

A1 В данной реализации есть проблема из-за присваивания ячейке значения **ERASED** — это ухудшает эффективность при пробировании.

Если несколько подряд идущих элементов удалить, то при вставке, поиске и удалении эти ячейки всё ещё будут учитываться, что усложняет временные затраты, а так же затрат по памяти, потому что эти ячейки мы больше не сможем использовать.

Пример: хеш-таблица размера 8 и $\text{hash}(x) = x \% 8$:

0	1	2	3	4	5	6	7
-	-	-	-	-	-	-	-

1) Вставим элементы 0, 8, 16, 24, 32, 40.
У всех хеш равен 0.

0	1	2	3	4	5	6	7
0	8	16	24	32	40	-	-

2) Теперь удалим элемент 8. Тогда ячейка 1 станет **ERASED**. Отметим её **X**.

0	1	2	3	4	5	6	7
0	X	16	24	32	40	-	-

3) И теперь, если мы хотим вставить элемент 80, то мы сделаем это только на ячейке 6, хотя хотелось бы сделать это на ячейке 2.

0	1	2	3	4	5	6	7
0	X	16	24	32	40	80	-

хочу сюда (2)

Улучшение: 1) можно проверять ячейки не только на **NULL**, но и на **ERASED**, если **ERASED**, то вставить элемент можно.

2) использовать при удалении сдвиг элементов в свободные ячейки

3) добавить перехеширование или проверку, что все ячейки пройдены.