

Zadanie 7.**Wiązka zadań *Scalanie***

Rozważmy następujący algorytm, który jako dane przyjmuje tablicę n -elementową, gdzie n jest potęgą dwójki:

Dane: tablica liczb rzeczywistych $T[1..n]$, gdzie $n = 2^m$, a m jest liczbą całkowitą nieujemną

funkcja uporządkuj($T[1..n]$):

jeżeli $n=1$

zwróć $T[1..n]$ i zakończ

$k \leftarrow n/2$

$A[1..k] \leftarrow \text{uporządkuj}(T[1..k])$

$B[1..k] \leftarrow \text{uporządkuj}(T[k+1..n])$

zwróć $\text{scal}(A, B)$ i zakończ

Funkcja $\text{scal}(A, B)$ dla danych dwóch tablic o rozmiarze k zwraca tablicę o rozmiarze $2k$, powstałą przez połączenie tablic A i B w sposób uporządkowany, tj. od elementu najmniejszego do największego. Na przykład dla tablic $A = [4, 6, 18, 22]$ i $B = [1, 3, 10, 15]$ wywołanie $\text{scal}(A, B)$ zwróci tablicę $[1, 3, 4, 6, 10, 15, 18, 22]$.

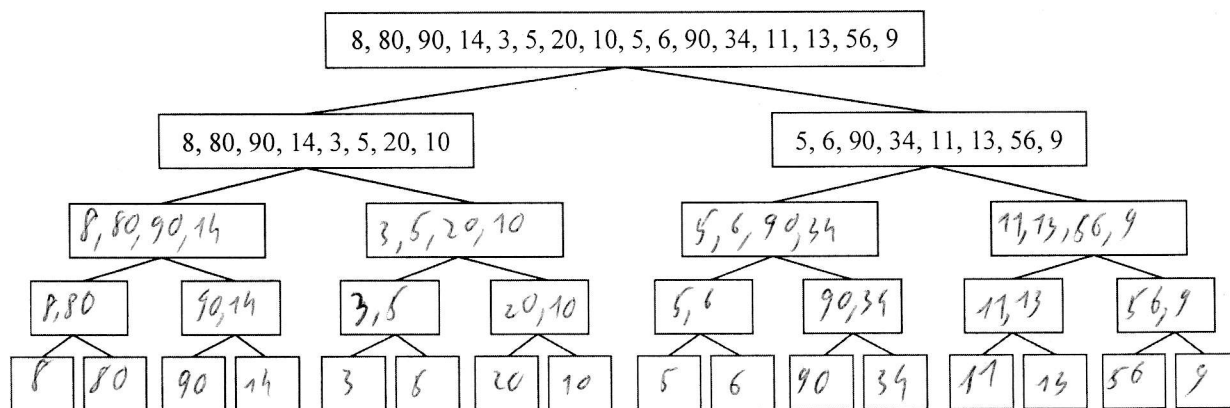
7.1.

Pośród danych tablic wybierz te, które są zgodne ze specyfikacją algorytmu, a następnie podaj dla nich wynik jego działania.

- $T_1, T_2 [1, 15, 1, 2, 5, 6, 9, 10, 15, 10] - A$
- $AB [1, 1, 2, 2, 5, 5, 6, 6, 9, 9, 10, 10, 11, 15, 15, 90, 90]$
- A {
 • $T = [15, 11]$,
 • $T = [1, 3, 8]$,
 • $T = [8, 4, 2, 1]$,
 B } • $T = [10, 15, 1, 6, 9, 2, 5, 90]$.

7.2.

Funkcja *uporządkuj* jest funkcją rekurencyjną. Uzupełnij poniższe drzewo wywołań rekurencyjnych dla danej tablicy $T = [8, 80, 90, 14, 3, 5, 20, 10, 5, 6, 90, 34, 11, 13, 56, 9]$.



7.3.

Założmy, że wywołanie procedury $scal(A,B)$ dla dwóch tablic o długości k wykonuje $2k-1$ kosztownych operacji (porównywania liczb). Podaj liczbę kosztownych operacji, jaka zostanie wykonana przez funkcję *uporządkuj* dla tablicy

$$T[1..16] = [8, 80, 90, 14, 3, 5, 20, 10, 5, 6, 90, 34, 11, 13, 56, 9].$$

15 operacji

Zadanie 8.

Wiązka zadań *Dwa ciągi*

Niech $A[1..n]$ i $B[1..n]$ będą uporządkowanymi rosnąco tablicami liczb całkowitych i niech x będzie liczbą całkowitą. Rozważmy następujący algorytm:

```

 $i \leftarrow 1$ 
 $j \leftarrow n$ 
dopóki ( $i \leq n$  oraz  $j > 0$ ) wykonuj

    dopóki ( $i \leq n$  oraz  $j > 0$ ) wykonuj
    (*) jeżeli  $A[i] + B[j] = x$ 
        podaj wynik PRAWDA i zakończ algorytm
    w przeciwnym razie
        jeżeli  $A[i] + B[j] < x$ 
             $i \leftarrow i + 1$ 
        w przeciwnym razie
             $j \leftarrow j - 1$ 
    podaj wynik FAŁSZ

```

8.1.

Uzupełnij poniższą tabelę. Podaj wynik działania algorytmu oraz liczbę porównań wykonanych w wierszu oznaczonym (*).

Tablica A	Tablica B	x	Wynik działania algorytmu	Liczba porównań w kroku (*)
3, 5, 12, 17	8, 10, 13, 14	21		
4, 6, 8, 10	5, 7, 9, 11	13		

8.2.

Przeanalizuj działanie zaprezentowanego algorytmu i uzupełnij poniższą specyfikację.

Dane:

n — dodatnia liczba całkowita

$A[1..n]$, $B[1..n]$ — n -elementowe tablice liczb całkowitych, posortowane rosnąco

x — liczba całkowita

Wynik:

PRAWDA, gdy

FAŁSZ, gdy