**Практическая работа №3: Числа, строки, списки, кортежи, множества, диапазоны, словари**

Выполните следующие задания:

1. Напишите программу для удаления заданного числа из набора.

Output:

Enter the number you want to remove:12

After Removing: {1, 2, 3, 4, 5, 6, 24}

2. Напишите программу для добавления нескольких элементов к множеству.

Output:

{1, 2, 4, 'Apple', 'John', 'CS', 'Mango', 'Grapes'}

3. Напишите программу для нахождения объединения двух множеств.

Output:

{96, 65, 2, 'Joseph', 1, 'Peter', 59}

4. Есть множество, содержащее название стран. Необходимо предоставить пользователю возможности:

* Добавление стран;
* Удаления стран;
* Поиска стран по введенным символам;
* Проверки существует ли страна внутри множества.

**Словари**

Выполните следующие задания:

1. Напишите функцию to\_dict(lst), которая принимает аргумент в виде списка и возвращает словарь, в котором каждый элемент списка является и ключом, и значением. Предполагается, что элементы списка будут соответствовать правилам задания ключей в словарях.

Output:

{1: 1, 2: 2, 3: 3, 4: 4}

{'grey': 'grey', (2, 17): (2, 17), 3.11: 3.11, -4: -4}

Дана строка в виде случайной последовательности чисел от 0 до 9.

2. Требуется создать словарь, который в качестве ключей будет принимать данные числа (т. е. ключи будут типом int), а в качестве значений – количество этих чисел в имеющейся последовательности. Для построения словаря создайте функцию count\_it(sequence), принимающую строку из цифр. Функция должна возвратить словарь из 3-х самых часто встречаемых чисел.

Output:

{1: 10, 2: 3}

{3: 8, 1: 7, 5: 7}

{7: 6, 4: 6, 6: 3}

3. Есть некоторый словарь, который хранит названия музыкальных групп(исполнителей) и альбомов. Название группы используется в качестве ключа, название альбомов в качестве значения. Необходимо реализовать: добавление данных, удаление данных, поиск данных, редактирование данных, сохранение и загрузку данных (используя упаковку и распаковку).

4. Имеется ряд словарей с пересекающимися ключами (значения - положительные числа). Напишите 2 функции, которые делают с массивом словарей следующие операции:

1-ая функция max\_dct(\*dicts) формирует новый словарь по правилу:

Если в исходных словарях есть повторяющиеся ключи, выбираем среди их значений максимальное и присваиваем этому ключу (например, в словаре\_1 есть ключ “а” со значением 5, и в словаре\_2 есть ключ “а”, но со значением 9. Выбираем максимальное значение, т. е. 9, и присваиваем ключу “а” в уже новом словаре).

Если ключ не повторяется, то он просто переносится со своим значением в новый словарь (например, ключ “с” встретился только у одного словаря, а у других его нет. Следовательно, переносим в новый словарь этот ключ вместе с его значением). Сформированный словарь возвращаем.

2-ая функция sum\_dct(\*dicts) суммирует значения повторяющихся ключей. Значения остальных ключей остаются исходными. (Проводятся операции по аналогу первой функции, но берутся не максимумы, а суммы значений одноименных ключей). Функция возвращает сформированный словарь.

Output:

{1: 12, 2: 33, 3: 10, 4: 10, 5: 2, 6: 90, 7: 112}

{1: 12, 2: 36, 3: 14, 4: 83, 5: 33, 6: 98, 9: 9, 10: 556, 7: 25, 8: 71}

{1: 12, 2: 33, 3: 10, 4: 60, 5: 31, 6: 90, 7: 112, 8: 71, 9: 9, 10: 556}

{1: 24, 2: 36, 3: 21, 4: 84, 5: 35, 6: 98, 7: 137, 8: 71, 9: 9, 10: 556}