



## Задача 1

### Задание

Необходимо вычислить квадратный корень заданного натурального числа  $n$  с округлением вниз до ближайшего целого, не используя функций  $\text{sqrt}(n)$  и  $n^{0.5}$ . Например, для числа 14 результатом будет число 3, для числа 28 – число 5. Также необходимо привести краткую информацию об используемом алгоритме.

### Входные данные

Произвольное натуральное число  $n$ .

### Выходные данные

Квадратный корень заданного числа, округлённый вниз до ближайшего целого.

## Задача 2

### Задание

Задан массив, элементами которого являются произвольные целые числа. Необходимо определить (максимально оптимальным образом) элемент с индексом  $k$  для массива, состоящего из таких же чисел, но отсортированного по возрастанию. Например, для массива  $[6, 1, 4, 1, 7]$  и  $k=3$  правильным ответом будет число 4 ( $[1, 1, 4, 6, 7]$ ).

### Входные данные

Массив произвольных целых чисел и натуральное число  $k$  ( $1 \leq k \leq \text{длина массива}$ ).

### Выходные данные

$k$ -й элемент массива, состоящего из таких же чисел, как и исходный массив, но отсортированного по возрастанию.



### Задача 3

#### Задание

На вход приложения подается список строк-шифров, заданных в определенном формате и содержащих информацию о запланированных грузоперевозках. Программа должна произвести дешифровку списка входящих строк-шифров и сохранить результирующие данные в CSV файл в виде таблицы, используя следующие имена столбцов, типы выходных значений и алгоритмы расшифровки:

№	Имя столбца таблицы	Тип значения	Информация для дешифровки значения
1	шифр	строка	исходный шифр
2	код водителя	строка	значение присутствует всегда, представляет собой первые 4 символа строки шифра
3	код путевого листа	строка	значение присутствует всегда, имеет следующий формат (по порядку, слева направо): <ul style="list-style-type: none"><li>○ символ «R» или «г»</li><li>○ код опасного груза «d» (опционально)</li><li>○ код хрупкого груза «f» (опционально)</li><li>○ номер путевого листа, который при наличии кода хрупкого и/или опасного груза состоит из 3х цифр, а при отсутствии обоих – из 4х</li><li>○ код водителя (опционален, при наличии является идентичным коду водителя, расшифрованному в столбце №2)</li></ul>
4	опасный	«true» или «false»	см. «код путевого листа» для дешифровки; при наличии кода в шифре следует сохранить в поле таблицы значение «true», при отсутствии - «false»
5	хрупкий	«true» или «false»	см. «код путевого листа» для дешифровки; при наличии кода в шифре следует сохранить в поле таблицы значение «true», при отсутствии - «false»
6	температура	число	является опциональным значением; при наличии состоит из 4 символов и имеет следующий формат: <ul style="list-style-type: none"><li>○ первый символ – знак «+» или «-»</li></ul>



			<ul style="list-style-type: none"><li>○ остальные символы – строка из цифр, задающих значение температуры; может иметь префикс из нулей</li></ul>
7	наименование	строка	значение присутствует всегда; представляет собой последовательность символов из таблицы, закодированную в виде цифр в восьмеричной системе счисления; при этом, ASCII-код каждого символа результирующей строки представлен в заданной последовательности тремя цифрами. Например, "water" <-> "167141164145162"

Также известно, что:

- все блоки информации, если они присутствуют в шифре, всегда расположены по отношению друг к другу в указанном в таблице порядке. Исключением являются поля №4 и №5, которые входят в состав кода путевого листа, однако при этом всё равно сохраняют заданный порядок по отношению друг к другу
- после блоков №2 и №7 может присутствовать до двух (включительно) незначащих алфавитно-цифровых символов (латиница)

Пример:

Исходный шифр:

CAZgRf820167151156145

CSV:

"шифр","код водителя","код путевого листа","опасный","хрупкий","температура","наименование"  
"CAZgRf820167151156145","CAZg","Rf820","false","true","wine"

Входные данные

Список строк-шифров. Например:

RMuiRdf010160141151156164
lims8r3860lims1631411561441
GZQRyr6870GZQR+0041431501451451631455A
qkMfPjrd0561411551551651561511641511571567
EOcTkerf389-0201511431450551431621451411550

Выходные данные

CSV файл, содержащий таблицу с дешифрованными данными.

## Задача 4

### Задание

Заданы  $n$  произвольных точек на плоскости. Необходимо объединить все имеющиеся точки замкнутой ломаной линией таким образом, чтобы она не имела самопересечений.

### Входные данные

Список произвольных координат точек на плоскости.

### Выходные данные

Список координат точек на плоскости, которые при последовательном соединении образуют замкнутую ломаную линию, не имеющую самопересечений.

## Примечания к выполнению заданий

Во время написания программ обратите внимание, что:

- код приложений должен строго соответствовать методологии ООП, а также быть компактным и легко читаемым
- приложения должны корректно реагировать на широкий спектр возможных входных значений, обрабатывать исключительные ситуации
- все задачи должны быть решены наиболее оптимальным образом, с наименьшим использованием ресурсов памяти и процессора
- в исходный код следует добавлять комментарии на английском языке

Задания могут быть выполнены с использованием любого языка программирования.

Выполненное задание (исходный код) присылайте на email [jobs@sysgears.com](mailto:jobs@sysgears.com), в качестве темы письма укажите: "Выполненные задания. [Имя Фамилия]".

Дополнительно к письму необходимо прикрепить резюме.