

G4011105 – Programación I

1. A continuación preséntase unha función para o cálculo do promedio de dous números, xunto cun exemplo de invocación dende un programa principal:

```
float encontprom(int num1, int num2)
{
    float promedio;

    promedio = (num1 + num2) / 2.0;
    return(promedio);
}

void main()
{
    int a,b;
    float resultado;
    printf("Dame dous valores enteiros: ");
    scanf("%d %d",&a,&b);

    resultado = encontprom(a, b);
    printf("Promedio=%f\n",resultado);
}
```

Engade outra función que devolva o máximo común divisor de dous enteiros positivos pasados como argumentos. Os datos pediranse dende a función `main()`, e nela tamén se imprimirá o resultado.

2. Implementa un programa que dada unha data calquera (dd, mm, aa) nos devolva o valor ordinal do día (1-366) correspondente. Para iso, implementa dúas funcións: unha que a partir do dato do ano (parámetro formal aa), devolva un valor 0 ou 1 para indicar se é bisesto ou non; e outra función que invoque á anterior para determinar se o ano é bisesto ou non (cando sexa necesario) e que despois calcule o valor ordinal. Podedes crear un vector que conteña o número de días correspondente a cada mes (ao mes de febreiro podedes sumarlle o valor devolto pola primeira función para ter en conta se é bisesto ou non) e así facilitar o cálculo.
3. Escribir unha función **Ordenar** que teña un vector de enteiros e o seu tamaño como parámetros formais, e que ordene os valores dentro do vector de menor a maior. Para o ordenamento empregade o método da burbulla. Este método percorre un vector varias veces ata que se completa unha iteración completa sen facer cambios. En cada iteración do bucle vaise comparando cada elemento do vector co seguinte, e cando ese par de elementos non están ordenados intercámbianse. Deste xeito os valores máis pequenos iranse desprazando cara as posicións máis baixas do vector cun desprazamento en cada iteración, ata que atopan o lugar que lles corresponde.
4. Escribir un programa que pida o valor de dúas datas (día, mes, ano) e calcule a distancia en días que as separa. Debe definirse unha estrutura (Date) para almacenar as datas e polo menos tres funcións: Ler, Mostrar, CalcularDistancia.

```
Date Ler (); // Esta función pedirá os valores de día, mes e ano, creará a estrutura para
// representar a data e devolveraa

int CalcularDistancia (Date d1, Date d2); //Determinará o número de días entre as dúas datas

void Mostrar (Date d, char f[]); // Imprimirá a data segundo o formato que se lle indique:
//f="1": dia/mes/ano; f="2": mes/dia/ano
```

5. Escribide un programa que realice as operacións indicadas a continuación sobre unha secuencia de enteiros almacenada nun arquivo. O programa deber ler os datos do arquivo e gardalos nun vector. Facede a lectura ata chegar ao final do arquivo (empregade o valor devolto pola función de lectura para detectalo). Implementade 3 funcións para: cálculo de máximo, mínimo e media. Estes valores deben imprimirse dende a función `main()`.

6. Repetide o exercicio, pero agora lendo os datos dun arquivo **datos.txt**, e gardando os resultados nun arquivo **datos_saída.txt**.
7. Consideremos o arquivo **weatherdata-420-72.csv** (abrídeo co Bloc de Notas) que contén información meteorolóxica en formato .csv, onde os datos van entre aspas (" ") e separados por comas. Os datos corresponden aos parámetros que se indican na primeira fila do arquivo: Data de recollida de datos, Ubicación da estación meteorolóxica, Temperatura (°C), Precipitación (mm), Velocidade do vento (m/s), Humidade relativa, Radiación solar (MJ/m²).
Escribide un programa que a partir dese arquivo xere outro co mesmo nome e extensión .txt, onde non aparezan os datos da primeira fila, e se eliminen os 3 caracteres " , /.
8. Repetide o exercicio pero agora almacenando todos os datos nun vector de estruturas. A estrutura **WeatherData** debe ter un campo por cada parámetro, e así almacenar os datos recollidos nunha data concreta (almacenar a data nunha estrutura Date). Gardar finalmente todos os datos en formato binario nun arquivo .dat.
9. Escribir un programa para facer consultas sobre ese arquivo binario. Tendo en conta que os datos están ordenados por data (01/01/2000-12/31/2010), e que polo tanto podemos precalcular a posición dentro do arquivo, crear estas dúas funcións de consulta:

WeatherData Consulta_por data(File* file, Date d) /*A esta función pásaselle unha data e devolve

unha estrutura cos datos recollidos dese día*/

float Consulta_precip_media(File* file, Date d1, Date d2) /*A esta función pásaselle dúas datas e

devolve a precipitación media entre elas*/