstdlib/rand/\)](http://www.cplusplus.com/reference/cstdlib/rand/)), guardar en el tipo de dato pertinente todos aquellos que sean múltiplos de 3 y posteriormente imprimirlos por pantalla (no se debe repetir ninguno).
27. A partir de un array de 20 numeros enteros aleatorios entre 0 y 100, imprimirlos por pantalla de modo ordenado y sin que se repitan.
28. Crear una función que reciba como parámetro una frase y devuelva un vector que contenga cada palabra por separado.
29. Crear una función que reciba como parámetros una frase y un caracter "separado" y devuelva un vector con cada uno de los elemtos separados. Por ejemplo, si le pasamos la frase "www.nebrija.es" y como elemento separado el caracter ' . ', devolverá el vector {"www", "nebrija", "es"}.
30. Crear una función que recorra un vector de números enteros positivos y devuelva un vector con aquellos números que sean primos. Para ello se creará previamente una función que diga si un número es primo o no.
31. Crear una función que devuelva un vector con los n primeros números primos. El número n se le debe pasar a la función como parámetro.
32. Crear una función que reciba un número entero positivo y devuelva un vector son sus factores ([ver aquí \(https://matematica.fandom.com/wiki/Descomposicion_de_numeros_naturales_en_factor\)](https://matematica.fandom.com/wiki/Descomposicion_de_numeros_naturales_en_factor) que son los factores de un número natural).