

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 заданная буква `a`, то результат должен быть `sab sss qqq fff`.

2. Создать стек, содержащий целые числа. В новый стек сначала вывести нечетные, потом четные 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 `1 1 3 5 1 2 4`.

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.

Например, 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 сначала вывести 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 `a v h i o f s`, то результат должен быть `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 числа. Удалить из стека все повторяющиеся элементы, оставив только их первые вхождения. Порядок вывода элементов результирующего стека на экран должен совпадать с порядком ввода элементов. Например, если введены 1 1 2 3 2 4 3 5 4 1, то результат должен быть 1 2 3 4 5.
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 все слова, совпадающие с последним словом.

Т. е., если введены `aaa bbb ytr gfd aaa aaa vvv aaa`, то результат должен быть `bbb ytr gfd vvv`.

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 символы, не являющиеся буквами или цифрами.

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 числа, 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 очереди вставить минимальный элемент очереди.

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 слова. Удалить из очереди все повторяющиеся слова, 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 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`.

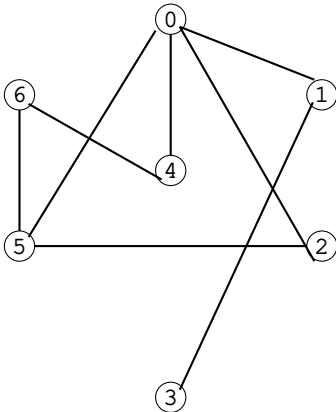
20. Создать очередь, содержащую целые числа. Удалить все элементы, содержащие максимальное количество делителей.

Например, если введены 10 3 14 15 18 11 16 12, то результат должен быть 10 3 14 15 11 16.

21. Дана строка, содержащая символы и круглые скобки. Используя стек, проверить баланс скобок в тексте (открывающая скобка должна идти раньше закрывающей и их количество должно совпадать). Например, $(5 + ((3 - 2) * 4))$ — баланс соблюден; $((a + b * (b - 5))$ — число открывающих скобок больше, чем закрывающих; $(s -) * 5 + (f - 5)$ — закрывающая скобка встречается раньше открывающей.

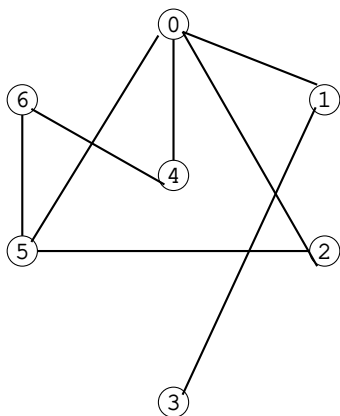
22. Дан граф, представленный матрицей смежности (матрица размера $n \times n$, где n — число вершин графа. $a[i][j] = 1$, если вершины i и j смежны (соединены ребром), $a[i][j] = 0$ в противном случае.) Реализовать нерекурсивный обход графа в глубину, используя стек.

Например, для графа, изображенного на рисунке, при начале из точки 0 результат будет: 0 1 3 2 5 6 4.



23. Дан граф, представленный матрицей смежности (матрица размера $n \times n$, где n — число вершин графа. $a[i][j] = 1$, если вершины i и j смежны (соединены ребром), $a[i][j] = 0$ в противном случае.) Реализовать обход графа в ширину, используя очередь.

Например, для графа, изображенного на рисунке, при начале из точки 0 результат будет: 0 1 2 4 5 3 6.



24. Дано математическое выражение в инфиксной нотации, то есть знаки операций расположены между операндами. Используя стек, переписать его в постфиксную нотацию (операнды расположены перед знаками операций). Например, выражение $(4 + 5) * 8 * (4 - 2) + 2 + 5$ будет представлено в виде $45 + 8 * 42 - * 2 + 5 +$.
25. Дано математическое выражение в постфиксной нотации, то есть операнды расположены перед знаками операций. Используя стек, подсчитать значение заданного выражения. Например, для выражения $45 + 8 * 42 - * 2 + 5 +$ результат будет 327.