

День 1**Общее описание задач**

№	Название задачи	Ограничение на объем используемой памяти	Ограничение на время выполнения программы, секунды	Количество баллов присвоенных задаче
1.	Анаграммы	≤ 1270 Кб	$\leq 0,008$	100
2.	Крот	≤ 37 Мб	$\leq 0,790$	100
3.	Клуб поваров	≤ 1100 Кб	$\leq 0,028$	100

Примечание. При равенстве общего числа очков, лучшим будет считаться решения с меньшим временем выполнения. В случае равенства и времени выполнения, лучшим будет считаться решение, использующее меньший объем памяти.

Анаграммы

Испокон веков поиск анаграмм было любимым развлечением интеллектуальных элит. В литературе операция аннаграммирования состоит в изменении порядка букв заданного текста. Например известный правитель *Antioh Cantemir*, который основал в Яссах Государеву Академию, анаграммировал свое имя как *Hariton Machentin*.

В дидактических целях, под операцией анаграммирования, применяемой к последовательности символов, будем подразумевать изменение порядка символов этой последовательности. Результат выполнения данной операции называется анаграммой. Очевидно что, посредством анаграммирования, из последовательности, состоящей из более чем двух символов, можно получить несколько анаграмм.

Примеры анаграмм:

<i>Исходное слово</i>	<i>Анаграммы исходного слова</i>
avocat	covata octava
lacrima	calmari malaric

Любимое в прошлом развлечение – анаграммирование, используется в настоящее время для шифрования корреспонденции. Однако, шифрование может быть использовано не только для сохранения тайны переписки, но и в неблагоприятных целях. Так, международные службы кибернетической безопасности перехватили в системах электронной почты серию подозрительных сообщений, зашифрованных, возможно, с помощью анаграммирования.

Текст каждого из таких сообщений представляет собой последовательность строчных букв английского алфавита. Обозначим такой текст через T .

Согласно правилам электронной почты, каждое сообщение сопровождается темой – кратким описанием. Тема представляет собой последовательность символов, состоящей из строчных букв английского алфавита. Обозначим такую последовательность через S .

Международные службы кибернетической безопасности считают подозрительными те сообщения, тексты T которых содержат анаграммы их тем S .

Так как количество сообщений пересылаемых системами электронной почты огромно, международные службы кибернетической безопасности нуждаются в компьютерной программе, которая выявляла бы подозрительные сообщения.

Задание. Напишите программу которая определяет, содержит ли последовательность символов T хотя бы одну из анаграмм последовательности символов S .

Входные данные. Стандартный ввод содержит в единственной строке целые числа P , T и N , разделенные пробелом.

Выходные данные. Стандартный вывод должен содержать в единственной строке целое число N – индекс первого из символов с которого в последовательности T начинается анаграмма последовательности S . Если T не содержит ни одной анаграммы

последовательности S , следует вывести “-1”. Напоминаем, что в последовательностях S и T символы нумеруются начиная с индекса 0.

Ограничения. $2 \leq L(S) \leq 10^5$; $2 \leq L(T) \leq 10^6$, где $L(S)$ количество символов в последовательности S , а $L(T)$ количество символов в последовательности T . Ограничения на время выполнения программы и на объем используемой памяти приведены в общем описании предложенных для решения задач. Исходный файл должен иметь имя `anagrame.pas`, `anagrame.c` или `anagrame.cpp`.

Оценивание. Тесты сгруппированы следующим образом:

- 20% баллов: $2 \leq L(S) \leq 50$; $2 \leq L(T) \leq 200$.
- 30% баллов: $50 < L(S) \leq 200$; $200 < L(T) \leq 10^4$.
- 20% баллов: $200 < L(S) \leq 10^4$; $10^4 < L(T) \leq 10^5$.
- 30% баллов: $10^4 < L(S) \leq 10^5$; $10^5 < L(T) \leq 10^6$.

Пример 1.

Ввод

```
urmarireportocal
clopotar
```

Вывод

```
8
```

Пример 2.

```
searaopreaiasisase
opera
```

```
5
```

Пример 3.

```
unudoitreioptzece
patru
```

```
-1
```

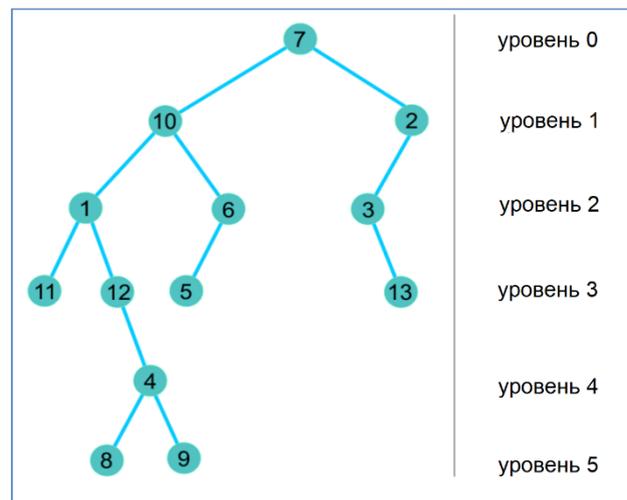
Крот

У Иона большая проблема – у него на огороде завелся крот. Хотя крот и не вредит, холмики которыми он усеял все грядки портят вид огорода. Ион хотел бы прогнать крота, не навредив ему. Для начала Ион решил разведать жилище крота.

В результате исследований, выполненных с помощью автономного микробота, Ион выяснил, что жилище крота имеет древовидную структуру, с множеством норок (смотри рисунок). Ион пронумеровал норки в произвольном порядке, используя последовательные числа $1, 2, 3, \dots, N$.

Крот перемещается между норками по прямым галереям. Входом в жилище крота является ближайшая к поверхности норка (норка находящаяся на уровне 0).

Каждая из находящихся на уровне $i, i = 1, 2, 3$ и т.д. норка обязательно соединена галереей с норкой находящейся на уровне $(i-1)$ и с не более чем двумя норками на уровне $(i+1)$. Галереи проложены наклонно, следовательно возможны следующие случаи:



- Норка, находящаяся на уровне i , не имеет галерей к норкам, находящимся на уровне $(i+1)$;
- Норка, находящаяся на уровне i , соединена галереей с наклоном влево с единственной норкой, находящейся на уровне $(i+1)$;
- Норка, находящаяся на уровне i , соединена галереей с наклоном вправо с единственной норкой, находящейся на уровне $(i+1)$;
- Норка, находящаяся на уровне i , соединена наклонными галереями с двумя норками находящимися на уровне $(i+1)$: с первой – через галерею с наклоном вправо, со второй – через галерею с наклоном влево.

Чтобы прогнать крота, Ион решил установить в его жилище два отпугивающих устройства – микроскопические ультразвуковые генераторы, излучение которых, как надеется Ион, заставит крота «переселиться» в соседний, пустующий огород.

После долгих размышлений, Ион решил что наилучшим вариантом установки генераторов будет их размещение в крайних, в самой левой и в самой правой норках, находящихся на одном и том же уровне жилища крота. Но, так как нумерация норок производилась произвольно, а данные переданные микроботом имеют специфическую структуру, Ион не знает номера крайних норок выбранного уровня K .

Например, если бы Ион решил установить отпугивающие устройства в крайние норки, находящихся на 3-м уровне, их номерами были бы 11 и 13.

Задание. Напишите программу, которая зная выбранный Ионом уровень K и данные собранные микро роботом, определяет номера крайних норок, самой левой и самой правой.

Входные данные. Первая строка стандартного ввода содержит целые числа N и K , разделенные пробелами. Каждая из следующих $N-1$ строк стандартного ввода содержит по два целых числа и одну из букв S, D, разделенные пробелами. Каждая пара чисел описывает пару норок, соединенных галереями: первое число является номером норки находящейся на уровне

i , второе число является номером норки находящейся на уровне $(i+1)$. Буква S указывает на галерею с наклоном влево (норка находящейся на уровне $(i+1)$ расположена левее), а буква D указывает на галерею с наклоном вправо (норка находящейся на уровне $(i+1)$ расположена правее) норки находящейся на уровне i .

Выходные данные. Стандартный вывод содержит в единственной строке два целых числа разделенных пробелом: номер крайней левой норки и номер крайней правой норки заданного уровня K .

Ограничения. $3 \leq N \leq 1000000$; $0 \leq K \leq 500$. Ограничения на время выполнения программы и на объем используемой памяти приведены в общем описании предложенных для решения задач. Имя исходного файла `cartita.pas`, `cartita.c` или `cartita.cpp`.

Пример.*Ввод*

```
13 3
4 8 S
10 6 D
7 2 D
4 9 D
1 12 D
6 5 S
12 4 D
2 3 S
3 13 D
1 11 S
10 1 S
7 10 S
```

Вывод

```
11 13
```

Клуб поваров

Дан является членом знаменитого клуба европейских поваров. В клубе состоят N членов. Клуб очень гордится своим особым рейтингом, названном „Лукоустойчивые повара”. Как следует из этого названия, на первой позиции в рейтинге находится самый стойкий к запаху лука повар, а на последней – самый слабый. „Лукоустойчивость” измеряется длительностью интервала времени, на протяжении которого чистящий лук повар сдерживает свои слезы. Очевидно, в рассматриваемом рейтинге, на первой позиции находится повар который, чистя лук, сдерживает слезы наибольший промежуток времени, а на последней - повар с наименьшим бесслезным промежутком времени.

Каждый год клуб поваров организывает конкурс, на основании результатов которого рейтинг «Лукоустойчивые повара» обновляется. Конкурс состоит из R последовательных раундов.

Во время каждого из раундов i , первые K поваров в текущем рейтинге приглашаются за стол и, по сигналу судьи, они начинают чистить лук. Количество лука неограничено. Как только судья определяет первого из сидящих за стол заплакавшего повара, его переводят на последнюю позицию рейтинга, с номером N , а все повара находящиеся в рейтинге ниже заплакавшего повара, перемещаются на одну позицию вверх. На этом раунд заканчивается.

Во время конкурса ведется протокол, в котором, для каждого раунда i , $i = 1, 2, 3, \dots, R$, заносится начальная позиция A_i участника, который в конце текущего раунда заплакал первым и, соответственно, был перемещен в конце рейтинга. Считается что позиции все участников различаются.

После соревнований, Дан знает свою конечную позицию P в финальном рейтинге „Лукоустойчивые повара”. Однако он забыл свою начальную позицию D , на которой он находился в соответствующем рейтинге до начала соревнования.

Внимательно анализируя протокол соревнований, Дан пытается определить свой начальный рейтинг D .

Задание. Напишите программу, которая, зная финальный рейтинг Дана P и протокол соревнований, определяет его рейтинг D до начала соревнований.

Входные данные. Стандартный ввод содержит в первой строке целые числа N, K, R , разделенные пробелами. Вторая строка стандартного ввода содержит целые числа $A_1, A_2, \dots, A_i, \dots, A_R$, разделенные пробелами. Третья строка стандартного ввода содержит целое число P .

Выходные данные. Стандартный вывод содержит в единственной строке целое число D .

Ограничения. Ограничения на время выполнения программы и на объем используемой памяти приведены в общем описании предложенных для решения задач. Исходный файл иметь имя `clubul.pas`, `clubul.c` или `clubul.cpp`.

Оценивание. Тесты сгруппированы в четыре категории, как указано ниже:

- 30% баллов: $1 \leq K < N \leq 10^3$; $1 \leq R \leq 10^4$.
- 20% баллов: $1 \leq K \leq 10^2$; $K < N \leq 10^6$; $1 \leq R \leq 10^4$.
- 20% баллов: $1 \leq K \leq 10^2$; $K < N \leq 10^9$; $1 \leq R \leq 10^5$.
- 30% баллов: $1 \leq K < N \leq 10^9$; $1 \leq R \leq 10^5$.

Пример.*Ввод*

6	2	3
1	2	1
5		

Вывод

3

Объяснение. Перед началом соревнований Дан занимал 3-ю позицию. После первого раунда, участник с позиции 1 был перемещен на последнюю, с номером 6. Очевидно, остальные участники поднялись в рейтинге на одну позицию, и, соответственно, Дан перешел на вторую позицию. Во втором раунде Дан был перемещен на последнюю позицию, с номером 6. В третьем раунде в соревновании приняли участие повара, которые в начальном рейтинге занимали позиции 2 и 4, а участник со второй позиции был перемещен на последнюю, 6-ю позицию. Таким образом, Дан перешел на 5-ю позицию.