

Име:..... Ф№:..... Група:....

Задача	1	2	3	4	5	ОБЩО
<i>получени точки</i>						
<i>от максимално</i>	<i>20</i>	<i>10</i>	<i>10</i>	<i>10</i>	<i>10</i>	<i>60</i>

**Зад. 1** Докажете или опровергайте всяко от следните пет твърдения. Дайте кратка аргументация на отговорите си. Отговори без никаква аргументация не се оценяват. Във **в**),  $G \setminus h$  означава  $G$  с премахнато ребро  $h$ .

**а)** Алгоритъмът QUICKSORT е стабилен сортиращ алгоритъм.

**б)** Нека  $G(V, E)$  е ориентиран граф. Нека  $(u, v)$  е произволно ребро от  $E$ , такова че алгоритъмът за обхождане в дълбочина DFS класифицира  $(u, v)$  като ребро настрани. Тогава задължително  $u$  и  $v$  са върхове от различни силно свързани компоненти на  $G$ .

**в)** Следният алгоритъм връща минимално покриващо дърво на произволен свързан тегловен неориентиран граф  $G(V, E, w)$ :

```

сортирай ребрата на  $G$  по тегло
 $h \leftarrow$  най-тежко ребро в  $G$ 
докато  $G \setminus h$  е свързан
    премахни реброто  $h$  от  $G$ 
 $h \leftarrow$  най-тежко ребро в  $G$ 
    
```

**г)** Нека  $G(V, E, w)$  е произволен неориентиран тегловен граф. Нека  $k = \max\{w(e) \mid e \in E\}$ . Нека поне половината от ребрата имат тегло  $k$ . Задължително (т.е., за **всеки** такъв  $G$ ) съществува минимално покриващо дърво, което не съдържа нито едно ребро измежду ребрата с тегло  $k$ .

**д)** Нека  $G(V, E, w)$  е произволен неориентиран тегловен граф, в който теглата на ребрата имат само две възможни стойности: 1 и 2. Нека в  $G$  има Хамилтонов цикъл, върху който едно ребро има тегло 2 и останалите, тегло 1. Можем да изчислим сумарното тегло на произволно минимално покриващо дърво на  $G$  (не самото дърво, а само сумата от теглата на ребрата му) във време  $O(\lg n)$ , където  $n = |V|$ .

Всяка от следващите пет задачи иска да се измисли и опише алгоритъм. Описанието на алгоритмите трябва да бъде свършено ясно, без никакво двусмислие или неясноти. Пишете детайлен псевдокод или словесно описание, от което да е абсолютно ясно как да се напише програма, имплементираща предложениия алгоритъм. Не дискутирайте примери и не разказвайте как бихте

решили задачата за конкретен пример. Описания на алгоритми, които съдържат неясноти или са прекалено недетайлни или използват конкретни примери няма да бъдат оценявани.

Добре е да се даде аргументация за коректността на всеки описан алгоритъм – колкото по-формална и прецизна, толкова по-добре. Задължително е да се даде оценка на сложността по време на всеки алгоритъм. Колкото по-прецизно бъде обоснована дадена оценка за сложност, толкова по-добре.

**Зад. 2** Даден е неориентиран тегловен свързан граф  $G(V, E, w)$ . Предложете колкото е възможно по-бърз (в асимптотичния смисъл) алгоритъм, който изчислява покриващо дърво  $T$  на  $G$ , такова че теглото на най-тежкото ребро в  $T$  е минимално.

*Забележка:* Изучаваните алгоритми за минимални покриващи дървета пресмятат дърво, на което общото тегло е минимално. Тук се иска да се намери дърво, в което се минимизира теглото на най-тежкото ребро.

**Зад. 3** Предложете алгоритъм, който по дадени  $n$  естествени числа, всяко от които е от интервала  $[1, \dots, k]$ , прави първоначална обработка (preprocessing) на числата, след което отговаря на запитвания от типа “Колко числа от дадените са в интервала  $[a, \dots, b]$ ?”. На всяко запитване трябва да се отговори във време  $\Theta(1)$ . Първоначалната обработка трябва да бъде извършена във време  $\Theta(n + k)$ .

**Зад. 4** Предложете алгоритъм с колкото е възможно по-малка сложност по време, който при вход неориентиран граф  $G(V, E)$  връща ДА, ако графът има цикъл с дължина 3, и НЕ, в противен случай.

**Зад. 5** Илюстрирайте работата на алгоритъма QUICKSORT върху следния входен масив:  $[4, 1, 3, 8, 2, 5, 7, 9]$ . За целта напишете ясно като псевдокод алгоритъма и после покажете колкото можете по-подробно как точно се променя този масив, докато стане сортиран.

**Бонус задача - 10 т.** Решете задача 4 при допълнителното условие, че  $G$  е такъв, че всяко ребро участва в най-много един цикъл.

*Упътване:* Използвайте модифициран DFS, който класифицира върховете в нива. Намерете необходимо и достатъчно условие да има цикъл с дължина три, което условие е изразено чрез класификацията на ребрата, която правим в контекста на DFS.

**Бонус задача - 10 т.** Дадено е множество интервали  $J = \{j_1, j_2, \dots, j_n\}$ , където всеки интервал  $j_i$  е наредена двойка  $(s_i, f_i)$  от положителни числа, такива че  $s_i < f_i$ . Казваме, че два интервала  $j_i$  и  $j_k$  се пресичат, ако  $s_i \leq s_k < f_i$  или  $s_k \leq s_i < f_k$ . Всеки интервал  $j_i$  има цена  $c_i$ , която е някакво положително число. Предложете колкото е възможно по-бърз (в асимптотичния смисъл) алгоритъм, който намира подмножество  $S \subseteq J$ , такова че никои два интервала  $x, y \in S$  не се пресичат, и освен това сумата от цените на елементите на  $S$  е максимална. Не е необходимо алгоритмът ви да връща  $S$  като множество интервали, достатъчно е да пресметне  $\sum_{x \in S} c(x)$ .

*Упътване:* Използвайте динамично програмиране. Лакомата стратегия не дава оптимален резултат.