3. Merge sort

Merge sort algorithm implemented to search for a student's surname from the combined list.

Start

Function merge
Struct student array[],int
l,mid,r

Int n1 = m- l + 1
Int n2 = r − m
Struct student
left[n1],right[n2]

i++

False | True
i < n1 → left[i] = array[mid + 1 + i]

J = 0

j++
j < n2 → Right[i] = array[mid + 1 + i]
True

False

Int I = 0
Int j = 0
Int k =1

False
I < n1 &&
j < n2

True

strcmp(left[i].surname,
right[j].surname))
<
0||(strcmp(left[i].surname,
right[j].surname))=

False → Array[k] = right[j]
j++

K++

True

Array[k] = left [i]
i++

Start

Function merge_sort
Struct student DT,int left,right

True
Left >=
right → Int mid = (left +
right) /2

False

Merge_sort
(DT[],left ,mid)

Merge_sort
Mid + 1, right

Merge_sort
(left ,mid,right)

End

13

```
                              ┌──────────┐
                ┌────────────│   K++    │◄──────────┐
                │             └──────────┘           │
                ▼                                     │
            ╱─────────╲      True   ┌────────────────┐│
           ╱           ╲───────────►│ Array[k] = left [i] │
          ╱  I < n1     ╲           │ i++                 ├┘
           ╲           ╱            └────────────────┘
            ╲─────────╱
                │
             False
                │
                ▼
                          ┌─────────────────────────────────┐
                ▼◄────────┤                                 │
            ╱─────────╲    True   ┌────────────────┐  ┌────────┐
           ╱           ╲─────────►│ Array[k] = right [j] │─►│  K++   │
          ╱  J < n2     ╲         │ i++                 │  └────────┘
           ╲           ╱          └────────────────┘
            ╲─────────╱
                │
             False
                │
                ▼
           ╭─────────╮
           │   End   │
           ╰─────────╯
```


14