

## Вариант №343

Написать программу для работы с таблицей, использующей два пространства ключей, по запросам оператора.

Каждый элемент таблицы имеет следующую структуру:

```
struct Item {
    // указатель на информацию
    InfoType *info;
    // версия элемента
    RelType release;
    // указатель на следующий элемент с данным составным ключом
    Item *next;

    // опциональные поля, для оптимизации выполнения операций,
    // состав и наличие которых должны быть обоснованы:

    // ключ элемента из 1-го пространства ключей
    KeyType1 key1;
    // ключ элемента из 2-го пространства ключей
    KeyType2 key2;
    // связь с элементом 1-го пространства ключей по индексу
    IndexType1 ind1;
    // связь с элементом 2-го пространства ключей по индексу
    IndexType2 ind2;
    // связь с элементом 2-го пространства ключей по указателю
    PointerType1 *p1;
    // связь с элементом 2-го пространства ключей по указателю
    PointerType2 *p2;
};
```

В таблице могут присутствовать элементы с одинаковым составными ключами (`key1`, `key2`) и разными номерами версий (номер версии элемента формируется как порядковый номер элемента в последовательности элементов с одинаковыми ключами, определяемый при включении элемента в таблицу).

Первое пространство ключей организовано как просматриваемая таблица, организованная списком; каждый элемент таблицы имеет следующую структуру:

```
struct KeySpacel {
    // ненулевой ключ элемента
    KeyType1 key;
    // ключ родительского элемента, может быть нулевым
    KeyType1 par;
    // указатель на информацию
    Item *info;
    // указатель на следующий элемент
    KeySpacel *next;
};
```

В пространстве не может быть двух элементов с одинаковыми ключами, а значение ключа родительского элемента для элемента должно совпадать с каким-либо значением ключа элемента, существующего в таблице, либо быть нулевым.

В данном пространстве ключей предусмотрены следующие особые операции:

- удаление из таблицы элемента, заданного своим ключом; при этом также должны быть удалены (рекурсивно) все элементы, ссылающиеся на удаляемый;
- поиск в таблице всех элементов, заданных значением ключа родительского элемента; результатом поиска должна быть новая таблица, содержащая найденные элементы.

Второе пространство ключей организовано как перемешанная таблица, использующая перемешивание сцеплением. Перемешанная таблица представлена массивом указателей на элементы таблицы, имеющие следующую структуру:

```
struct KeySpace2 {  
    // ключ элемента  
    KeyType2 key;  
    // указатель на информацию  
    InfoType *info;  
    // указатель на следующий элемент  
    KeySpace2 *next;  
};
```

Максимальный размер пространства ключей ограничен величиной `msize2`, значение которой определяется при инициализации таблицы.

В пространстве не может быть двух элементов с одинаковыми ключами.

В данном пространстве ключей особые операции не предусмотрены.