

Tri fusion

```
In [ ]: def partage(liste):
        l1=[]
        l2=[]
        for k in range(len(liste)//2):
            l1.append(liste[k])
        for k in range(len(liste)//2, len(liste)):
            l2.append(liste[k])
        return l1, l2
```

```
In [ ]: liste=[38, 27, 43, 3, 9, 82, 10]
        l1, l2=partage(liste)
```

```
In [ ]: def fusion(l1, l2):
        ind1=0
        ind2=0
        l = []
        while ind1<len(l1) and ind2<len(l2):
            if l1[ind1]<l2[ind2]:
                l.append(l1[ind1])
                ind1+=1
            else:
                l.append(l2[ind2])
                ind2+=1
        if ind1==len(l1):
            for k in range(ind2, len(l2)):
                l.append(l2[k])
        else:
            for k in range(ind1, len(l1)):
                l.append(l1[k])
        return l
```

```
In [ ]: print(fusion(l1, l2))
```

```
In [ ]: def tri_fusion(liste):
        long = len(liste)
        if long <= 1:
            return liste
        else:
            l1, l2 = partage(liste)
            l1 = tri_fusion(l1)
            l2 = tri_fusion(l2)
            return fusion(l1, l2)
```

In []: `print(tri_fusion(liste))`

