

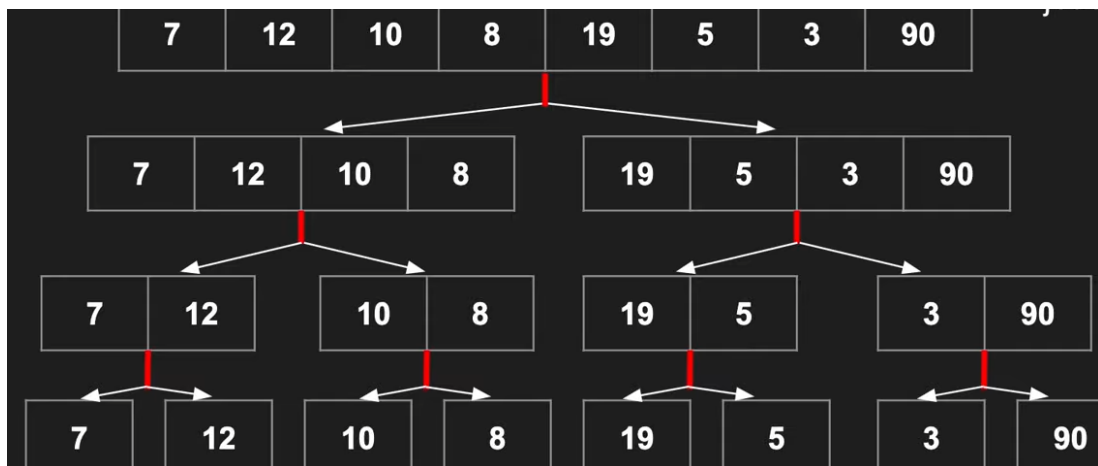
O merge sort é um algoritmo de ordenação que trabalha com recursividade de modo que, utiliza a quebra de um vetor em metades até que não seja possível fazer mais a divisão e faz a remontagem desse vetor em partes, ordenando a cada passo.

```
void mergeSort(ListaClientes *lista, int esquerda, int direita, int* comparacoes, int*
copiasRealizadas)
{
    (*comparacoes)++;
    if (esquerda >= direita)
        return;

    (*copiasRealizadas)++;
    int meio = (esquerda + direita) / 2;

    mergeSort(lista, esquerda, meio, comparacoes, copiasRealizadas);
    mergeSort(lista, meio + 1, direita, comparacoes, copiasRealizadas);

    merge(lista, esquerda, meio, direita, comparacoes, copiasRealizadas);
}
```



Nas chamadas da função mergeSort, o vetor vai sendo quebrado da parte esquerda até o “meio” que está sendo considerado, e desse “meio” até a extremidade direita do vetor.

Uma vez que o vetor está inteiramente fracionado, a função merge é acionada e vai juntar todas as partes do vetor reordenando-as.

