zad 3

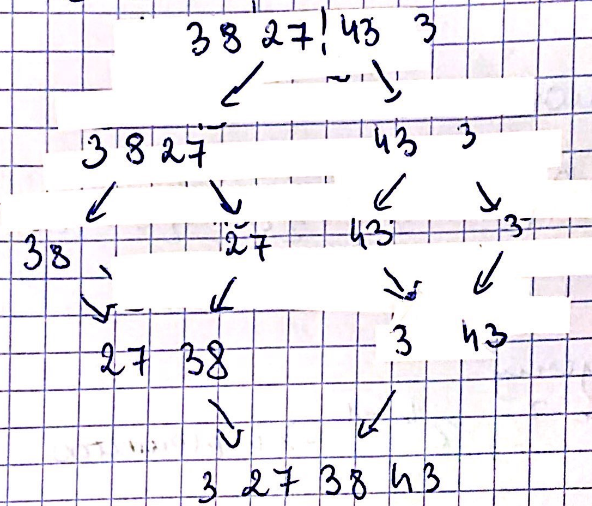
merge_sort → przykład algorytmu z zastosowaniem dziel i zwyciężaj

dziel: dzielimy $n$ elementowy ciąg na dwa podciągi po $\frac{n}{2}$ elementów każdy

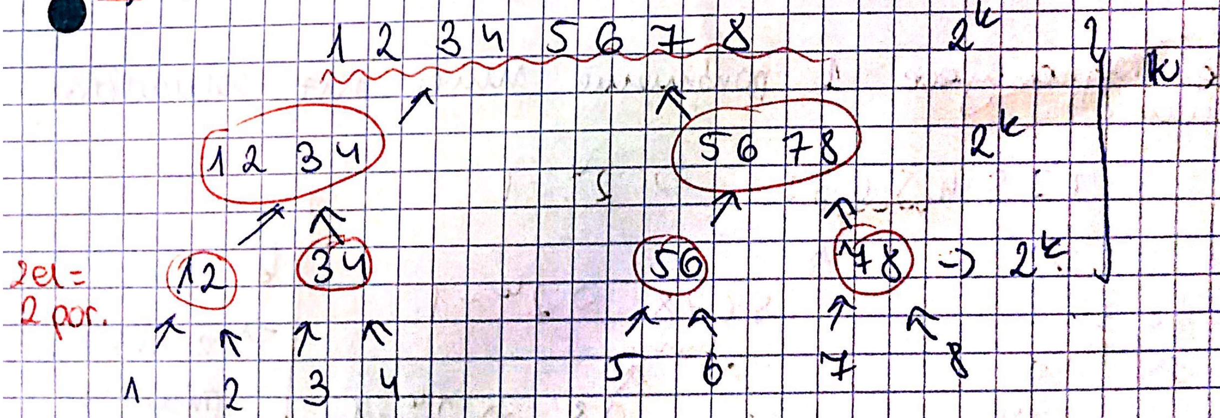
zwyciężaj: sortujemy otrzymane podciągi w jeden posortowany ciąg

połącz: łączymy posortowane podciągi w jeden ciąg

np.




a) $n = 2^k$




procedura merge wykonuje dokładnie tyle porównań ile jest elementów po scaleniu (liści nie bierzemy pod uwagę)

na każdym poziomie zawsze będzie $2^k$ elementów

l. porównań $= 2^k \cdot k$

b) ile razy jest wywołana procedura merge?
pom
"wywoływane jest przy każdej "gałęzi" (jej wynik [illegible])
k∈n.

$\frac{n}{4}$ $\frac{n}{2}$ $\frac{n}{8}$ $\frac{n}{2}$ $\frac{n}{4}$ $\frac{n}{2}$

1 2 3 4 5 6 7 8 merge → 1 × merge
1 2 3 4 merge 5 6 7 8 merge → 2 × merge
1 2 merge 3 4 merge 5 6 merge 7 8 merge → 4 × merge
1 2 3 4 5 6 7 8

(4)

Procedura merge wykona się
: $\frac{n}{2}+\frac{n}{4}+\frac{n}{8}+\ldots+\frac{n}{n} = \frac{2^k}{2}+\frac{2^k}{2^2}+\frac{2^k}{2^3}+\ldots+\frac{2^k}{2^k} =$
$= 2^{k-1}+2^{k-2}+2^{k-3}+\ldots+2^0$
powstaje ciąg geometryczny
$1, \ldots, 2^{k-3}, 2^{k-2}, 2^{k-1}$ → k elementów
$a_1 = 1 \quad q = 2$
$S_k = 1\cdot\frac{1-2^k}{1-2} = 1\cdot\frac{1-2^k}{-1} = -1+2^k = 2^k-1 = n-1$

dla merge wykonujące 1 porównanie mniej niż zakładaliśmy (niż podpunkt a)

1 2 3 4 5 6 7 8 merge → $2^k - 1$
1 2 3 4 merge 5 6 7 8 merge → $2^k - 1$
1 2 merge 3 4 merge 5 6 merge 7 8 merge → $2^k - 1$
1 2 3 4 5 6 7 8

k
−1 z każdego merge'a

l. porównań $= \underbrace{2^k \cdot k}_{\text{z punktu a)}} - \underbrace{(n-1)}_{\text{ilość merge}} = 2^k\cdot k - n + 1 = n\cdot k - n + 1 =$
$= n(k-1)+1$