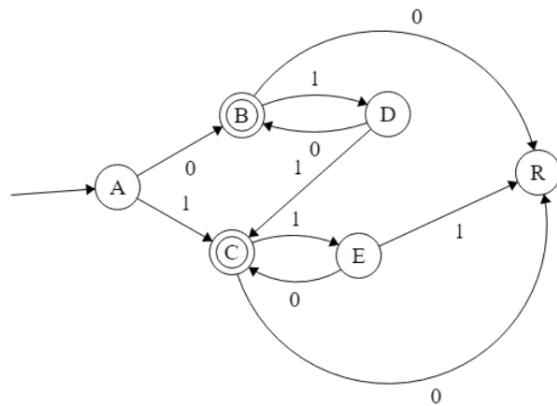


Формальные языки
Домашнее задание 3
Виноградов Александр

Задание №1

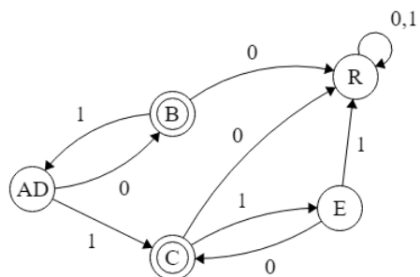
1 шаг алгоритма - если КА неполный, то добавить сток. Добавим стоковое состояние **R**.



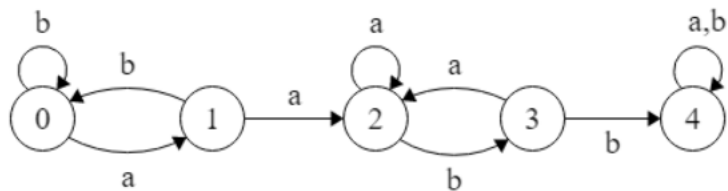
Получаем следующую табличку, откуда видно, что можно объединить состояния **A** и **D**.

| | 0 | 1 | A | B | C | D | E | R |
|---|---|---|---|---|---|---|---|---|
| A | - | - | | | | | | |
| B | A | - | ✓ | | | | | |
| C | E | A | ✓ | ✓ | | | | |
| D | - | B | | ✓ | ✓ | | | |
| E | - | C | ✓ | ✓ | ✓ | ✓ | | |
| R | B | E | ✓ | ✓ | ✓ | ✓ | ✓ | |

В итоге получаем следующий КА:

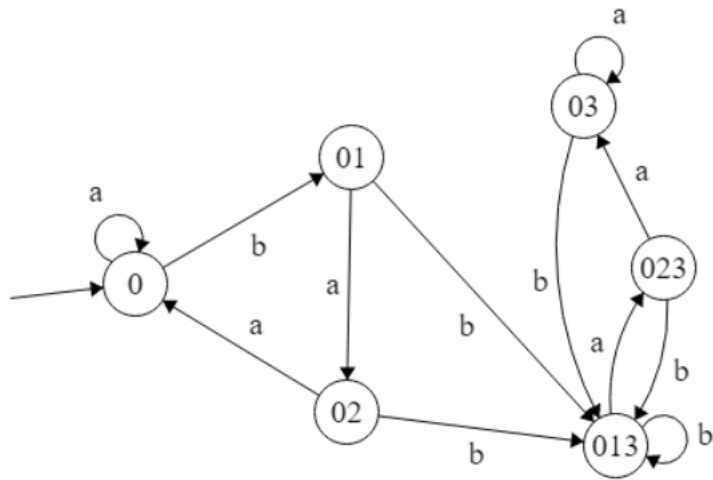


Задание №2



Задание №3

Воспользуемся стандартным алгоритмом для преобразования НКА \rightarrow ДКА. Но перед этим заметим, что КА, где вместо пути $1 \rightarrow 2 \rightarrow 4$ путь $1 \rightarrow 4$ по b - эквивалентен исходному.



Задание №4

Второе регулярное выражение задаёт любую строчку, оканчивающуюся на нечётное число a (Почему? Если в строке нет b , то получим $a(aa)^*$, если есть - набираем произвольную строку, пока не встретим последнюю b , далее дописываем нечётное число a).

Чтобы доказать требуемое - нужно показать, что 1 регулярка описывает все такие строки и не задаёт ничего лишнего.

1. Она не задаёт ничего лишнего, так как после подстроки $b \dots b$ всегда идёт a . А дальше мы умеем дописывать только aa , либо $b^k a$, либо $ab^k a$.
2. Почему можно получить любую строку с нечётным числом a на конце? Достаточно показать, что $((a | b) b^* a)^*$ задаёт любую строку с нечётным числом a на конце. Это действительно так, произвольная строка имеет вид $a^{k_1} b^{k_2} a^{k_3} \dots$. Покажем, что a^k можно набрать, давайте набирать по aa , если k чётное, то набрали, если нечётное, то на последнем шаге в $(a | b)$ выберем a , а после гарантированно идёт b .