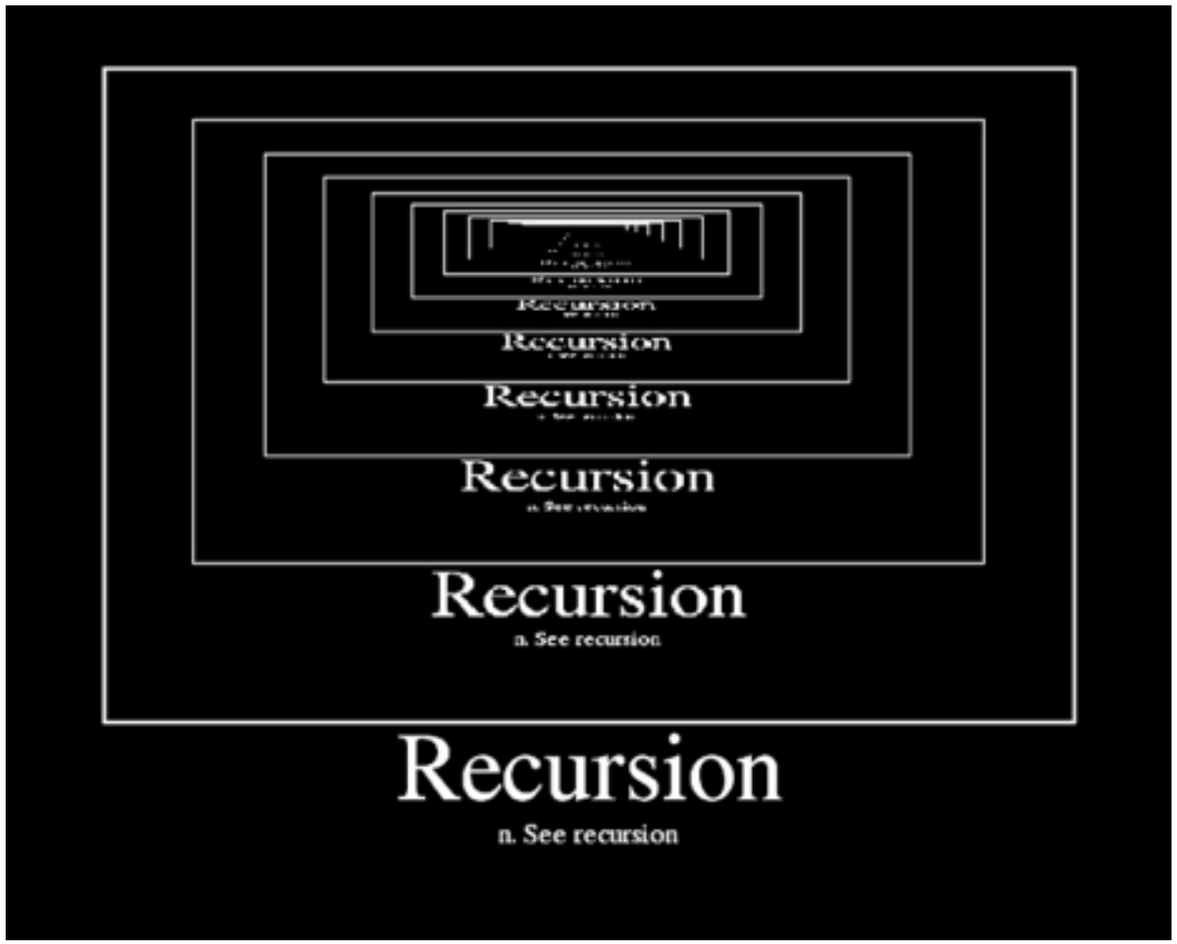
Divisão e Conquista

Carlos Alberto Ynoguti



Atividade em grupos de 4

1. Considere fazer esta atividade com uma balança e, digamos, 32 pesos idênticos, à exceção de um. Agora pense: quantas pesagens adicionais seriam necessárias se houvessem 64 pesos?

2. Se houvessem 2048 caixas, quantas pesagens este processo iria requerer?

3. Em geral, se você dobrar o número de presentes, quantas pesagens adicionais serão necessárias?

4. Quantas pesagens se houverem 4 vezes mais presentes?

5. Quantas pesagens se houverem 1024 vezes mais presentes?

6. Quantos presentes haveriam se houvessem 1024 vezes mais do que na história? Quantas pesagens seriam necessárias neste caso para encontrar a caixa com as meias sujas?

Atividade em grupos de 4

1. Considere fazer esta atividade com uma balança e, digamos, 32 pesos idênticos, à exceção de um. Agora pense: quantas pesagens adicionais seriam necessárias se houvessem 64 pesos? (1 a mais)

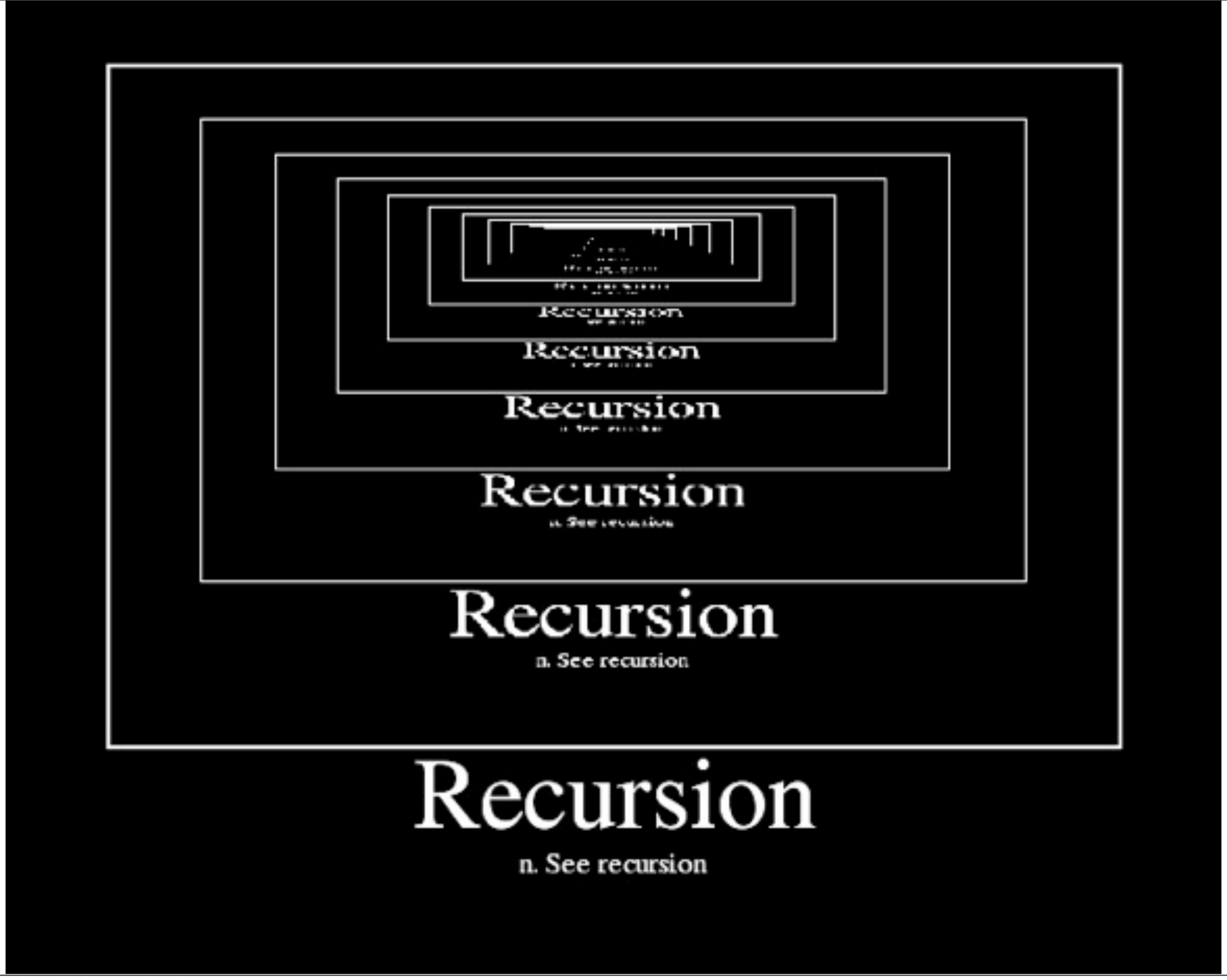
2. Se houvessem 2048 caixas, quantas pesagens este processo iria requerer? (11)

3. Em geral, se você dobrar o número de presentes, quantas pesagens adicionais serão necessárias? (1)

4. Quantas pesagens se houverem 4 vezes mais presentes? (2)

5. Quantas pesagens se houverem 1024 vezes mais presentes? (10)

6. Quantos presentes haveriam se houvessem 1024 vezes mais do que na história? Quantas pesagens seriam necessárias neste caso para encontrar a caixa com as meias sujas? (1024x1024, 10+10)


Recursion
n. See recursion
Recursion
n. See recursion
Recursion
n. See recursion
Recursion
n. See recursion
Recursion
n. See recursion
Recursion
n. See recursion

Divisão e conquista - estratégia

1. Quebrar o problema em subproblemas que são instâncias menores do mesmo tipo de problema.

2. Resolver estes problemas recursivamente.

3. Combinar as respostas parciais de forma adequada.

Divisão e conquista - estratégia

1. Quebrar o problema em subproblemas que são instâncias menores do mesmo tipo de problema.

2. Resolver estes problemas recursivamente.

3. Combinar as respostas parciais de forma adequada.

Um exemplo: Merge Sort

- Suponha que temos um vetor *A* de n elementos para ordenar.
- Algoritmo:
  - dividir *A* em duas partes de mesmo tamanho (ou possivelmente com a diferença de um elemento) e copiar a primeira metade para o vetor B e a segunda metade para o vetor *C*;
  - ordenar os vetores B e *C*;
  - unir os vetores B e *C* ordenados de modo a gerar um vetor ordenado.

```
algoritmo mergesort(A[0..n-1])
se (n>1)
    copie A[0.. ⌊n/2⌋ -1] para B[0.. ⌊n/2⌋ -1]
    copie A[ ⌊n/2⌋ ..n-1] para C[0.. ⌊n/2⌋ -1]
    mergesort(B[0.. ⌊n/2⌋ -1])
    mergesort(C[0.. ⌊n/2⌋ -1])
    merge(B,C,A)
```

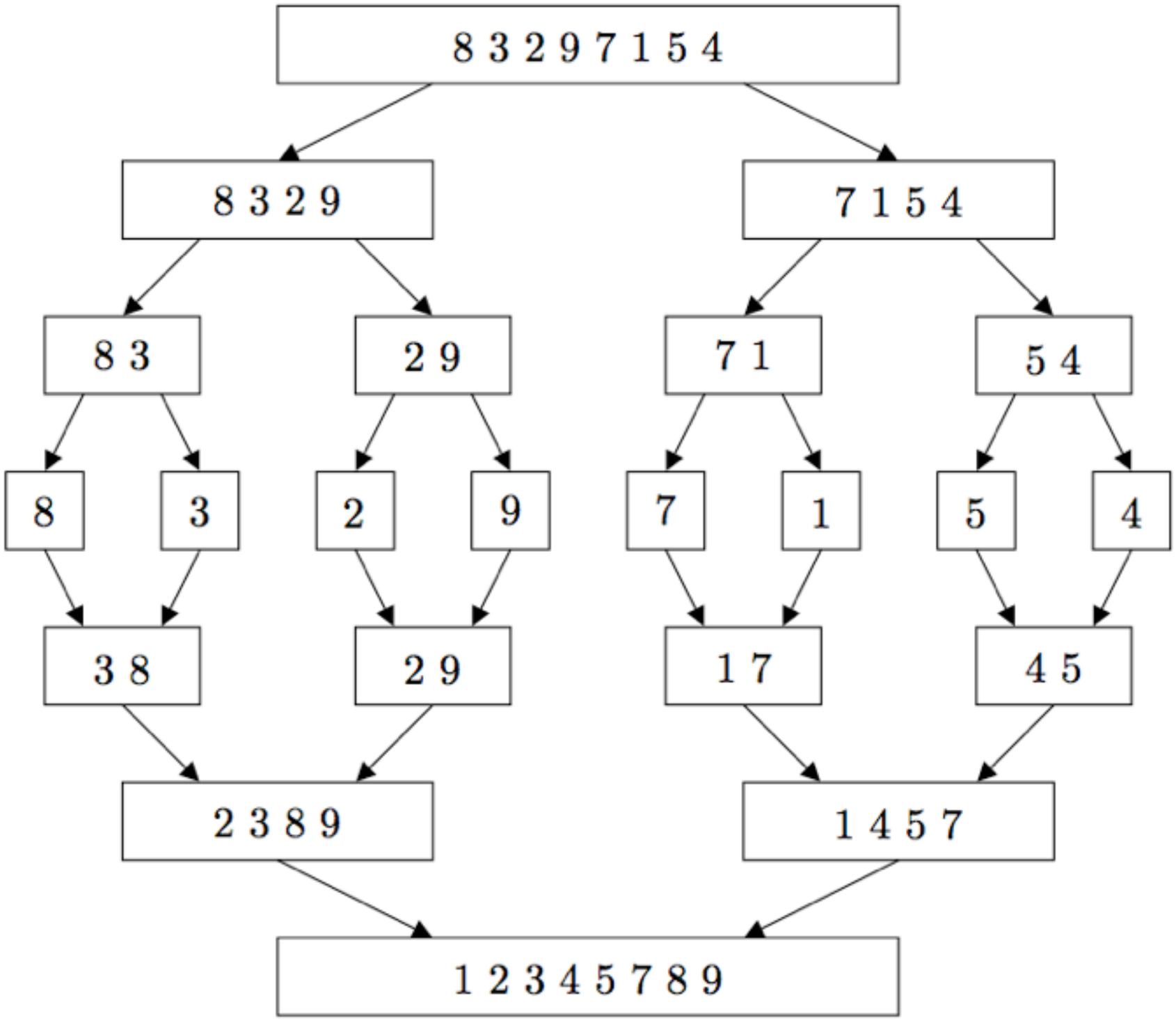
Rotina merge

- enquanto houver elementos em B e C:
  - se o primeiro elemento de B for menor ou igual ao primeiro elemento de C, mova o primeiro elemento de B para o vetor A;
  - caso contrário, mova o primeiro elemento de C para o vetor A;

- quando acabarem os elementos de um vetor, copiar os elementos que sobraram do outro para o vetor A.

```
algoritmo merge(B[0..p-1],C[0..q-1],A[0..p+q-1])
 i=0; j=0; k=0;
enquanto (i<p e j<q)
    se (B[i]<=C[j])
      A[k++] = B[i++];
    senão
        A[k++] = C[j++];
se (i==p)
    copie C[j..q-1] para A[k..p+q-1]
senão
    copie B[i..p-1] para A[k..p+q-1]
```

Exemplo


8 3 2 9 7 1 5 4
8 3 2 9
7 1 5 4
8 3
2 9
7 1
5 4
8
3
2
9
7
1
5
4
3 8
2 9
1 7
4 5
2 3 8 9
1 4 5 7
1 2 3 4 5 7 8 9