

Используя динамические структуры, реализовать следующие задания. Вспомогательные функции должны быть написаны самостоятельно (Встроенную библиотеку использовать нельзя). Сначала создается динамическая структура, затем выполняется задание.

1. Создать стек, содержащий слова. В новый стек записать все слова, которые не заканчиваются на заданную букву. Порядок вывода элементов результирующего стека на экран должен совпадать с порядком ввода элементов. (Например, если введены `aa sab aaa sss qqq saa fff` и заданна буква `a`, то результат должен быть `sab sss qqq fff`).
2. Создать стек, содержащий целые числа. В новый стек сначала вывести четные, потом нечетные числа. Должен быть именно результирующий стек, а не два `cout`. Порядок вывода элементов результирующего стека на экран должен совпадать с порядком ввода элементов. (Например, если введены `1 1 2 3 4 5 1`, то результат должен быть `2 4 1 1 3 5 1`).
3. Создать стек, содержащий целые числа. Вставить нулевой элемент после последнего четного элемента. Порядок вывода элементов результирующего стека на экран должен совпадать с порядком ввода элементов. (То есть, если введены `1 1 2 3 4 5 1`, то результат должен быть `1 1 2 3 4 0 5 1`).
4. Создать стек, содержащий целые числа. Удалить первый кратный X элемент. Порядок вывода элементов результирующего стека на экран должен совпадать с порядком ввода элементов. (Например, если введены `1 1 2 3 4 5 1` и $X = 2$, то результат должен быть `1 1 3 4 5 1`).
5. Создать стек, содержащий латинские буквы. В новый стек вывести сначала гласные, потом согласные буквы. Должен быть именно результирующий стек, а не два `cout`. Порядок вывода элементов результирующего стека на экран должен совпадать с порядком ввода элементов. (Например, если введены буквы `a v h i o f s`, то результат будет `a i o v h f s`).
6. Создать стек, содержащий числа. Удалить из стека все максимальные элементы. Порядок вывода элементов результирующего стека на экран должен совпадать с порядком ввода элементов. (Например, если введены `5 1 1 5 3 4 5 1`, то результат должен быть `1 1 3 4 1`).
7. Создать стек, содержащий числа. Перед каждым минимальным элементом вставить новый элемент. Порядок вывода элементов результирующего стека на экран должен совпадать с порядком ввода элементов. (Например, если введены `1 1 2 3 4 5 1`, то результат должен быть `0 1 0 1 2 3 4 5 0 1`).
8. Создать стек, содержащий целые числа. Удалить из стека все простые числа (1 — не простое число). Порядок вывода элементов результирующего стека на экран должен совпадать с порядком ввода элементов. (Например, если введены `10 11 2 9 3 4 5 6 1`, то результат должен быть `10 9 4 6 1`).

9. Создать стек, содержащий слова. Удалить из стека все слова, содержащие буквы, находящиеся в самом длинном слове стека. Порядок вывода элементов результирующего стека на экран должен совпадать с порядком ввода элементов. (Например, если введены `dre asdf bbb ytr gfd aaa sre vvv`, то результат должен быть `bbb ytr vvv`).
10. Создать стек, содержащий целые числа. Удалить из стека все элементы с максимальной суммой цифр. Порядок вывода элементов результирующего стека на экран должен совпадать с порядком ввода элементов. (Например, если введены `100 12 28 314 2 460 37 54 442 1111`, то результат должен быть `100 12 314 2 54 1111`).
11. Создать очередь, содержащую числа. В новой очереди сначала вывести отрицательные, потом положительные числа. (Например, если введены `1 -5 7 9 -8 -6 2 3 1`, то результат должен быть `-5 -8 -6 1 7 9 2 3 1`).
12. Создать очередь, содержащую слова. Удалить из очереди элементы, совпадающие с последним словом. (Например, если введены `vvv dre bbb vvv ytr vvv aaa vvv`, то результат должен быть `dre bbb ytr aaa`).
13. Создать очередь, содержащую целые числа. После всех чисел, кратных X , вставить новый элемент. (Например, если введены `1 1 2 3 2 4 3 5 4 1` и $X = 2$, то результат должен быть `1 1 2 0 3 2 0 4 0 3 5 4 0 1`).
14. Создать очередь, содержащую любые символы. Удалить из очереди все элементы, не являющиеся буквой или цифрой. (Например, если введены `a 2 - e * ^ f + & 0)`, то результат должен быть `a 2 e f 0`).
15. Создать очередь, содержащую целые числа. После каждого элемента, равного либо первому, либо последнему элементу очереди, вставить новый элемент. (Например, если введены `5 1 2 1 5 3 4 5 1`, то результат должен быть `5 0 1 0 2 1 0 5 0 3 4 5 0 1 0`).
16. Создать очередь, содержащую целые числа. Перед каждым максимальным элементом вставить минимальный элемент. (Например, если введены `5 1 2 1 5 3 4 5 1`, то результат должен быть `1 5 1 2 1 1 5 3 4 1 5 1`).
17. Создать очередь, содержащую целые числа. Удалить из очереди все повторяющиеся числа, оставив только их первые вхождения. (Например, если введены `5 1 2 1 5 3 4 5 1`, то результат должен быть `5 1 2 3 4`).
18. Создать очередь, содержащую целые числа. Исключить все повторяющиеся, идущие подряд элементы, оставив только один из них. (Например, если введены `1 1 2 2 3 3 3 4 5 5`, то результат должен быть `1 2 3 4 5`).
19. Создать очередь, содержащую целые числа. Выполнить циклический сдвиг очереди, так чтобы в начале очереди находился первый максимальный элемент. (Например, если введены `1 2 1 5 3 4 5 1`, то результат должен быть `5 3 4 5 1 1 2 1`).

20. Создать очередь, содержащую целые числа. Удалить все элементы, содержащие максимальное количество делителей. (Например, если введены 10 3 14 15 18 11 16 12, то результат должен быть 10 3 14 15 11 12).