

**Тема:** Std, STL

**Вариант:** 4.1.2

**Задача:** Разработать класс для работы с детерминированными и недетерминированными конечными автоматам (ДКА и НКА).

НКА рассматриваются без  $\epsilon$ -переходов.

Необходимо реализовать:

1. Считывание из файла ДКА или НКА: количество состояний, начальное состояние и множество конечных состояний, а также значения функции переходов.
2. Функцию, проверяющую распознается ли заданная строка этим конечным автоматом.
3. Детерминацию — получение по имеющемуся НКА нового ДКА, распознающего то же множество последовательностей

**Замечания по реализации:**

1. Для реализации рекомендуется использовать классы из стандартной библиотеки (`std::vector`, `std::unordered_map`, `std::string`, ...)
2. В коде должно быть реализовано корректное управление динамической памятью: не должно быть утечек памяти, некорректных указателей и т. д

### **Входные данные:**

Во входном файле задан автомат. Он может быть как детерминированным, так и недетерминированным.

В первой строке входного файла задано значение  $N$  – количество состояний автомата.

В следующей строке задано  $0 \leq k < N$  – номер начального состояния.

В следующей строке задано  $0 \leq f < N$  – количество конечных состояний.

В следующих  $f$  строках заданы номера конечных состояний автомата.

В следующей строке задано  $0 \leq p$  – количество функций переходов.

В следующих  $p$  строках заданы функции переходов в формате *from to value*, где  $0 \leq from < N$ ;  $0 \leq to < N$  – номера состояний автомата, а *value* – символ, соответствующий значению функции перехода.

В следующей строке задано значение  $T$  – количество строк, распознаваемость конечным автоматом которых необходимо проверить. Наконец, следуют  $T$  строк, которые необходимо проверить заданным выше автоматом.

### **Выходные данные:**

В выходной файл записать  $T$  строк, в каждой из которых записано "YES", если автомат распознал соответствующую строку и "NO" в противном случае.

**Пример входных и выходных данных:**

input.txt	output.txt
3 0 1 0 6 0 1 a 1 0 b 0 2 b 1 2 a 2 2 a 2 2 b 3 ab aaa abababab	YES NO YES
4 0 2 1 2 12 0 1 a 0 1 b 0 1 c 1 1 a 1 2 b 1 3 c 2 2 a 2 2 b 2 2 c 3 3 a 3 3 b 3 3 c 3 baaaaaaabcababac cbacccc bcabcabcabcabc	YES YES NO

### **Дополнительные задания:**

Реализовать создание ДКА по регулярному выражению, заданному строкой