

1. Generar las siguientes funciones para realizar conversiones de temperaturas entre grados Fahrenheit y Celsius:

```
double F2C(double fahr);
double C2F(double celsius);
```

2. Implemente un programa que a partir del uso de la función `rand()` obtenga un histograma con N intervalos de clase, en donde cada intervalo contenga la cantidad de veces que la función `rand` dio un elemento en él (en el primer intervalo contendrá la cantidad de elementos con valores entre 0 y `RAND_MAX/N - 1`, el segundo la cantidad de elementos con valores entre `RAND_MAX/N` y `2 * RAND_MAX/N - 1`, etc).
3. Implementar una función que reciba un vector y su dimensión y lo imprima
Prototipo: `void ImprimeVector(double Vector[], int NumElementos);`
4. Implementar una función como la anterior para ingresar los elementos del vector.
5. Implementar una función que reciba un vector y su dimensión y los ordene en sentido creciente utilizando algún algoritmo.
6. Arme un `.h` con los prototipos de las funciones de los ej. 3, 4 y 5 y copie las definiciones de esas funciones en un archivo separado llamado `misfuncs.c`. Escriba un programa que utilice todas esas funciones y compílelo con:

```
gcc miprog.c misfuncs.c -o miprog
```

7. Correr repetidas veces el programa realizado en el problema 5 para distintos tamaños del vector (inicializado a través de la función `rand`) e ir haciendo manualmente una tabla (tamaño, tiempo) para encontrar alguna correlación entre estas dos variables. Analizar el algoritmo utilizado para corroborarlo.
8. Implementar la función `InvierteVector` que recibe un vector de enteros y su dimensión e invierte a todos sus elementos (El primer elemento pasa a ser el último, el segundo pasa a penúltimo, etc.) Prototipo: `void InvierteVector(int V[], int n);`
9. Implemente las siguientes funciones estadísticas:

```
double promedio(double datos[], int n);
double dstandard(double datos[], int n);
```

que calculan el promedio y la desviación estándar de un conjunto de 'n' valores:

$$\bar{x} = \frac{1}{N} \sum_{i=1}^N x_i$$

$$\sigma = \sqrt{\frac{1}{N-1} \sum_{i=1}^N (x_i - \bar{x})^2}$$

10. Implemente las siguientes funciones de manejo de bits:

```
// retorna la cantidad de bits en 1 presentes en la representacion de 'value'
int numOnes(int value);

// devuelve el valor 'value', con el bit 'bit' en 1
int setBit(int value, int bit);

// devuelve el valor 'value', con el bit 'bit' en 0
int resetBit(int value, int bit);

// chequea si el bit 'bit' esta en 1
int testBit(int value, int bit);
```

11. Escriba un programa que utilizando la siguiente estructura para representar puntos en 2D:

```
typedef struct {
    double x;
    double y;
} Punto2D_t;
```

pida al usuario los puntos correspondientes a los 3 vértices de un triángulo y los almacene en un arreglo. El programa deberá clasificar dicho triángulo de acuerdo a las longitudes de sus lados como: "equilátero", "isósceles" o "escaleno" y también de acuerdo a sus ángulos como: "acutángulo", "rectángulo" u "obtusángulo". Ayuda: recuerde la relación que existe entre el producto escalar de dos vectores y el ángulo que forman. ¿Qué criterio utilizará para decidir si dos lados son iguales o si un ángulo es exactamente 90 grados? Implemente las funciones necesarias como para que el programa luzca de la siguiente manera:

```
int main()
{
    Punto2D_t triangulo[3];
    int clasificacionLados, clasificacionAngulos;
    cargaVertices(triangulo, 3);
    clasificacionLados = analizaLados(triangulo);      // 0: equil., 1: isóc., 2: escal.
    clasificacionAngulos = analizaAngulos(triangulo); // 0: acut., 1: rect., 2: obtus.
    printf(...); // imprimir resultados
    return 0;
}
```

12. El problema del cumpleaños está relacionado con cuál es la probabilidad de que en un grupo de personas haya al menos dos que cumplan años el mismo día. Se sabe que si en una misma sala si hay 23 personas reunidas, la probabilidad es del 50,7%. Para 57 o más personas la probabilidad es mayor del 99%. Si bien esto se puede deducir de manera analítica, es posible hacer una comprobación empírica. Para eso, considere 1000 grupos de un tamaño fijo de personas, tomando sus fechas de nacimiento al azar y analice en cuántos casos al menos dos personas cumplen el mismo día. Los tamaños de los grupos deben tomarse entre 10 y 60, aumentando de a 10. Hacer una tabla que indique la probabilidad en función del tamaño de grupo. Válgase de funciones para descomponer el problema en sub-problemas.