



Zadanie E4

Tablica A i B mają po n elementów, przy czym tablica A jest posortowana nierosnąco, a tablica B niemalejąco. Dysponujesz procedurą

`void zamień(int *A, int *B, int i),`

która zamienia element $A[i]$ z $B[i]$ dla $0 \leq i < n$. Jest to jedyna procedura, za pomocą której możesz modyfikować wartości tablic.

Napisz procedurę

`int rozdziel(int *A, int *B, int n),`

która wyznaczy możliwie małą liczbę zamian prowadzących do sytuacji, w której wszystkie elementy jednej z tablic będą większe lub równe od wszystkich elementów drugiej z nich.

Na przykład dla tablic o wartościach A: 8 8 8 8 7 1 1 oraz B: 1 2 3 4 5 9 9 wystarczy zrobić zamianę dwóch ostatnich elementów, więc wynikiem powinno być 2.



Zadanie E5

Zadanie E5

Na pręcie o numerze 1 znajdują się kolejno od dołu białe krążki o rozmiarach $n, n-1, \dots, 1$, a na pręcie o numerze 2 czarne krążki o takich samych rozmiarach $n, n-1, \dots, 1$. Napisz procedurę

```
void zamiana(int n) //  $n \geq 0$ ,
```

która zamieni wieże miejscami tak, aby na pręcie o numerze 1 były wszystkie czarne krążki, a na pręcie o numerze 2 wszystkie białe. Krążki przenosimy zgodnie z regułami wież Hanoi: w każdym ruchu po jednym i nie wolno większego krążka położyć na mniejszym. Kolor nie ma znaczenia. Rozwiązanie powinno wypisywać na standardowe wyjście ciąg poleceń w postaci tekstów

„przenieś krążek z wieży k na wieżę m ”, gdzie k oraz m są ze zbioru $\{1, 2, 3\}$ i k jest różne od m .



Zadanie E6

Zadanie E6

Węzeł w [drzewa](#) d nazwiemy *centralnym*, jeśli jego odległość od najgłębszego potomka jest równa jego odległości od korzenia. Innymi słowy, węzeł w o głębokości g jest centralny, jeśli najgłębszy liść powieszony pod w ma głębokość $2 \cdot g$. Napisz funkcję

```
int IleCentralnych(Twezel *d),
```

która wyznaczy liczbę centralnych węzłów [drzewa](#) d . Przykładowo pełne drzewo binarne o wysokości 2^h ma 2^h węzłów centralnych.