# Merge Sort

Von Neumann, 1945

# Pseudo-Code

```
procedure MergeSort(A,p,r)
    if p<r then
        q ← ⌊(p+r)/2⌋
        MergeSort(A,p,q)
        MergeSort(A,q+1,r)
        Merge(A,(p,q),(q+1,r))
    end if
end procedure
```

# Pseudo-Code

```
procedure MergeSort(A,p,r)
    if p<r then
        q ← ⌊(p+r)/2⌋      Find middle of list.
        MergeSort(A,p,q)
        MergeSort(A,q+1,r)
        Merge(A,(p,q),(q+1,r))
    end if
end procedure
```

# Pseudo-Code

```
procedure MergeSort(A,p,r)
    if p<r then
        q ← ⌊(p+r)/2⌋          Find middle of list.
        MergeSort(A,p,q)         Sort left half.
        MergeSort(A,q+1,r)
        Merge(A,(p,q),(q+1,r))
    end if
end procedure
```

# Pseudo-Code

```
procedure MergeSort(A,p,r)
    if p<r then
        q ← ⌊(p+r)/2⌋          Find middle of list.
        MergeSort(A,p,q)          Sort left half.
        MergeSort(A,q+1,r)          Sort right half.
        Merge(A,(p,q),(q+1,r))
    end if
end procedure
```

# Pseudo-Code

```
procedure MergeSort(A,p,r)
    if p<r then
        q ← ⌊(p+r)/2⌋          Find middle of list.
        MergeSort(A,p,q)         Sort left half.
        MergeSort(A,q+1,r)        Sort right half.
        Merge(A,(p,q),(q+1,r))      Merge two halves.
    end if
end procedure
```

# Pseudo-Code

```
procedure MergeSort(A,p,r)
    if p<r then       Check for base case.
        q ← ⌊(p+r)/2⌋       Find middle of list.
        MergeSort(A,p,q)       Sort left half.
        MergeSort(A,q+1,r)       Sort right half.
        Merge(A,(p,q),(q+1,r))       Merge two halves.
    end if
end procedure
```

# Pseudo-Code

```
procedure MergeSort(A,p,r)
    if p<r then        Check for base case.
        q ← ⌊(p+r)/2⌋        Find middle of list.
        MergeSort(A,p,q)        Sort left half.
        MergeSort(A,q+1,r)        Sort right half.
        Merge(A,(p,q),(q+1,r))        Merge two halves.
    end if
end procedure
```

## Divide-and-conquer algorithm

# Analysis

## Tree method

Assume $n$ is a power of 2.

# Analysis for "Workers"

# Analysis for “Workers”

Number of leaves (in complete tree): $[\text{branching factor}]^{[\text{height}]}$

# Analysis for “Workers”

Number of leaves (in complete tree): $[\text{branching factor}]^{[\text{height}]}$

Branching factor: 2

# Analysis for “Workers”

Number of leaves (in complete tree): $[\text{branching factor}]^{[\text{height}]}$

Branching factor: 2

Height: $\lg n$

# Analysis for "Workers"

Number of leaves (in complete tree): $[\text{branching factor}]^{[\text{height}]}$

Branching factor: 2

Height: $\lg n$

Number of leaves: $2^{\lg n} = n$.

# Analysis for “Workers”

Number of leaves (in complete tree): $[\text{branching factor}]^{[\text{height}]}$

Branching factor: 2

Height: $\lg n$

Number of leaves: $2^{\lg n} = n$.

Each worker does 0 comparisons.

# Analysis for “Workers”

Number of leaves (in complete tree): [branching factor]$^{[\text{height}]}$

Branching factor: 2

Height: $\lg n$

Number of leaves: $2^{\lg n} = n$.

Each worker does 0 comparisons.

Total comparisons: 0.

# Analysis for “Vice Presidents”

# Analysis for "Vice Presidents"

$$\sum_{i=0} 2^i \left( \frac{n}{2^i} - 1 \right)$$

# Analysis for “Vice Presidents”

$$\sum_{i=0}^{\lg n - 1} 2^i \left( \frac{n}{2^i} - 1 \right)$$

# Analysis for "Vice Presidents"

$$\sum_{i=0}^{\lg n - 1} 2^i \left( \frac{n}{2^i} - 1 \right) \quad = \quad \sum_{i=0}^{\lg n - 1} (n - 2^i)$$

# Analysis for “Vice Presidents”

$$\begin{aligned}\sum_{i=0}^{\lg n-1} 2^i \left(\frac{n}{2^i} - 1\right) &= \sum_{i=0}^{\lg n-1} (n - 2^i) \\ &= \sum_{i=0}^{\lg n-1} n \; - \; \sum_{i=0}^{\lg n-1} 2^i\end{aligned}$$

# Analysis for "Vice Presidents"

$$
\begin{aligned}
\sum_{i=0}^{\lg n-1} 2^i \left(\frac{n}{2^i} - 1\right) &= \sum_{i=0}^{\lg n-1} (n - 2^i) \\
&= \sum_{i=0}^{\lg n-1} n \; - \; \sum_{i=0}^{\lg n-1} 2^i \\
&= n \sum_{i=0}^{\lg n-1} 1 \; - \; \left(2^{(\lg n-1)+1} - 1\right)
\end{aligned}
$$

# Analysis for "Vice Presidents"

$$
\begin{aligned}
\sum_{i=0}^{\lg n-1} 2^i \left(\frac{n}{2^i} - 1\right) &= \sum_{i=0}^{\lg n-1} (n - 2^i) \\
&= \sum_{i=0}^{\lg n-1} n \; - \; \sum_{i=0}^{\lg n-1} 2^i \\
&= n \sum_{i=0}^{\lg n-1} 1 \; - \; \left(2^{(\lg n-1)+1} - 1\right) \\
&= n \lg n \; - \; \left(2^{\lg n} - 1\right)
\end{aligned}
$$

# Analysis for “Vice Presidents”

$$
\begin{aligned}
\sum_{i=0}^{\lg n-1} 2^i \left(\frac{n}{2^i} - 1\right) &= \sum_{i=0}^{\lg n-1} (n - 2^i) \\
&= \sum_{i=0}^{\lg n-1} n \; - \; \sum_{i=0}^{\lg n-1} 2^i \\
&= n \sum_{i=0}^{\lg n-1} 1 \; - \; \left(2^{(\lg n-1)+1} - 1\right) \\
&= n \lg n \; - \; \left(2^{\lg n} - 1\right) \\
&= n \lg n \; - \; (n - 1)
\end{aligned}
$$

# Analysis for "Vice Presidents"

$$
\begin{aligned}
\sum_{i=0}^{\lg n-1} 2^i \left(\frac{n}{2^i} - 1\right) &= \sum_{i=0}^{\lg n-1} (n - 2^i) \\
&= \sum_{i=0}^{\lg n-1} n \; - \; \sum_{i=0}^{\lg n-1} 2^i \\
&= n \sum_{i=0}^{\lg n-1} 1 \; - \; \left(2^{(\lg n-1)+1} - 1\right) \\
&= n \lg n \; - \; \left(2^{\lg n} - 1\right) \\
&= n \lg n \; - \; (n - 1) \\
&= n \lg n \; - \; n \; + \; 1
\end{aligned}
$$

# Total comparisons

$$
\begin{aligned}
& 0 + (n \lg n - n + 1) \\
= {} & n \lg n - n + 1
\end{aligned}
$$