

Kolokwium nr 2**Zadanie 1**

Dane są tablice $\text{int } A[n]$ oraz $\text{int } B[n]$, $n > 0$, uporządkowane ściśle rosnąco. Dodatkowo różnice ~~każdego~~ par kolejnych elementów w A są uporządkowane ściśle rosnąco, zaś w B ściśle malejąco, tzn. dla każdego j : $0 \leq j \leq n-3$, mamy $A[j+1] - A[j] < A[j+2] - A[j+1]$, zaś $B[j+1] - B[j] > B[j+2] - B[j+1]$. Napisz funkcję

```
bool tesame(int n, int A[n], int B[n])
```

która przyjmie wartość `true` jeżeli istnieje i takie, że $0 \leq i \leq n-1$ oraz $A[i] = B[i]$, a wartość `false` w przeciwnym przypadku.

Zadanie 2

Krażki na wieżach Hanoi są pokolorowane na zielono lub czerwono. Wieża o numerze 0 jest bezbarwna i początkowo zawiera n krażków numerowanych od 0 do $n-1$. Dwie pozostałe wieże są pokolorowane: wieża o numerze 1 ma kolor zielony, a wieża o numerze 2 kolor czerwony. Tablica $\text{int } K[n]$ zawiera informację o kolorach krażków: pod indeksem j mamy kolor j -tego krażka. Kolor zielony jest kodowany przez 1, kolor czerwony przez 2.

Napisz procedurę

```
void KolorHanoi(int n, int K[n]),
```

która przestrzegając reguł obowiązujących w niepokolorowanych wieżach Hanoi umieści wszystkie zielone krażki na zielonej wieży, a czerwone na czerwonej.

Zadania oddajemy na osobnych kartkach czytelnie podpisane (proszę podać też inicjały prowadzącego ćwiczenia ze zbioru AZ, KG, MD, MK, OS, PCh, TW) i skomentowane. Każde rozwiązanie należy uzasadnić i podać jego koszt. Niezmienniki pętli bardzo mile widziane. Można użyć napisanych przez siebie funkcji i procedur pomocniczych.

Przypominamy, że interesuje nas nie tylko sam poprawny kod, ale i jego złożoność. Przede wszystkim staramy się zminimalizować koszt czasowy. W szczególności nie należy się spodziewać zbyt wielu ($>30\%$) punktów za zrobienie pierwszego zadania w czasie $O(n)$.