



ТРЕНИРОВЪЧНО СЪСТЕЗАНИЕ НА РАЗШИРЕНИЯ НАЦИОНАЛЕН ОТБОР

София, 19 юли 2025 г

Група С

Задача С3. ГРАФ

0.1 сек. 256 MB

Във вселената съществува пълен претеглен граф с n върха, за който съществуват две стойности – A и B – такива че теглото на всяко ребро е или A , или B . Вие не се вълнувате от целия граф, но ще се радвате да знаете минималното му покриващо дърво. За тази цел Ви е позволено да задавате заявки от следния тип: “За множества S и T от върховете на графа, такива че всеки връх участва в най-много едно от тях, каква е сумата на теглата на всички ребрата, за които единият им връх е в S , а другият в T ?”.

Напишете програма, съдържаща функция `play`, която ще се компилира с програмата на вселената и ще комуникира с нея, задавайки въпроси от гореописания вид. В края на изпълнението си тя трябва да е открила едно примерно минимално покриващо дърво и да репортува неговите ребра.

Детайли по имплементацията

Функцията `std::vector<std::pair<int, int>> play(int n)`, която трябва да напишете, ще бъде извикана само веднъж от програмата на журито и като аргумент ще получи цялото число n – броя върхове в графа. След като откриете кое да е минимално покриващо дърво на графа, вашата функция ще трябва да върне вектор, съдържащ ребрата му. За комуникация с програмата на журито Ви се предоставя следната функция:

```
int query(const std::vector<int>& S,  
const std::vector<int>& T);
```

При всяко извикване на функцията `query`, тя ще върне сумата на ребрата, за които единия край е в S , а другия – в T . Векторите трябва да отговарят на следните условия: да съдържат естествени числа със стойности между 1 и n ; да няма числа, които се съдържат в двата вектора едновременно; да няма числа, които се съдържат многократно в един вектор. Функцията има сложност $O(|S||T|)$. Имате право да използвате функцията най-много 2000 пъти.

Вашата програма трябва да имплементира функцията `play`. Тя може да съдържа и друг код, и функции, необходими за работата Ви, но не трябва да съдържа главната функция `main`. Също така, не трябва да четете от стандартния вход или да отпечатвате на стандартния изход. Програмата Ви трябва да включва хедър файла `graph.h` чрез указание към препроцесора:

```
#include "graph.h"
```

Ограничения

- $1 \leq n \leq 140$
- $1 \leq A < B \leq 1000$

Примерна комуникация

Функция на участника	Програма на журито
	<code>play(4)</code>
<code>query({1}, {2, 3})</code>	2
<code>query({1, 2, 3}, {4})</code>	6
<code>query({2}, {3})</code>	2
<code>return {{1, 2}, {1, 3}, {1, 4}};</code>	

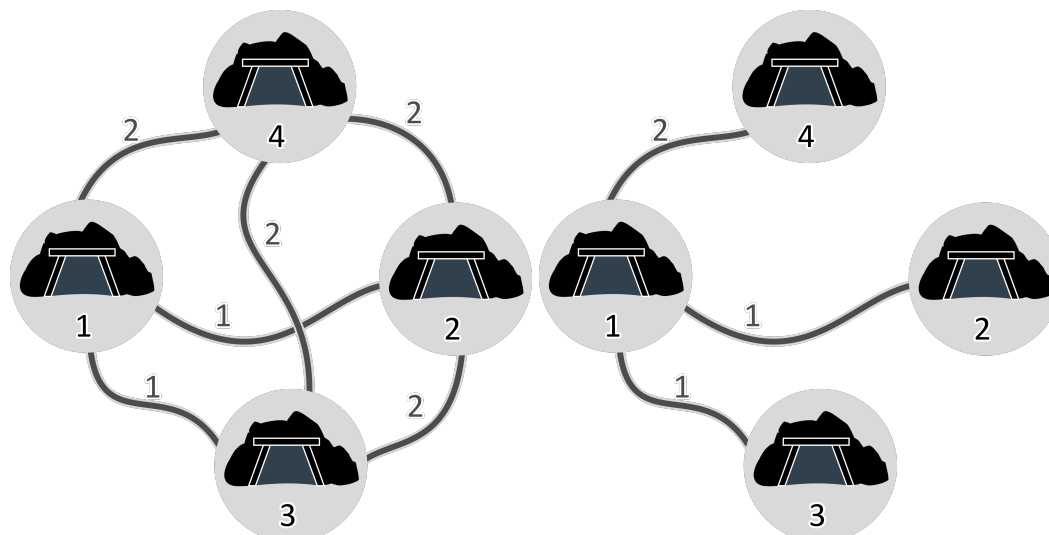


ТРЕНИРОВЪЧНО СЪСТЕЗАНИЕ НА РАЗШИРЕНИЯ НАЦИОНАЛЕН ОТБОР

София, 19 юли 2025 г

Група С

Лявата картинка показва целия граф, а дясната – конструираното в примерната комуникация дърво.



Подзадачи

Подзадача	Точки	Необходимы подзадачи	Други ограничения
0	0	—	Примерна комуникация.
1	14	0	$n \leq 5$
2	11	1	$n \leq 60$
3	16	—	$A = 1, B = 2$, всеки връх има най-много едно инцидентно ребро с тегло A .
4	15	0, 3	$A = 1, B = 2$
5	9	—	Има най-много едно ребро с тегло A .
6	35	0 – 5	Няма допълнителни ограничения. Има частично оценяване.

Точките за дадена подзадача се получават само ако се преминат успешно всички тестове, предвидени за нея и необходимите подзадачи.

Частичното оценяване на последната подзадача е описано в долната таблица. Нека P са точките за текущия тест.

Условие	Точки
$2000 < Q$	0
$1170 < Q \leq 2000$	$\lfloor 5 + \frac{2000}{Q-1000} \rfloor \times P$
$1000 < Q \leq 1170$	$\lfloor 36 - 2^{\frac{Q-1000}{40}} \rfloor \times P$
$Q \leq 1000$	$35 \times P$



ТРЕНИРОВЪЧНО СЪСТЕЗАНИЕ НА РАЗШИРЕНИЯ НАЦИОНАЛЕН ОТБОР

София, 19 юли 2025 г

Група С

Локално тестване

Предоставен Ви е файлът `Lgrader.cpp`, който може да компилирате заедно с вашата програма, за да я тествате. При стартиране програмата ще чете от стандартния вход стойността на n , A и B , след което ще чете матрица на съседството на графа. Може да модифицирате предоставения файл, както искате.