

Ejercicios del Tema 2

Oscar Perpiñán Lamigueiro

1. Códigos ASCII

- Escriba un programa que permita averiguar qué acrónimo está representado por los códigos ASCII: 101 116 115 105 100 105

```
#include <stdio.h>

int main()
{
    printf("El acrónimo es %c%c%c%c%c%c",
           101, 116, 115, 105, 100, 105);
    return 0;
}
```

- Escriba un programa que imprima en pantalla las letras vocales y sus correspondientes caracteres ASCII.

```
#include <stdio.h>

int main()
{
    printf("Los códigos ASCII de las vocales son:\n");
    // Usando %d se imprime un char como entero.
    // El código \t inserta un tabulador.
    printf("a: %d\t", 'a');
    printf("e: %d\t", 'e');
    printf("i: %d\t", 'i');
    printf("o: %d\t", 'o');
    printf("u: %d\n", 'u');
    return 0;
}
```

- Desarrolle un programa que lea un carácter en minúscula y lo imprima en mayúscula. Tenga en cuenta que las letras mayúscula de la A a la Z y las minúsculas de la a a la z tienen códigos consecutivos (ver tabla ASCII).

```
#include <stdio.h>

int main()
{
    int letra, Letra;
    const int conv = 'A' - 'a'; // Diferencia entre mayuscula y minuscula
    printf("Escriba una letra minuscula:\n");
    scanf("%c", &letra);
    // Paso a mayuscula mediante la constante conv
    Letra = letra + conv;
    printf("En mayuscula es %c\n", Letra);
    return 0;
}
```

2. Cálculo del IVA

Realice un programa para calcular el IVA de un producto cuyo precio es introducido por el usuario. Suponga que el IVA es del 21 %.

```
#include <stdio.h>

int main()
{
    float precio, iva;
    printf("Indique el valor del producto en euros:\n");
    scanf("%f", &precio);
    iva = 0.21 * precio;
    printf("El IVA del producto es %.2f euros.\n", iva);
    return 0;
}
```

3. Conversión de temperatura

Escriba un programa para convertir de grados centígrados a grados Fahrenheit ($T_F = 9/5 * T_C + 32$)

```
#include <stdio.h>

int main()
{
    float tempC, tempF;

    printf("Escriba una temperatura en grados centigrados\n");
    scanf("%f", &tempC);
    // La clave está en usar 9.0 y 5.0 para que la división sea de
    // números reales. Si escribimos 9/5 estamos indicando una división
    // de números enteros.
    tempF = (9.0 / 5.0) * tempC + 32;

    printf("%.2f grados centigrados son %.2f grados Fahrenheit\n",
           tempC, tempF);
    return 0;
}
```

4. Conversión de monedas

Escriba un programa para convertir una cantidad de pesetas introducidas por el usuario en euros y euros en dólares.

```
#include <stdio.h>

int main()
{
    float pts, euros, dolar;
    // Constantes de conversión
    const float p2e = 1/166.386; //Pesetas a euros
    const float e2d = 1.06105; //Euros a dolares
    printf ("Introduce una cantidad en pesetas: \n");
    scanf ("%f", &pts);
    euros = pts * p2e;
    dolar = euros * e2d;
    printf ( "%.2f pesetas son %.2f euros y %.2f dolares\n",
```

```

        pts, euros, dolar);
    return 0;
}

```

5. Volumen de una esfera

El volumen de una esfera se calcula como: $\frac{4}{3} \cdot \pi \cdot r^3$. Realice un programa que permita calcularlo introduciendo por teclado el radio.

5.1. Versión sin math.h

```

#include <stdio.h>

int main()
{
    float r, area;
    // Definimos el valor de pi como una constante de tipo float
    const float pi = 3.141592;
    printf("Introduce el radio de la esfera \n");
    scanf("%f", &r);
    area = 4.0 / 3.0 * pi * r * r * r;
    printf("El volumen de la esfera es %f \n", area);
    return 0;
}

```

5.2. Versión usando math.h

```

#include <stdio.h>
#include <math.h>

int main()
{
    // La funcion pow de math devuelve el resultado como double. Por
    // tanto, debemos usar variables de este tipo.
    double r, area;
    printf("Introduce el radio de la esfera \n");
    scanf("%lf", &r);
    // La libreria math define el valor de pi en M_PI. Ademas, incluye
    // la funcion pow para calcular la potencia de un numero.
    area = 4.0 / 3.0 * M_PI * pow(r, 3);
    printf("El volumen de la esfera es %g \n", area);
    return 0;
}

```

6. Distancia entre puntos

Escriba un programa que pida al usuario las coordenadas de 2 puntos bidimensionales en coordenadas cartesianas, y que imprima por pantalla la distancia entre ambos puntos.

```

#include <stdio.h>
// La libreria math incluye las funciones pow y sqrt
#include <math.h>

int main()
{

```

```

float x1, x2, y1, y2;
double dx2, dy2, d;

printf("Primer punto\n");
scanf("%lf %lf", &x1, &y1);

printf("Segundo punto\n");
scanf("%lf %lf", &x2, &y2);

// La funcion pow realiza la potencia de un número
dx2 = pow(x1 - x2, 2);
dy2 = pow(y1 - y2, 2);
// La funcion sqrt realiza la raiz cuadrada
d = sqrt(dx2 + dy2);

printf("La distancia entre dos puntos es %f\n", d);
return 0;
}

```

7. Años bisiestos

Un año es bisiesto si es divisible por 4 pero no por 100, o si es divisible por 400.

- Escriba un programa que indique si un número es bisiesto a partir de un número introducido por el usuario.

```

#include <stdio.h>
int main()
{
    int x;
    _Bool bisiesto;

    printf("Dime un número entero.\n");
    scanf("%d", &x);
    // Definimos la condicion de ser bisiesto, y el resultado lo
    bisiesto = ((x % 4 == 0) && (x % 100 != 0)) || (x % 400 == 0);
    printf("%d es %s\n",
        x, bisiesto ? "bisiesto" : "normal");
    return 0;
}

```

- Escriba un programa que muestre en pantalla el número de días del mes de febrero a partir del año introducido por el usuario.

```

#include <stdio.h>
int main()
{
    int x;
    _Bool bisiesto;

    printf("Dime un número entero.\n");
    scanf("%d", &x);
    bisiesto = ((x % 4 == 0) && (x % 100 != 0)) || (x % 400 == 0);
    printf("El mes de febrero de %d tiene %d días\n",
        x, bisiesto ? 29 : 28);

    return 0;
}

```

8. Cifras de un número entero

Escriba un programa que, a partir de un número entero introducido por el usuario, muestre en pantalla las centenas, decenas y unidades (en este orden).

```
#include <stdio.h>
int main()
{
    int x, u, d, c;

    printf("Dime un número entero.\n");
    scanf("%d", &x);

    // Las unidades son el resto de dividir por 10.
    u = x % 10;
    // El siguiente paso consiste en eliminar las unidades: hacemos una
    // división de enteros con 10 (en lugar de 10.0), y por tanto se
    // pierden los decimales.
    x /= 10;
    // Para obtener las decenas y centenas repetimos la rutina
    d = x % 10;
    x /= 10;
    c = x % 10;

    printf("Unidades %d, Decenas %d, Centenas %d\n",
           u, d, c);

    return 0;
}
```