

## Divide y vencerás

### Objetivos

- Implementar y trazar una versión del algoritmo de ordenación por mezcla (MergeSort).
- Trabajar en grupo.

### Actividades (Equipos de 2 o 3 alumnos)

#### Actividad 1:

El algoritmo de ordenación MergeSort es un algoritmo que sigue una estrategia divide y vencerás para ordenar vectores. Una implementación de este algoritmo es:

```
función mergesort(V:&tipo[n])
    nInf , nSup: natural
    nInf ← n/2
    nSup ← n-nInf
    V1:tipo[nInf+1] // Recuerda definir variable y reservar memoria con una
    V2:tipo[nSup+1] // posición adicional al final del vector. Ayuda: T3, diapo 24.

    si casobase(V) // Implementar el caso n<3
        ordenardirectamente(V)
    si no
        V11,...,nInf ← V1,...,nInf // Esta asignación de vectores en pseudocódigo
        V21,...,nSup ← VnInf+1,...,n // debe realizarse elemento a elemento en C++.
        mergesort(V1)
        mergesort(V2)
        combinar(V1,V2,V) // Ayuda: en C++ pasar tamaños nInf/nSup para V1/V2
    fsi
    // aunque se ha reservado una posición más al final.
función
```

Y una posible implementación de la función combinar es:

```
función combinar(V1:tipo[n1+1], V2:tipo[n2+1], V:&tipo[n1+n2])
    i1, i2, i: natural
    i1 ← 1
    i2 ← 1
    V1n1+1 ← ∞ // En la posición adicional del final se almacenará ∞
    V2n2+1 ← ∞ // ∞ ⇒ 999999999 (por ejemplo)
    para i←1 hasta n1+n2 hacer
        si V1i1 < V2i2
            Vi ← V1i1
            i1 ← i1 + 1
        si no
            Vi ← V2i2
            i2 ← i2 + 1
        fsi
    fpara
función
```

Implementa un algoritmo según los pseudocódigos que se proporcionan. Crea un programa que pida por teclado el tamaño del vector y los elementos de este. Una vez introducidos los valores se imprimirá el vector antes de ordenarlo y después de ordenarlo utilizando el algoritmo MergeSort que se proporciona.

#### Actividad 2:

Reproducir la traza de la diapositiva 15 a la 22 del Tema 3 de teoría usando el depurador GDB y sus expresiones para mostrar las diferentes variables involucradas. Grabar un vídeo breve (2 minutos máximo) mostrando la traza del ejemplo con vuestro código.

\* *Notas:* puedes utilizar la herramienta ScreenCastify con tu cuenta UMH para realizar la grabación y descargar el vídeo (<https://www.screencastify.com/>). Si no se consigue hacer uso de GDB, se valorará con la mitad de puntos realizar la traza a través de la salida estándar (printf, cout...).

## Modo de entrega

La práctica se realizará en equipos de **dos o tres alumnos** y se entregarán los siguientes ficheros con los nombres que se indican.

Archivo comprimido: practica5.zip  
Contenido del archivo: p5.cpp, vídeo de la traza  
El fichero con el código fuente de la actividad debe incluir los nombres de los miembros del equipo.

Todos los componentes del equipo entregarán el archivo practica5.zip en la tarea llamada “Práctica 5: Divide y vencerás”, dentro del campus virtual.

**Fecha fin de entrega:** Domingo, 27 de marzo de 2022 a las 23:59.

## Evaluación

La calificación total de la actividad es 0,15 puntos (0,1 A1 y 0,05 A2).

A continuación, se indica el sistema de evaluación:

- **Opción A: La práctica se entrega durante la sesión de prácticas.**

Cada alumno/-a del equipo entregará la práctica (practica5.zip) en la tarea de la web de la asignatura. Antes de subir la tarea a la web se debe recibir el visto bueno del profesorado y todos los miembros del equipo deben estar presentes y explicar cualquier aspecto que se solicite. Se evaluará cogiendo al azar la práctica de uno de los miembros del equipo, de forma que dicha práctica será la que se corrija.

Si hay algún miembro del equipo que no entrega la práctica en la tarea, no responde correctamente a las preguntas realizadas por el profesorado o no está presente..., no se le calificará según esta opción y puede optar a la calificación según la opción B.

- **Opción B: La práctica se entrega en horario posterior a la sesión de prácticas.**

A esta opción optarán los equipos y miembros de equipos que no cumplen los requisitos para ser evaluados según la opción A. Cada miembro del equipo debe entregar tanto la práctica en la tarea de la web de la asignatura como el programa que funcione correctamente. Se calificará cogiendo al azar la práctica de uno de los miembros del equipo, de forma que dicha práctica será la que se corrija. Los programas deben seguir las especificaciones que se dan.

La calificación máxima que se puede obtener bajo esta Opción B es un 0,15 (0,05 A1 y 0,05 A2).

- **Opción C: 0 puntos**

- El alumno/a:
  - No entrega la práctica en la tarea de la web de la asignatura.
  - No asiste a la sesión de prácticas cumpliendo con las condiciones establecidas.
- El programa no funciona correctamente y no realiza lo que se pide.
- Se detecta copia con otras prácticas. La nota será un 0 en esta práctica para todas las prácticas implicadas, aun cuando la práctica haya sido valorada previamente de forma positiva por parte del profesorado.