



UNIVERSIDAD DEL MAGDALENA
Facultad de Ingeniería - Ingeniería Electrónica
Microprocesamiento I
Ing. Carlos Cabas Meriño

1. Dado un vector definido como:

```
unsigned char palabras[] = "amor, bueno, anillo, cuatro, dado, fecha, iglesia,  
jabon"
```

Realice un programa que organice las palabras por orden alfabético. Tenga en cuenta que la cantidad de palabras puede cambiar, al igual que las palabras en si.

2. Escriba un programa que simule el tiro de dos dados. El programa debe utilizar la función **rand()** para lanzar el primer dado y debe utilizar **rand()** nuevamente para el segundo dado. Luego se debe calcular la suma de los dos valores.

Nota: Cada dado puede mostrar un valor entero de 1 a 6, de manera que la suma de ambos puede variar de 2 a 12 donde 7 es la suma mas frecuente y 2 y 12 la menos frecuente. En la siguiente imagen se puede observar el comportamiento posible de los dados:

	1	2	3	4	5	6
1	2	3	4	5	6	7
2	3	4	5	6	7	8
3	4	5	6	7	8	9
4	5	6	7	8	9	10
5	6	7	8	9	10	11
6	7	8	9	10	11	12

Su programa debe realizar 10.000 lanzamientos, utilice un vector para almacenar los resultados, muestre los resultados en forma tabular y compruebe cual es la suma mas frecuente.

3. Dado un numero hexadecimal en formato string como:

```
unsigned char hex[] = "2AFD5"
```

indique cuantos bits en 1 tiene el número. Tenga en cuenta que la cantidad de dígitos puede variar al igual que el número.

4. A usted se le ha contratado para romper la seguridad de una cuenta de Facebook, el usuario que perdió la contraseña solo recuerda que era una contraseña de 6 caracteres alfabéticos en minúsculas (a-z).

Este tipo de contraseñas es posible encontrarlas con un ataque de **fuerza bruta** (ir probando combinaciones hasta encontrar la coincidencia). La contraseña estará almacenada en un vector ej: `pass[] = "axzptd"`.

Al finalizar el programa debe mostrar la contraseña encontrada y cuantos intentos realizó.

Nota: se tendrá en cuenta lo optimo del programa y si muestra información del progreso al usuario.

5. Para programar un microcontrolador PIC, se hace uso de un archivo .hex, conocido como formato Intel Hex 86 el cual tiene la siguiente estructura:

: 02 0000 00 4728 8F

De donde:

: → Caracter de Inicio
02 → Longitud de datos
0000 → Dirección en Memoria
00 → tipo de dato
4728 → Dato
8F → Checksum

El checksum es igual a byte menos significativo del complemento a 2 de la suma de todos los bytes anteriores sin incluir el carácter de inicio ej:

+0x02
+0x00
+0x00
+0x00
+0x47
+0x28

0x71 → en binario = 01110001 su complemento a 1 es igual a 10001110 a esto se le suma un 1 para hallar el complemento a 2 = 10001111 → **0x8F**

Se pide realizar un programa que interprete tramas en formato Intel Hex de cualquier longitud y verificación de checksum.