MergeSort
Recursion in Python

Thomas Erlebach

Mergesort

- **Divide and conquer**: Split a problem into smaller ones, solve those recursively, use those solution to solve the original problem.
- We want to sort this list:
  17, 12, 34, 15, 28, 27, 14, 5, 18

Mergesort

- **Divide and conquer**: Split a problem into smaller ones, solve those recursively, use those solution to solve the original problem.
- We want to sort this list:
  17, 12, 34, 15, 28, 27, 14, 5, 18
- Let us divide the list into two parts of roughly equal size:
  17, 12, 34, 15 and 28, 27, 14, 5, 18

Mergesort

- **Divide and conquer**: Split a problem into smaller ones, solve those recursively, use those solution to solve the original problem.
- We want to sort this list:
  17, 12, 34, 15, 28, 27, 14, 5, 18
- Let us divide the list into two parts of roughly equal size:
  17, 12, 34, 15 and 28, 27, 14, 5, 18
- Once we have solved these smaller sorting problems: 12, 15, 17, 34 and 5, 14, 18, 27, 28

Mergesort

- **Divide and conquer**: Split a problem into smaller ones, solve those recursively, use those solution to solve the original problem.
- We want to sort this list:
  17, 12, 34, 15, 28, 27, 14, 5, 18
- Let us divide the list into two parts of roughly equal size:
  17, 12, 34, 15 and 28, 27, 14, 5, 18
- Once we have solved these smaller sorting problems: 12, 15, 17, 34 and 5, 14, 18, 27, 28
- We only need to merge the two sorted lists:

Mergesort

- **Divide and conquer**: Split a problem into smaller ones, solve those recursively, use those solution to solve the original problem.
- We want to sort this list:
  17, 12, 34, 15, 28, 27, 14, 5, 18
- Let us divide the list into two parts of roughly equal size:
  17, 12, 34, 15 and 28, 27, 14, 5, 18
- Once we have solved these smaller sorting problems: 12, 15, 17, 34 and 5, 14, 18, 27, 28
- We only need to merge the two sorted lists: 5,

Mergesort

- **Divide and conquer**: Split a problem into smaller ones, solve those recursively, use those solution to solve the original problem.
- We want to sort this list:
  17, 12, 34, 15, 28, 27, 14, 5, 18
- Let us divide the list into two parts of roughly equal size:
  17, 12, 34, 15 and 28, 27, 14, 5, 18
- Once we have solved these smaller sorting problems: 12, 15, 17, 34 and 5, 14, 18, 27, 28
- We only need to merge the two sorted lists: 5, 12,

Mergesort

- **Divide and conquer**: Split a problem into smaller ones, solve those recursively, use those solution to solve the original problem.
- We want to sort this list:
  17, 12, 34, 15, 28, 27, 14, 5, 18
- Let us divide the list into two parts of roughly equal size:
  17, 12, 34, 15 and 28, 27, 14, 5, 18
- Once we have solved these smaller sorting problems: 12, 15, 17, 34 and 5, 14, 18, 27, 28
- We only need to merge the two sorted lists: 5, 12, 14,

Mergesort

- **Divide and conquer**: Split a problem into smaller ones, solve those recursively, use those solution to solve the original problem.
- We want to sort this list:
  17, 12, 34, 15, 28, 27, 14, 5, 18
- Let us divide the list into two parts of roughly equal size:
  17, 12, 34, 15 and 28, 27, 14, 5, 18
- Once we have solved these smaller sorting problems: 12, 15, 17, 34 and 5, 14, 18, 27, 28
- We only need to merge the two sorted lists: 5, 12, 14, 15,

Mergesort

- **Divide and conquer**: Split a problem into smaller ones, solve those recursively, use those solution to solve the original problem.
- We want to sort this list:
  17, 12, 34, 15, 28, 27, 14, 5, 18
- Let us divide the list into two parts of roughly equal size:
  17, 12, 34, 15 and 28, 27, 14, 5, 18
- Once we have solved these smaller sorting problems: 12, 15, 17, 34 and 5, 14, 18, 27, 28
- We only need to merge the two sorted lists: 5, 12, 14, 15, 17,

Mergesort

- **Divide and conquer**: Split a problem into smaller ones, solve those recursively, use those solution to solve the original problem.
- We want to sort this list:
  17, 12, 34, 15, 28, 27, 14, 5, 18
- Let us divide the list into two parts of roughly equal size:
  17, 12, 34, 15 and 28, 27, 14, 5, 18
- Once we have solved these smaller sorting problems: 12, 15, 17, 34 and 5, 14, 18, 27, 28
- We only need to merge the two sorted lists: 5, 12, 14, 15, 17, 18,

Mergesort

- **Divide and conquer**: Split a problem into smaller ones, solve those recursively, use those solution to solve the original problem.
- We want to sort this list:
  17, 12, 34, 15, 28, 27, 14, 5, 18
- Let us divide the list into two parts of roughly equal size:
  17, 12, 34, 15 and 28, 27, 14, 5, 18
- Once we have solved these smaller sorting problems: 12, 15, 17, 34 and 5, 14, 18, 27, 28
- We only need to merge the two sorted lists: 5, 12, 14, 15, 17, 18, 27,

Mergesort

- **Divide and conquer**: Split a problem into smaller ones, solve those recursively, use those solution to solve the original problem.
- We want to sort this list:
  17, 12, 34, 15, 28, 27, 14, 5, 18
- Let us divide the list into two parts of roughly equal size:
  17, 12, 34, 15 and 28, 27, 14, 5, 18
- Once we have solved these smaller sorting problems: 12, 15, 17, 34 and 5, 14, 18, 27, 28
- We only need to merge the two sorted lists: 5, 12, 14, 15, 17, 18, 27, 28,

Mergesort

- **Divide and conquer**: Split a problem into smaller ones, solve those recursively, use those solution to solve the original problem.
- We want to sort this list:
  17, 12, 34, 15, 28, 27, 14, 5, 18
- Let us divide the list into two parts of roughly equal size:
  17, 12, 34, 15 and 28, 27, 14, 5, 18
- Once we have solved these smaller sorting problems: 12, 15, 17, 34 and 5, 14, 18, 27, 28
- We only need to merge the two sorted lists: 5, 12, 14, 15, 17, 18, 27, 28, 34

Mergesort in Python

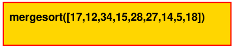
```python
def mergesort(L):
    if len(L) <= 1:
        return L
    else:
        mid = len(L) // 2
        L1 = mergesort(L[:mid])
        L2 = mergesort(L[mid:])
        return merge(L1,L2)

print(mergesort([17,12,34,15,28,27,14,5,18]))
```

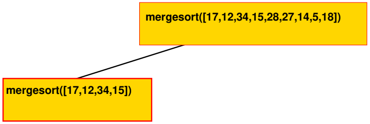
Merging two sorted lists

```python
def merge(L1,L2):
    L=[]
    while L1 and L2:
        if L1[0]<= L2[0]:
            L.append(L1.pop(0))
        else:
            L.append(L2.pop(0))
    L.extend(L1)
    L.extend(L2)
    return L
```

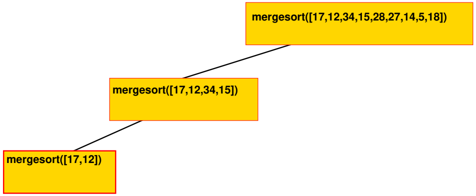
Visualisation of the calls to mergesort: recursion tree



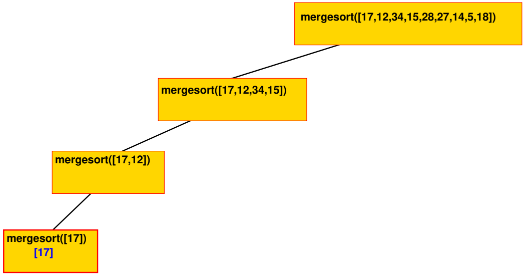
Visualisation of the calls to mergesort: recursion tree



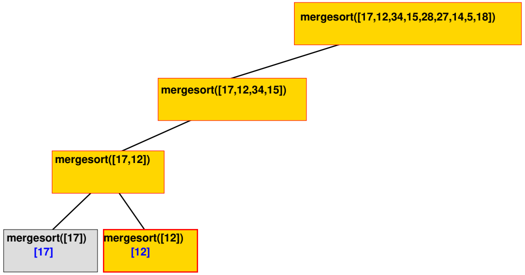
Visualisation of the calls to mergesort: recursion tree



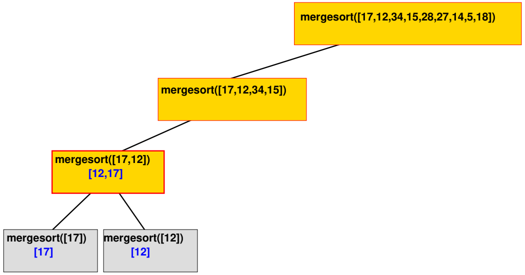
Visualisation of the calls to mergesort: recursion tree



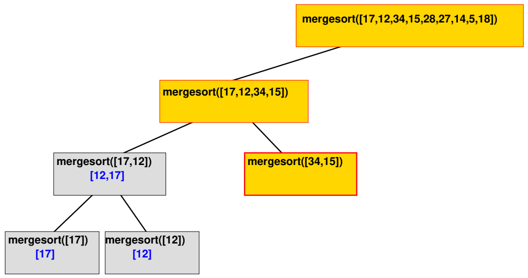
Visualisation of the calls to mergesort: recursion tree



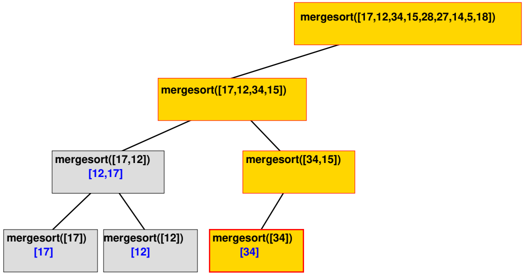
Visualisation of the calls to mergesort: recursion tree



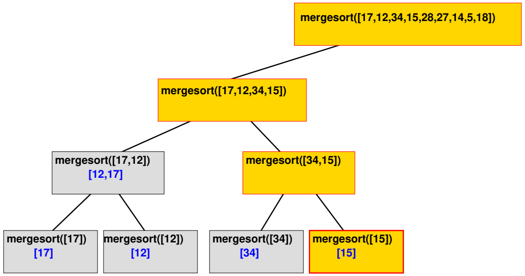
Visualisation of the calls to mergesort: recursion tree



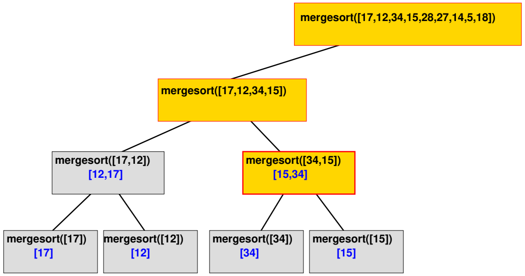
Visualisation of the calls to mergesort: recursion tree



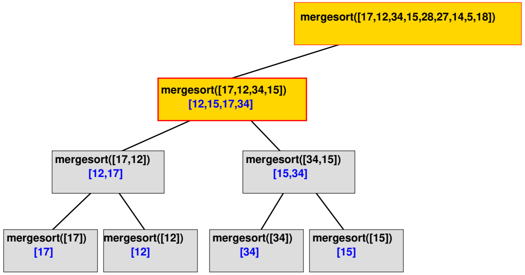
Visualisation of the calls to mergesort: recursion tree



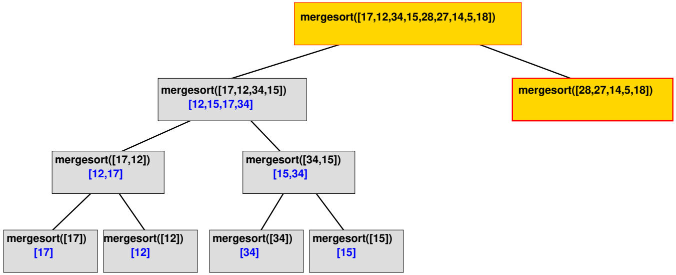
Visualisation of the calls to mergesort: recursion tree



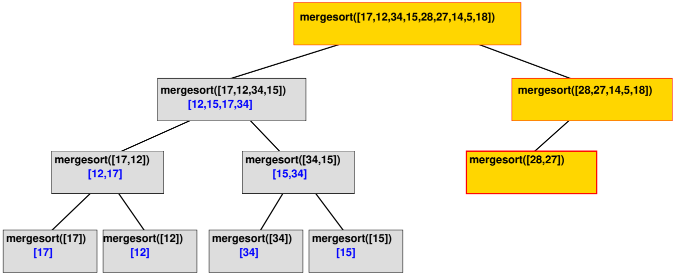
Visualisation of the calls to mergesort: recursion tree



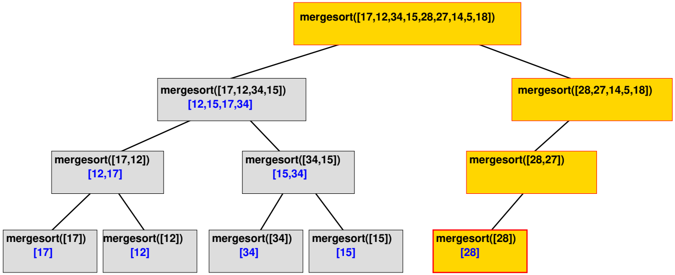
Visualisation of the calls to mergesort: recursion tree



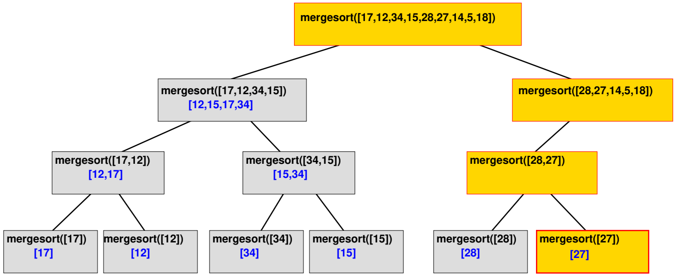
Visualisation of the calls to mergesort: recursion tree



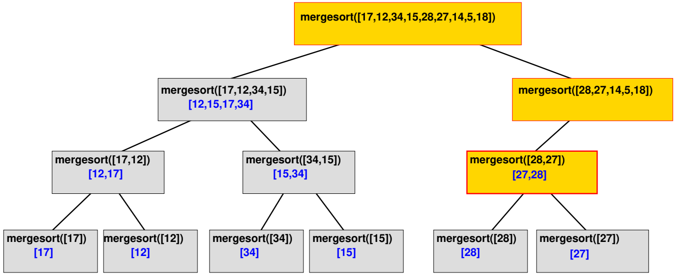
Visualisation of the calls to mergesort: recursion tree



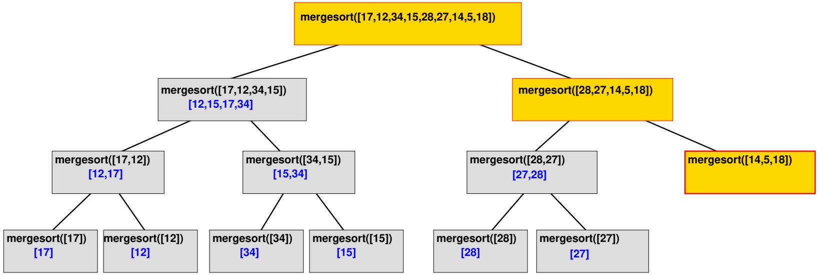
Visualisation of the calls to mergesort: recursion tree



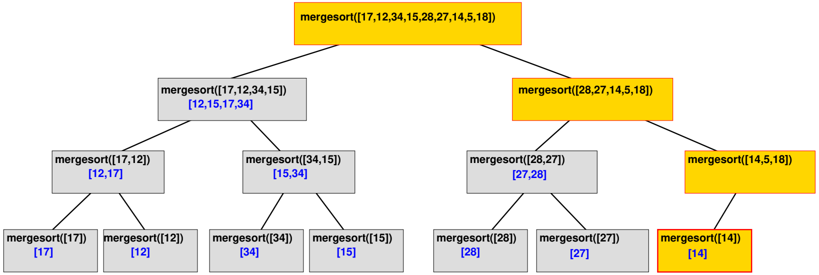
Visualisation of the calls to mergesort: recursion tree



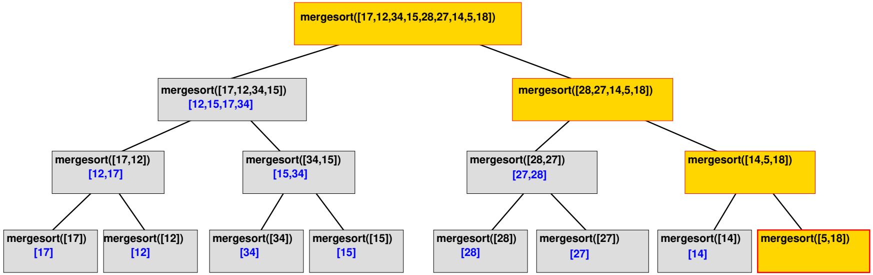
Visualisation of the calls to mergesort: recursion tree



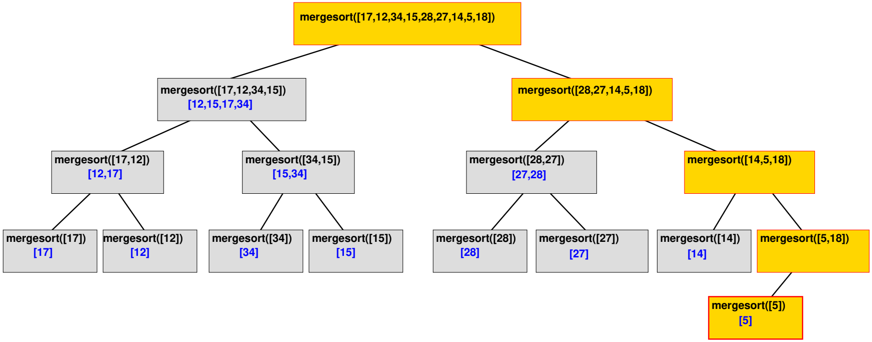
Visualisation of the calls to mergesort: recursion tree



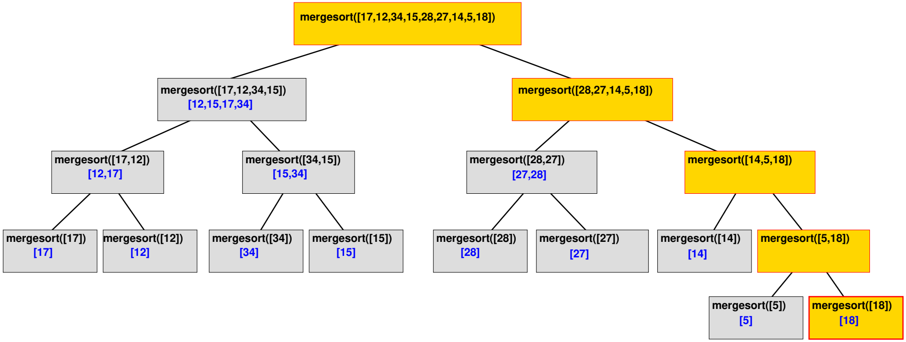
Visualisation of the calls to mergesort: recursion tree



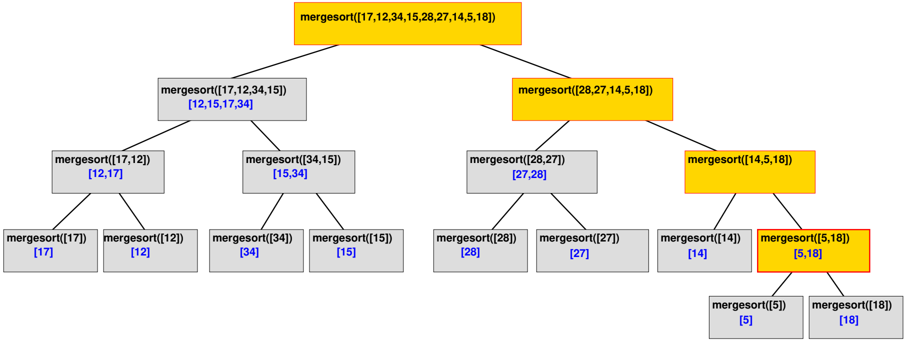
Visualisation of the calls to mergesort: recursion tree



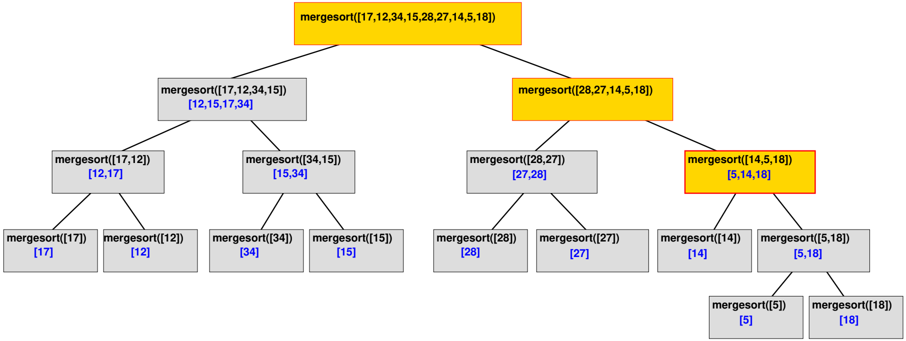
Visualisation of the calls to mergesort: recursion tree



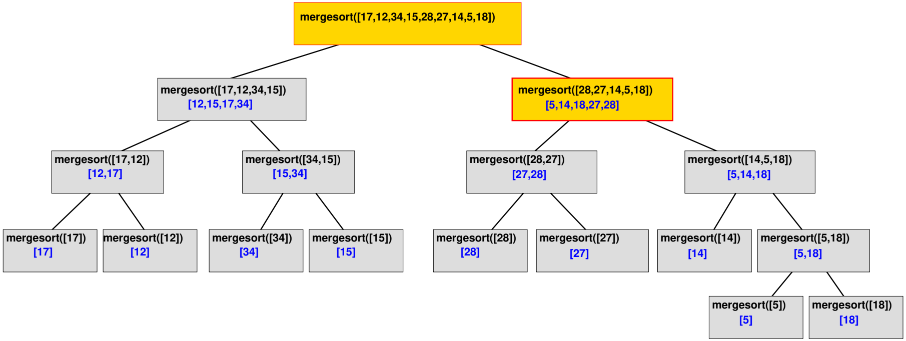
Visualisation of the calls to mergesort: recursion tree



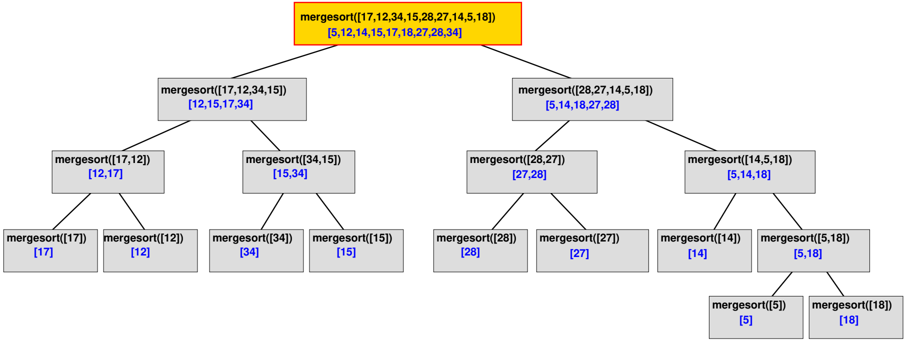
Visualisation of the calls to mergesort: recursion tree



Visualisation of the calls to mergesort: recursion tree



Visualisation of the calls to mergesort: recursion tree


mergesort([17,12,34,15,28,27,14,5,18])
[5,12,14,15,17,18,27,28,34]
mergesort([17,12,34,15])
[12,15,17,34]
mergesort([28,27,14,5,18])
[5,14,18,27,28]
mergesort([17,12])
[12,17]
mergesort([34,15])
[15,34]
mergesort([28,27])
[27,28]
mergesort([14,5,18])
[5,14,18]
mergesort([17])
[17]
mergesort([12])
[12]
mergesort([34])
[34]
mergesort([15])
[15]
mergesort([28])
[28]
mergesort([27])
[27]
mergesort([14])
[14]
mergesort([5,18])
[5,18]
mergesort([5])
[5]
mergesort([18])
[18]