

Kolokwia. Semestr I.

Poniższe zadania są wyborem zadań ze Wstępu do Informatyki z kolokwiów jakie przeprowadziłem w ciągu ostatnich lat. Zadania obejmują materiał pierwszego semestru.

Marek Zawadowski

Kolokwium ze Wstępu do Informatyki. 9 grudnia 2019.

Zadanie 1

Napisz funkcję, która dla liczby całkowitej dodatniej złożonej n zwróci iloczyn dwóch największych liczb pierwszych dzielących n .

Dla liczby $2926 = 2 * 7 * 11 * 19$ funkcja powinna zwrócić $209 = 11 * 19$.

Zadanie 2

Napisz funkcję, która dla tablicy a zawierającej n liczb całkowitych i liczby k sprawdzi czy w tablicy a jest k sąsiadujących ze sobą elementów uporządkowanych ściśle rosnąco, tzn. zwróci 1, gdy taki ciąg elementów istnieje, i zwróci 0 w przeciwnym przypadku.

Dla tablic $a = [12, 9, 7, 8, 23, 7]$, $a = [12, 9, 7, 8, 23, 241, 7]$ i liczby $k = 3$ funkcja powinna zwrócić 1, a dla tablic $a = [12, 9, 7, 8, 8, 23, 241, 7]$, $a = [12, 13, 14, 8, 5, 6, 7, 7]$, i liczby $k = 4$ funkcja powinna zwrócić 0.

Zadanie 3

Sformułuj problem znajdowania najdłuższego wspólnego podciągu. Opisz algorytm znajdowania najdłuższego wspólnego podciągu. Jaka jest złożoność tego algorytmu? Zdefiniuj klasę funkcji $\Omega(f(n))$ dla funkcji $f : N \rightarrow N$.

Kolokwium ze Wstępu do Informatyki. 10 grudnia 2018.

Zadanie 1

Napisz funkcję, która dla dwóch liczb całkowitych dodatnich m i n zwróci **1**, gdy mają te same dzielniki pierwsze i **0** w przeciwnym przypadku. Dla par liczb 2, 64 oraz 63, 147 funkcja powinna zwrócić **1** a dla par 10, 12 oraz 18, 27 funkcja powinna zwrócić **0**.

Zadanie 2

Napisz funkcję, która dla tablicy zawierającej liczby całkowite przestawi tak elementy tej tablicy by element najmniejszy w tablicy stał na początku, a element największy na końcu. Dla tablicy $A = [12, -7, 18, 2, -2, -11, 7]$ funkcja powinna zwrócić tablicę $A = [-11, -7, 7, 2, -2, 12, 18]$. Rozwiązania używające dodatkowej tablicy lub o złożoności większej niż $O(n)$ będą oceniane w skali od 0 do 5 punktów.

Zadanie 3

Sformułuj problem sortowania ciągu liczb. Opisz algorytm sortowania przez scalanie. Jaka jest złożoność tego algorytmu? Czy algorytm sortujący przy pomocy porównań może mieć lepszą pesymistyczną czasową złożoność obliczeniową?

Opisz jak działa algorytm scalania na tablicach $A = [10, 12, 17, 20, 35]$ i $B = [3, 4, 11, 15, 19]$.

Kolokwium ze Wstępu do Informatyki. 11 grudnia 2017.

Zadanie 1

Napisz funkcję, która dla dwóch liczb całkowitych dodatnich m i n zwróci **true**, gdy mają one nieparzystą liczbę wspólnych dzielników pierwszych oraz zwróci **false**, w przeciwnym przypadku.

Dla liczb par 35, 49 oraz 48, 60 funkcja powinna zwrócić **true** a dla par 13, 14 oraz 18, 27 **false**.

Zadanie 2

Napisz procedurę która dla danej tablicy typu `tab=array[1..n] of integer` zawierającej wyłącznie liczby 0, 1, 2 tak pozamienia jej miejscami by zera stały przed jedynkami i nie zmieni miejsc jedynek.

Dla tablicy $A = [2, 0, 1, 2, 1, 2, 1, 0, 0]$ procedura powinna zwrócić tablice $A = [0, 0, 1, 0, 1, 2, 1, 2, 2]$.

Rozwiązania używające dodatkowej tablicy lub o złożoności większej niż $O(n)$ będą oceniane w skali od 0 do 5 punktów.

Zadanie 3

Sformułuj problem sortowania ciągu liczb. Zdefiniuj pojecie kopiec? Opisz algorytm sortowania przez kopcowanie. Jaka jest złożoność algorytmu sortowania przez kopcowanie? Opisz jakie zamiany elementów w tablicy $A = [10, 18, 17, 15, 14, 12, 11, 9, 8, 6, 5, 4]$ wykona procedura `kopcuje(1,12)` na tablicy.

Kolokwium ze Wstępu do Informatyki. 12 grudnia 2016.

Zadanie 1

Napisz funkcję, która dla dwóch liczb całkowitych dodatnich m i n zwróci **true**, gdy mają one wspólny dzielnik większy od 1 oraz zwróci **false**, w przeciwnym przypadku.

Zadanie 2

Dana jest tablica A typu **tab1**. Napisz procedurę o nagłówku

```
procedure tablice(var A : tab1; var B : tab2);
```

która wypełni tablicę B typu **tab2** w taki sposób, że $B[i]$ będzie liczbą wyrazów tablicy A o wartościach mniejszych lub równych i . Deklaracje typów

type

```
tab1 = array [1..n] of integer;  
tab2 = array [1..m] of integer;
```

Uwaga. Tablica A nie musi zawierać wyłącznie liczb z przedziału od 1 do m .

Rozwiązania używające dodatkowej tablicy lub o złożoności większej niż $O(n + m)$ będą oceniane w skali od 0 do 5 punktów.

Zadanie 3

Sformułuj problem scalania dwóch tablic. Opisz algorytm sortowania przez scalanie. Jaka jest złożoność algorytmu sortowania przez scalanie? Zdefiniuj klasę funkcji $O(f(n))$ dla funkcji $f : N \rightarrow N$.

Kolokwium ze Wstępu do Informatyki. 7 grudnia 2015.

Zadanie 1

Napisz funkcję, która dla danej liczby całkowitej n większej od 1 zwróci największy dzielnik pierwszy n . W szczególności jeśli n jest liczbą pierwszą funkcja powinna zwrócić n .

Zadanie 2

Napisz funkcję, która dla tablicy n liczb całkowitych A , w której nie wszystkie wartości są równe, zwróci drugi co do wielkości element w tablicy.

Na przykład dla tablicy $A = [2, 3, 5, 11, -1, 11, 8, 4]$, funkcja powinna zwrócić wartość 8.

Zadanie 3

Zdefiniuj co to znaczy, że program I jest poprawny ze względu na warunek początkowy P i warunek końcowy Q . Zdefiniuj klasę funkcji $O(f(n))$ dla funkcji $f : N \rightarrow N$. Jaka jest złożoność algorytmu sortowania przez scalanie?