

Летняя стажировка  
2021 — мобильная  
разработка

🕒 13 май 2021, 15:46:43

старт: 13 май 2021, 14:03:44

финиш: 13 май 2021, 20:03:44

до финиша: 04:16:56

...

Объявления жюри

Завершить

Задачи    Псылки    Сообщения

С. Алиса и тайный дневник

✖

А. Баг в библиотеке

В. Градиент

С. Алиса и тайный дневник

Д. Игра на торе

Е. Сокращение маршрута

Ф. TAF23

Ограничение времени	1 секунда
Ограничение памяти	64Mb
Ввод	стандартный ввод
Вывод	стандартный вывод

Алиса и Боб живут вместе уже много лет. Внезапно выяснилось, что Боб совершенно не помнит дату, когда они поженились. Данный факт очень возмутил Алису. Чтобы Боб лучше запомнил эту памятную дату, она придумала игру со следующими правилами:

- Алиса в тайном дневнике выписала все-все-все памятные даты, расположив их в хронологическом порядке. В один день произошло ровно одно памятное событие. Алиса ни под каким предлогом не показывает тайный дневник Бобу.
- Оказалось, что дата, когда они поженились, расположилась ровно в середине списка. Причем список имеет нечетную длину. Обо всем этом Алиса сообщила Бобу.
- Каждый день Боб оставляет на столе конверт, где пишет ровно одну дату - предполагаемую дату, когда они поженились. И уходит на работу.
- Алиса забирает конверт, смотрит на дату, открывает свой тайный дневник и считает количество событий, которые произошли до этой даты включительно, а затем пишет Бобу ответную записку, где указывает это число.
- Пункты 3-4 повторяются, пока Боб не определит точную дату памятного события.

Боб очень любит Алису и хочет как можно скорее определить дату, когда они поженились. Кроме того, Алиса установила Бобу предельное количество попыток  $N$ , за которые Боб должен угадать памятную дату. Боб просит вас написать компьютерную программу, которая поможет ему в этом.

Формат ввода

Эта задача немного необычна: в ней вам предстоит реализовать интерактивное взаимодействие с тестирующей системой. Это означает, что вы можете делать запросы и получать ответы в `online`-режиме.

Для чтения ответов тестирующей системы программа должна использовать стандартный ввод.

В первой строке входных данных будет содержаться целое число  $N$  ( $1 \leq N \leq 20$ ) - предельное количество попыток угадать памятную дату.

Известно, что все памятные даты Алисы находятся в промежутке `[01.01.1970...31.12.2020]` и что существует алгоритм, позволяющий угадать дату за требуемое количество попыток. Все даты заданы в формате «dd.mm.yyyy» (без кавычек). В таком же формате их необходимо задавать в запросах к тестирующей системе.

В следующих строках на вход вашей программе будут подаваться строки, содержащие информацию об ответе Алисы.  $i$ -я из этих строк является ответом системы на ваш  $i$ -й запрос.

Ваша программа должна выводить запросы к тестирующей системе и обрабатывать ответы от нее. После вывода каждой строки программа должна выполнить операцию `flush`. Ответ на запрос программа сможет прочесть из стандартного ввода.

Доступные ответы системы:

«?  $K$ » (без кавычек) - ответ тестирующей системы, означающий, что до посланной проверяемой даты произошло  $K$  памятных событий ( $0 \leq K \leq 20000$ ).

«!» (без кавычек) - ответ тестирующей системы, означающий, что превышено количество попыток  $N$  угадать памятную дату.

В шаблоне решения реализовано считывание конфигурации тестирующей системы, посылка запроса. См. результат функции `print(_answer)(date)` и `ask(date)`.

Формат вывода

Для запросов к тестирующей системе программа должна использовать стандартный вывод.

Доступные запросы системы:

«?  $M$ » (без кавычек) - проверить, является ли дата  $M$  ( $01.01.1970 \leq M \leq 31.12.2020$ ) - памятной датой.

«!  $M$ » (без кавычек) - указать, что дата  $M$  ( $01.01.1970 \leq M \leq 31.12.2020$ ) является памятной. После того, как ваша программа вывела данную команду, следует завершить работу.

Приложение должно отвечать тестирующей системе непрерывно одной из описанных команд, пока либо дата не будет угадана, либо пока приложение не получит команду «!» (без кавычек) на завершение от тестирующей системы. Если приложение перестанет отвечать, а дата не угадана, вы получите ошибку `Presentation Error` или `Time Limit Exceeded`.

В шаблоне решения реализовано чтение ответа от тестирующей системы. См. результат функции `ask(date)`.

стандартный ввод	стандартный вывод
20	
? 1	? 01.01.2020
? 1	? 31.12.2020
? 3	? 14.03.2020
? 2	
	! 14.03.2020
!	

Примечания

Обратите внимание, что ввод/вывод в задаче — стандартный (то есть с экрана на экран). При написании решения для интерактивной задачи важно помнить, что если вы что-то вывели, то без специального указания в программе эти данные могут на самом деле попасть во внутренний буфер и не быть выведенными немедленно. После вывода очередного запроса обязательно используйте функции очистки потока, чтобы часть вывода не осталась в каком-нибудь буфере. Например, на Java, Kotlin вызывайте `System.out.flush()`, при программировании на C++ надо использовать `fflush(stdout)` или `cout << flush` (в зависимости от того вы выводите с помощью `scanf/print` или `cout`). В языке Swift, Objective C можно применять `fflush(stdout)`. [Подробнее](#).

Вы можете воспользоваться шаблоном решения: скачайте архив по ссылке «Скачать условие задачи» ниже.

[Скачать условие задачи](#)

Язык

Swift 5.3

Набрать здесь

Отправить файл

1

Отправить

1 осталось 100 попыток

Предыдущая

Следующая

1

Посылок нет

Справка    Обратная связь    Пользовательское соглашение

© 2013–2021 ООО «Яндекс»