КОНТРОЛНО СЪСТЕЗАНИЕ
НА РАЗШИРЕНИЯ НАЦИОНАЛЕН ОТБОР
Пловдив, 2 – 4 юни 2022 г.,
Група А, 11 – 12 клас

Задача А1. Най-малко общо кратно

Марти разполага с произволно генерирана пермутация p на числата $1, 2, 3, \ldots, n$, която Радко иска да научи. Марти не намира за забавно да я каже директно и ще отговаря само на следния въпрос:

- "Колко е най-малкото общо кратно на числата p_i и p_j?"

За съжаление, Радко е твърде зает и аутсорсва задачата на Вас.

Програмата ви ще бъде тествана на n_{tests} подтеста за всеки тест и резултатът Ви ще бъде изчислен спрямо максималния брой на въпросите, с които откривате поредната пермутация.

Обърнете внимание на ограничението по време.

Детайли по имплементацията

Вашата функция `guessPermutation` има следния прототип:

```
std::vector<int> guessPermutation(int n);
```

Тя ще бъде извикана n_{tests} пъти за всеки тест и ще получи като аргумент размера на пермутацията n. Функцията трябва да върне вектор от n числа – поредната търсена пермутация.

Функция `lcm` на журито има следния прототип:

```
long long lcm(int i, int j);
```

Вашата програма може да я вика колкото пъти иска. Като аргумент ѝ се подават два различни индекса i и j от 1 до n, за които искате да зададете въпрос. Функцията връща най-малкото общо кратно на числата p_i и p_j. Тя работи със сложност $O(log_2(n))$.

Вашата програма трябва да имплементира функцията `guessPermutation`, но не трябва да съдържа функция `main`. Освен това, тя трябва да не чете на стандартния вход или да печата на стандартния изход. Програмата ви също така трябва да включва хедър файла **lcm.h** чрез указание към предпроцесора:

```
#include "lcm.h"
```

Стига да спазва тези условия, програмата ви може да съдържа каквито и да е помощни функции, променливи, константи и прочее.

Ограничения

Всяка пермутация е произволно генерирана.

Подзадачи и оценяване

Частта от точките, които ще получите на дадена подзадача зависи от максималния брой заявки, които правите на подтест, $q_{participant}$, и от константата за подзадачата q_{author}.

Ако $q_{participant} \leq q_{author}$:

$$score = 1$$

Иначе:

$$score = 1 - \sqrt{1 - \frac{q_{author}+1}{q_{participant}+1}}$$

КОНТРОЛНО СЪСТЕЗАНИЕ
НА РАЗШИРЕНИЯ НАЦИОНАЛЕН ОТБОР
Пловдив, 2 – 4 юни 2022 г.,
Група А, 11 – 12 клас

Подзадача	Точки	n	q_{author}	n_{tests}
1	20	10	13	5 000 000
2	20	100	106	500 000
3	20	1 000	1 002	50 000
4	20	10 000	10 001	5 000
5	20	100 000	99 999	500

Локално тестване

В системата ви е предоставен файлът **Lgrader.cpp**, чрез който може да тествате локално програмата си. За целта трябва да добавите `#include "Lgrader.cpp"` към кода си.

На първия ред на стандартния вход се въвеждат числата n и n_{tests}.

Следват n_{tests} теста, за всеки от които се въвеждат по n числа – поредната пермутация.

Ако програмата Ви успешно намери правилната пермутация за всеки тест, накрая ще се изведе максималния брой на заявки, които сте използвали за една пермутация.