**1. Условие задачи**

В типизированном файле, хранящем записи вида **С1**, найти (и вывести) **С2**, и переставить её по правилу **С3**

|  |  |  |
| --- | --- | --- |
| **C1** | **C2** | **C3** |
| Вид транспорта (автобус, трамвай и т.д.)  Номер маршрута (например, 10, 10А, А)  Дата утверждения маршрута | *Последнюю* запись с самым давним (по времени) утвержденным маршрутом | Перед *k*-ой записью файла (со сдвигом) |

**2. Уточненное условие задачи**

Даны несколько записей, содержащих информацию о транспорте Вид (перечисляемый тип), строку номера маршрута и целочисленные день, месяц и год утверждения маршрута. Найти среди них последнюю запись с самым давним утверждением маршрута, и поставить перед k-й(целочисленная, вводится через диалог) со сдвигом тех, что лежат между ними.

Запись transp:**TRun**

TTransport = (Trolley, Bus, Mini, Tram); //0 - 3

TRun = record

tr\_name : TTransport; //"Тип" транспорта, выраженный значеникм перечисляемого типа.

num : string[4]; //Номер маршрута

Date : record //Дата утверждения маршрута, выраженная (под)записью

day : Integer; //День, месяц и год соответственно

month : Integer;

year : Integer;

end;

end;

**3.Таблица данных**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Класс | Имя | Описание (смысл) | Тип | Структура |
| Исходные | ch | Символ для управления меню | символьный | простая переменная |
| transp | Входная запись | пользовательский | запись |
| (Процедура SearchChange) | k | Номер записи, перед которой нужно поставить найденную запись | целочисленный | простая переменная |
| Выходные | S1 | Отсортированная строка без своей таблицы символов | символьный | строка |
|  | S2 | Отсортированная строка со своей таблицей символов | символьный | строка |
| Промежуточные | Nom | Номер аномальной ситуации (1 или 2), или 0, если нет аномалий | целочисленный | простая переменная |

**Функциональные тесты**

|  |  |  |  |
| --- | --- | --- | --- |
| № | Исходные данные | Ожидаемый результат | Смысл теста |
| 1 | S= ‘’ | ‘Пустая строка’ | Аномалия 1 Пустая строка |
| 2 | S= ‘iJщюкQ4’ | ‘Некорректные символы’ | Аномалия 2 Есть запрещенные символы |
| 3 | S= ‘01234’ | S= ‘01234’ | Есть только символы *первой части*, и они в требуемом порядке возрастания |
| 4 | S= ‘9104’ | S= ‘0149’ | Есть только символы *первой части* НЕ в требуемом порядке |
| 5 | S= ‘5430’ | S= ‘0345’ | Есть только символы *первой части* в порядке противоположном требуемому |
| 6 | S= ‘ЯФЕВРАЛЁК’ | S= ‘ЯФЕВРАЛЁК’ | Есть только символы *средней* части – **сохраняется исходный порядок (как введено)** |
| 7 | S= ‘zsrqjia’ | S= ‘zsrqjia’ | Есть только символы *последней* части и в требуемом порядке |
| 8 | S= ‘iqsqjr’ | S= ‘srqqji’ | Есть только символы *последней* части НЕ в требуемом порядке |
| 9 | S= ‘aijqqrsz’ | S= ‘zsrqqjia’ | Есть только символы *последней* части в порядке противоположном требуемому |
| 10 | S= ‘91i0j4nq’ | S= ‘0149qnji’ | *Нет «средней части»* – заглавных букв кириллицы |
| 11 | S=’ЯЁЖbadcИК’ | S=’ЯЁЖИКdcba’ | *Нет «первой части»* – цифр |
| 12 | S=’БА3БА9Я2Г0А’ | S=’0239БАБАЯГА’ | *Нет «последней части»* – строчных букв латиницы |
| 13 | S= ‘0149АМЁБАqnji’ | S= ‘0149АМЁБАqnji’ | *Есть все части* и в правильном порядке |
| 14 | S= ‘ijnqАМЁБА9410’ | S= ‘0149АМЁБАqnji’ | *Есть все части* в противоположном порядке |
| 15 | S= ‘А91iМ0Ёj4БnqА’ | S= ‘0149АМЁБАqnji’ | *Есть все части* не в правильном порядке – Пример 1 |