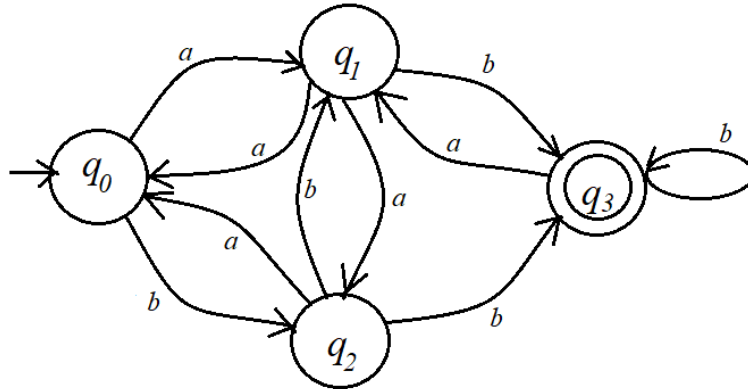
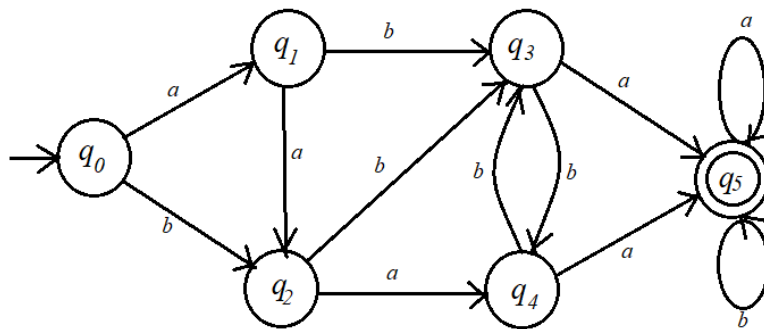


## Вариант 1

1. Построить конечный автомат по заданному регулярному выражению:  $(ab|ba)^*ab^*$ .
2. Построить ДКА, эквивалентный заданному НКА



3. Минимизировать заданный конечный автомат:



4. Дана грамматика ( $E$  – стартовый символ):

$E \rightarrow E \& T \mid T$

$T \rightarrow T \vee F \mid F$

$F \rightarrow \neg F \mid (E) \mid a$

Построить вывод слова  $\neg a \& \neg(a \vee a) \vee a$ .

5. Дана грамматика ( $P$  – стартовый символ):

$P \rightarrow S$ .

$E \rightarrow E+E \mid E<E \mid \text{id} \mid \text{num}$

$S \rightarrow \text{id} := E$

$S \rightarrow \text{if } E \text{ then } S \text{ else } S$

$S \rightarrow \text{while } E \text{ do } S$

$S \rightarrow \text{begin } L \text{ end}$

$L \rightarrow S$

$L \rightarrow S; L$

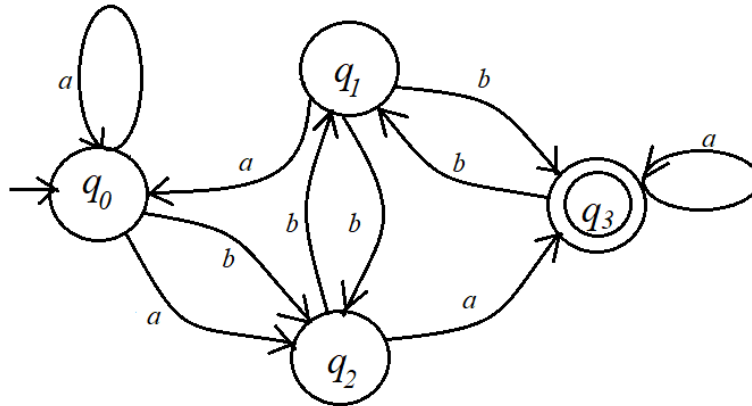
Лексический анализатор превращает все идентификаторы, отличные от ключевых слов языка, в символ `id`, а все целые числа – в `num`. Переводы строк, пробелы и табуляции

игнорируются. Постройте дерево разбора указанного текста:

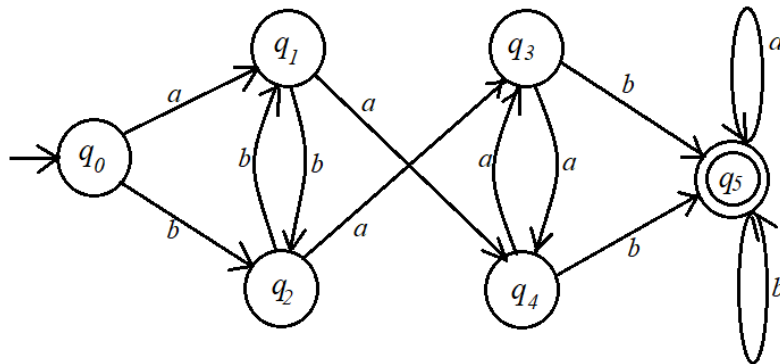
```
begin
  if a < b
    then res := a + 1
    else begin
      res := 0;
      while res < b do
        res := res + 2;
        res := res + b
      end
    end
  end.
```

## Вариант 2

1. Построить конечный автомат по заданному регулярному выражению:  $(ab(ba)^*)^*$ .
2. Построить ДКА, эквивалентный заданному НКА



3. Минимизировать заданный конечный автомат:



4. Дана грамматика (E – стартовый символ):

$E \rightarrow E \& T \mid T$

$T \rightarrow T \vee F \mid F$

$F \rightarrow \neg F \mid (E) \mid a$

Построить вывод слова  $\neg(a \& (a \vee a)) \vee a$ .

5. Дана грамматика (P – стартовый символ):

$P \rightarrow S$ .

$E \rightarrow E+E \mid E<E \mid \text{id} \mid \text{num}$

$S \rightarrow \text{id} := E$

$S \rightarrow \text{if } E \text{ then } S \text{ else } S$

$S \rightarrow \text{while } E \text{ do } S$

$S \rightarrow \text{begin } L \text{ end}$

$L \rightarrow S$

$L \rightarrow S; L$

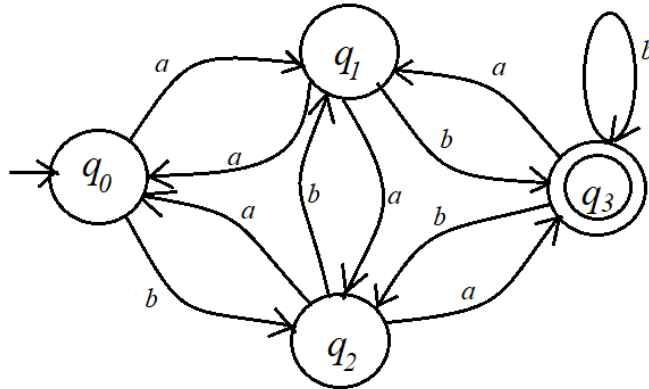
Лексический анализатор превращает все идентификаторы, отличные от ключевых слов языка, в символ `id`, а все целые числа – в `num`. Переводы строк, пробелы и табуляции

игнорируются. Постройте дерево разбора указанного текста:

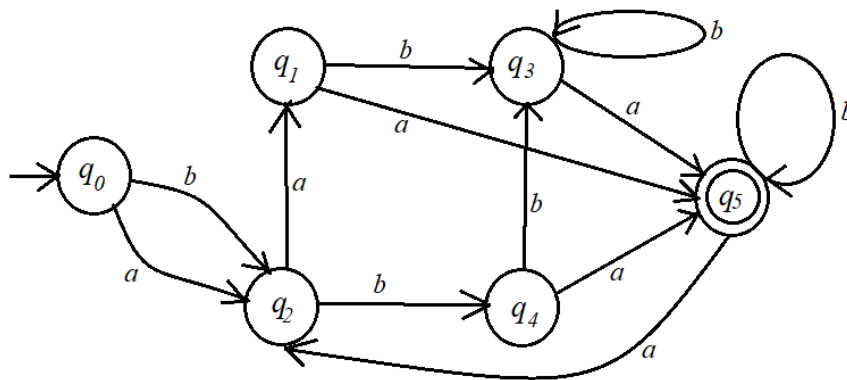
```
begin
  res := 0;
  while res < b do
    res := res + 2;
  if a < b
    then res := res + 1
    else begin
      a := res + b;
      b := a;
    end
  end
end.
```

## Вариант 3

1. Построить конечный автомат по заданному регулярному выражению:  $(ab|a^*)^*$ .
2. Построить ДКА, эквивалентный заданному НКА



3. Минимизировать заданный конечный автомат:



4. Дана грамматика ( $E$  – стартовый символ):

$E \rightarrow E \& T \mid T$

$T \rightarrow T \vee F \mid F$

$F \rightarrow \neg F \mid (E) \mid a$

Построить вывод слова  $\neg a \vee \neg(a \vee a) \vee a$ .

5. Дана грамматика ( $P$  – стартовый символ):

$P \rightarrow S$ .

$E \rightarrow E+E \mid E<E \mid \text{id} \mid \text{num}$

$S \rightarrow \text{id} := E$

$S \rightarrow \text{if } E \text{ then } S \text{ else } S$

$S \rightarrow \text{while } E \text{ do } S$

$S \rightarrow \text{begin } L \text{ end}$

$L \rightarrow S$

$L \rightarrow S; L$

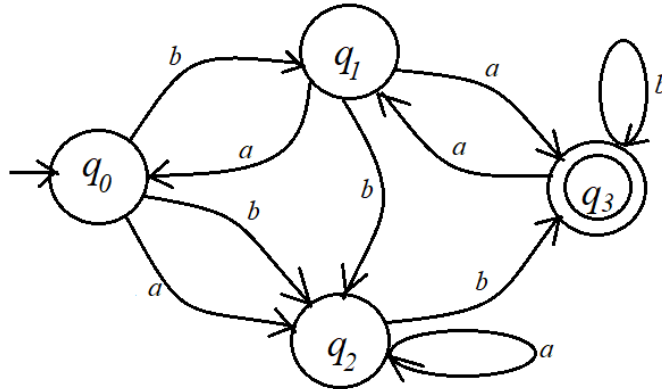
Лексический анализатор превращает все идентификаторы, отличные от ключевых слов языка, в символ `id`, а все целые числа – в `num`. Переводы строк, пробелы и табуляции

игнорируются. Постройте дерево разбора указанного текста:

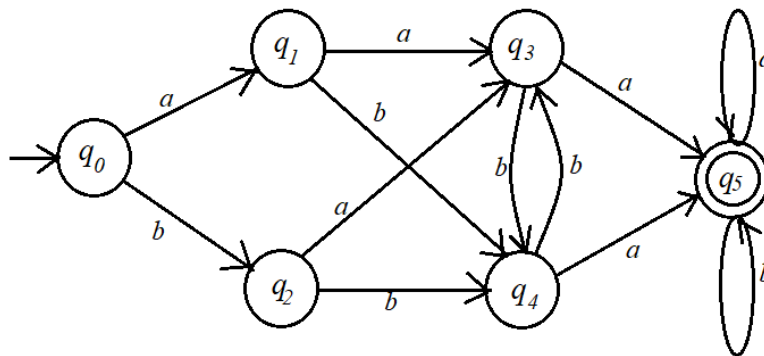
```
begin
  if a < b
    then begin
      res := 0;
      while res < b do
        res := res + 2;
        res := res + b
      end
    else res := a + 1
  end.
```

## Вариант 4

1. Построить конечный автомат по заданному регулярному выражению:  $a^*|b^*|(ab|ba)^*$ .
2. Построить ДКА, эквивалентный заданному НКА



3. Минимизировать заданный конечный автомат:



4. Дана грамматика (E – стартовый символ):

$E \rightarrow E \& T \mid T$

$T \rightarrow T \vee F \mid F$

$F \rightarrow \neg F \mid (E) \mid a$

Построить вывод слова  $\neg(a \vee \neg(a\&a)) \vee a$ .

5. Дана грамматика (P – стартовый символ):

$P \rightarrow S$ .

$E \rightarrow E+E \mid E<E \mid \text{id} \mid \text{num}$

$S \rightarrow \text{id} := E$

$S \rightarrow \text{if } E \text{ then } S \text{ else } S$

$S \rightarrow \text{while } E \text{ do } S$

$S \rightarrow \text{begin } L \text{ end}$

$L \rightarrow S$

$L \rightarrow S; L$

Лексический анализатор превращает все идентификаторы, отличные от ключевых слов языка, в символ `id`, а все целые числа – в `num`. Переводы строк, пробелы и табуляции

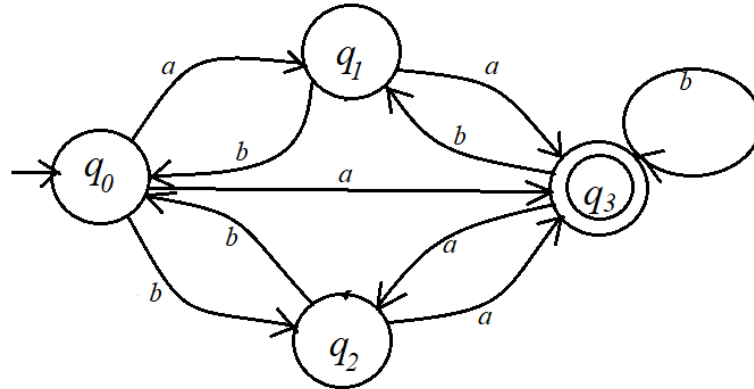
игнорируются. Постройте дерево разбора указанного текста:

```
begin
  res := 0;
  while res < b do
    res := res + 2;
  if a < b
    then res := res + 1
    else begin
      a := res + b;
      b := a;
    end
  end
end.
```

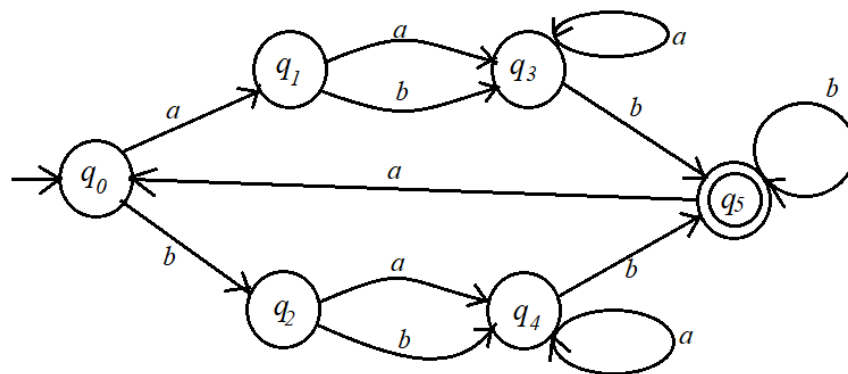


## Вариант 5

1. Построить конечный автомат по заданному регулярному выражению:  $(aa|bbb)^*$ .
2. Построить ДКА, эквивалентный заданному НКА



3. Минимизировать заданный конечный автомат:



4. Дана грамматика ( $E$  – стартовый символ):

$E \rightarrow E \& T \mid T$

$T \rightarrow T \vee F \mid F$

$F \rightarrow \neg F \mid (E) \mid a$

Построить вывод слова  $\neg(\neg(a\&a) \vee a) \&a$ .

5. Дана грамматика ( $P$  – стартовый символ):

$P \rightarrow S$ .

$E \rightarrow E+E \mid E<E \mid \text{id} \mid \text{num}$

$S \rightarrow \text{id} := E$

$S \rightarrow \text{if } E \text{ then } S \text{ else } S$

$S \rightarrow \text{while } E \text{ do } S$

$S \rightarrow \text{begin } L \text{ end}$

$L \rightarrow S$

$L \rightarrow S; L$

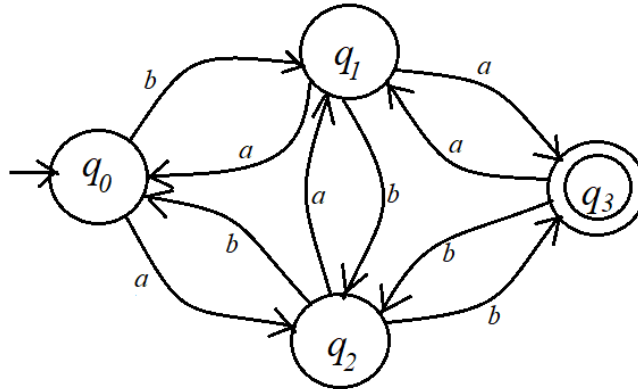
Лексический анализатор превращает все идентификаторы, отличные от ключевых слов языка, в символ `id`, а все целые числа – в `num`. Переводы строк, пробелы и табуляции

игнорируются. Постройте дерево разбора указанного текста:

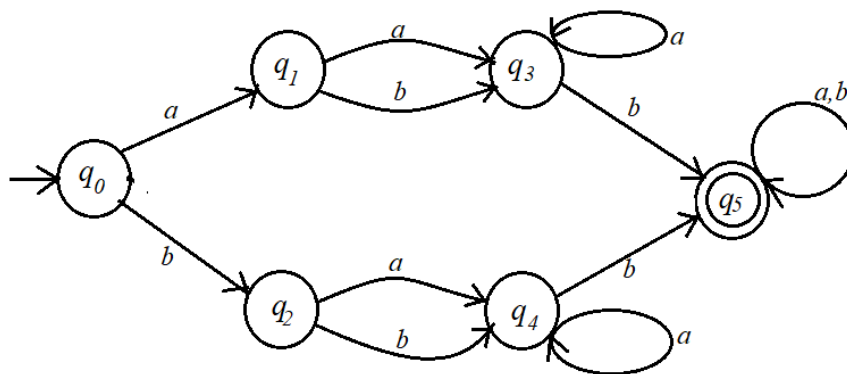
```
begin
  if a < b
    then res := a + 1
    else begin
      res := 0;
      while res < b do
        res := res + 2;
        res := res + b
      end
    end
  end.
```

## Вариант 6

1. Построить конечный автомат по заданному регулярному выражению:  $(aa)^*|(bb)^*$ .
2. Построить ДКА, эквивалентный заданному НКА



3. Минимизировать заданный конечный автомат:



4. Дана грамматика ( $E$  – стартовый символ):

$E \rightarrow E \& T \mid T$

$T \rightarrow T \vee F \mid F$

$F \rightarrow \neg F \mid (E) \mid a$

Построить вывод слова  $\neg a \& \neg(\neg(a \& a) \vee \neg a)$ .

5. Дана грамматика ( $P$  – стартовый символ):

$P \rightarrow S$ .

$E \rightarrow E+E \mid E<E \mid \text{id} \mid \text{num}$

$S \rightarrow \text{id} := E$

$S \rightarrow \text{if } E \text{ then } S \text{ else } S$

$S \rightarrow \text{while } E \text{ do } S$

$S \rightarrow \text{begin } L \text{ end}$

$L \rightarrow S$

$L \rightarrow S; L$

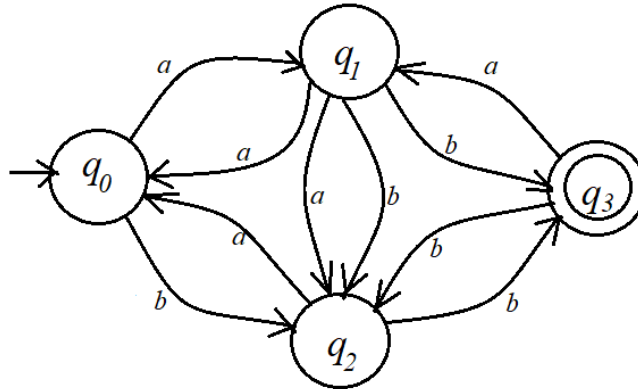
Лексический анализатор превращает все идентификаторы, отличные от ключевых слов языка, в символ `id`, а все целые числа – в `num`. Переводы строк, пробелы и табуляции

игнорируются. Постройте дерево разбора указанного текста:

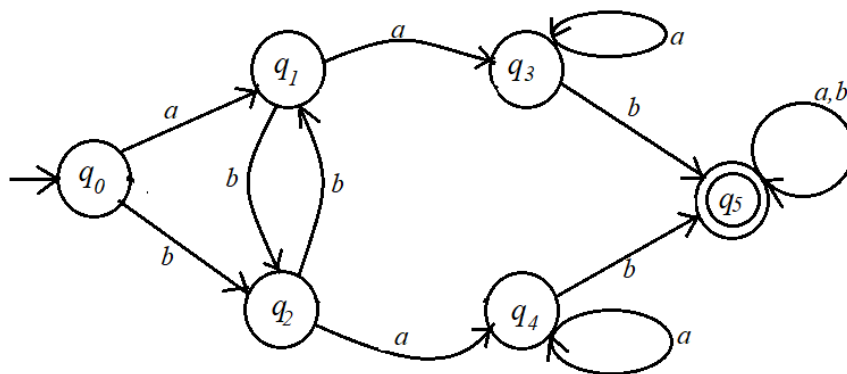
```
begin
  if a < b
    then begin
      res := 0;
      while res < b do
        res := res + 2;
        res := res + b
      end
    else res := a + 1
  end.
```

## Вариант 7

1. Построить конечный автомат по заданному регулярному выражению:  $(a|ab)^*b^*$ .
2. Построить ДКА, эквивалентный заданному НКА



3. Минимизировать заданный конечный автомат:



4. Дана грамматика (E – стартовый символ):  
 $E \rightarrow E \& T \mid T$   
 $T \rightarrow T \vee F \mid F$   
 $F \rightarrow \neg F \mid (E) \mid a$   
 Построить вывод слова  $\neg(a\&a) \vee \neg(a\&a) \vee \neg a$ .

5. Дана грамматика (P – стартовый символ):  
 $P \rightarrow S$   
 $E \rightarrow E+E \mid E<E \mid \text{id} \mid \text{num}$   
 $S \rightarrow \text{id} := E$   
 $S \rightarrow \text{if } E \text{ then } S \text{ else } S$   
 $S \rightarrow \text{while } E \text{ do } S$   
 $S \rightarrow \text{begin } L \text{ end}$   
 $L \rightarrow S$   
 $L \rightarrow S; L$

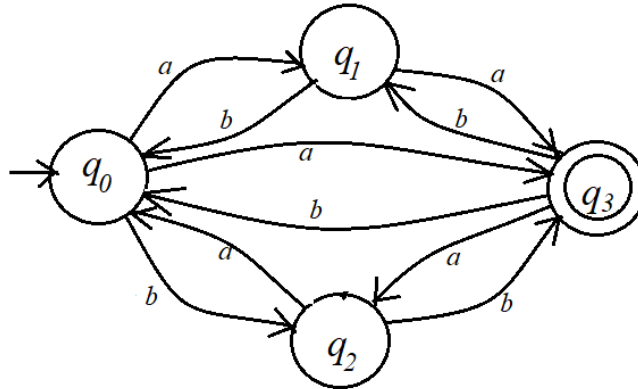
Лексический анализатор превращает все идентификаторы, отличные от ключевых слов языка, в символ `id`, а все целые числа – в `num`. Переводы строк, пробелы и табуляции

игнорируются. Постройте дерево разбора указанного текста:

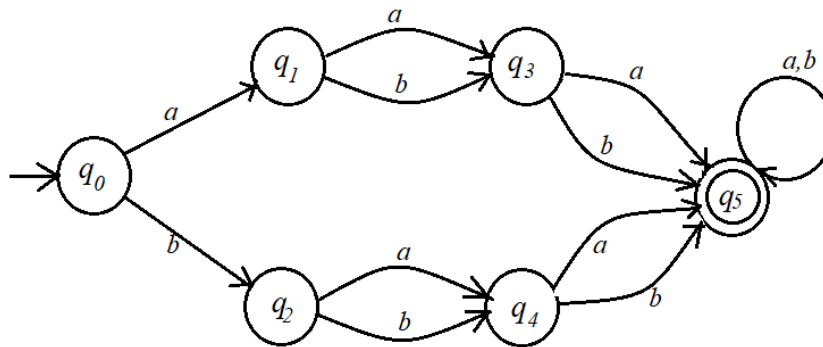
```
begin
  i := 0;
  j := 0;
  while i < 10 do begin
    if i < 5
      then j := j+2
      else j := i+j;
    i := i+1
  end
end.
```

## Вариант 8

1. Построить конечный автомат по заданному регулярному выражению:  $a^{ast}|(ab)^*b^*$ .
2. Построить ДКА, эквивалентный заданному НКА



3. Минимизировать заданный конечный автомат:



4. Дана грамматика ( $E$  – стартовый символ):  
 $E \rightarrow E \& T \mid T$   
 $T \rightarrow T \vee F \mid F$   
 $F \rightarrow \neg F \mid (E) \mid a$   
 Построить вывод слова  $\neg(a\&a) \vee \neg a \vee \neg(a\&a)$ .

5. Дана грамматика ( $P$  – стартовый символ):  
 $P \rightarrow S$   
 $E \rightarrow E+E \mid E<E \mid \text{id} \mid \text{num}$   
 $S \rightarrow \text{id} := E$   
 $S \rightarrow \text{if } E \text{ then } S \text{ else } S$   
 $S \rightarrow \text{while } E \text{ do } S$   
 $S \rightarrow \text{begin } L \text{ end}$   
 $L \rightarrow S$   
 $L \rightarrow S; L$

Лексический анализатор превращает все идентификаторы, отличные от ключевых слов языка, в символ `id`, а все целые числа – в `num`. Переводы строк, пробелы и табуляции

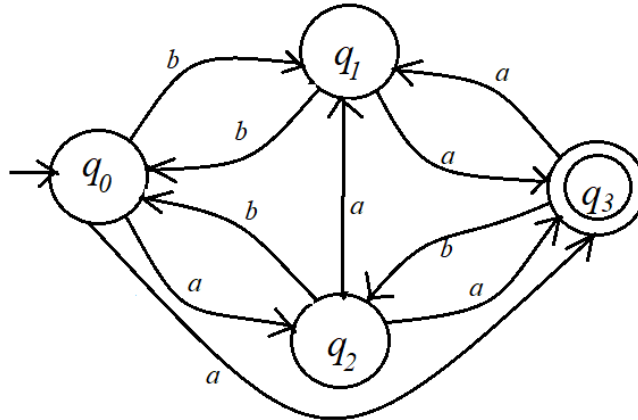
игнорируются. Постройте дерево разбора указанного текста:

```
begin
  if a < b
    then res := a + 1
    else begin
      res := 0;
      while res < b do
        res := res + 2;
        res := res + b
      end
    end
  end.
```

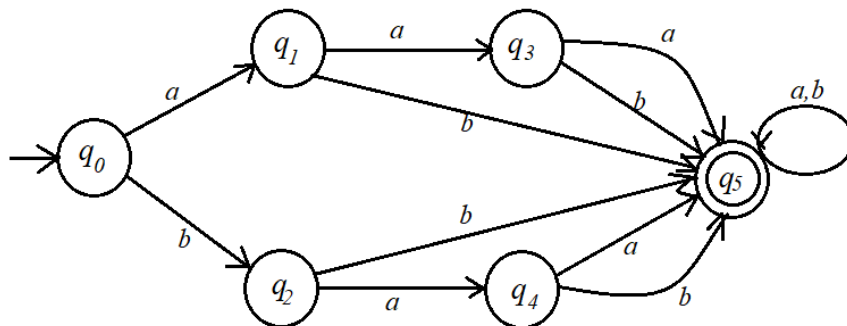


## Вариант 9

1. Построить конечный автомат по заданному регулярному выражению:  $a^{ast}|b^*|(ab)^*$ .
2. Построить ДКА, эквивалентный заданному НКА



3. Минимизировать заданный конечный автомат:



4. Дана грамматика (E – стартовый символ):

$E \rightarrow E \& T \mid T$

$T \rightarrow T \vee F \mid F$

$F \rightarrow \neg F \mid (E) \mid a$

Построить вывод слова  $(a\&a)\&\neg a \vee \neg(\neg a\&a)$ .

5. Дана грамматика (P – стартовый символ):

$P \rightarrow S$ .

$E \rightarrow E+E \mid E<E \mid \text{id} \mid \text{num}$

$S \rightarrow \text{id} := E$

$S \rightarrow \text{if } E \text{ then } S \text{ else } S$

$S \rightarrow \text{while } E \text{ do } S$

$S \rightarrow \text{begin } L \text{ end}$

$L \rightarrow S$

$L \rightarrow S; L$

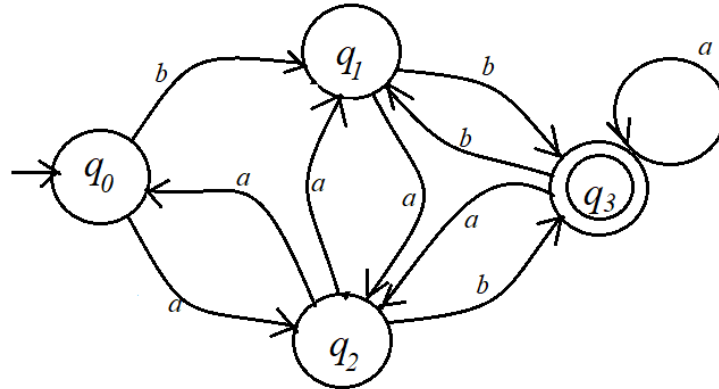
Лексический анализатор превращает все идентификаторы, отличные от ключевых слов языка, в символ `id`, а все целые числа – в `num`. Переводы строк, пробелы и табуляции

игнорируются. Постройте дерево разбора указанного текста:

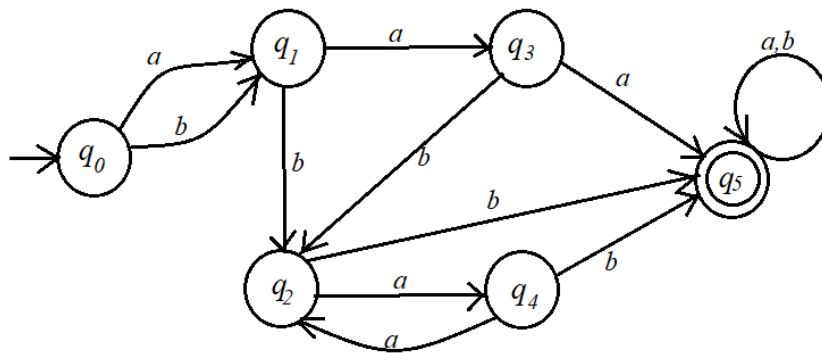
```
begin
  i := 0;
  j := 0;
  while i < 10 do begin
    if i < 5
    then j := j+2
    else j := i+j;
  i := i+1
  end
end.
```

## Вариант 10

1. Построить конечный автомат по заданному регулярному выражению:  $(abba|a^*)^*$ .
2. Построить ДКА, эквивалентный заданному НКА



3. Минимизировать заданный конечный автомат:



4. Дана грамматика ( $E$  – стартовый символ):

$E \rightarrow E \& T \mid T$

$T \rightarrow T \vee F \mid F$

$F \rightarrow \neg F \mid (E) \mid a$

Построить вывод слова  $(a\&\neg a) \vee (\neg a \vee \neg a)$ .

5. Дана грамматика ( $P$  – стартовый символ):

$P \rightarrow S$ .

$E \rightarrow E+E \mid E<E \mid \text{id} \mid \text{num}$

$S \rightarrow \text{id} := E$

$S \rightarrow \text{if } E \text{ then } S \text{ else } S$

$S \rightarrow \text{while } E \text{ do } S$

$S \rightarrow \text{begin } L \text{ end}$

$L \rightarrow S$

$L \rightarrow S; L$

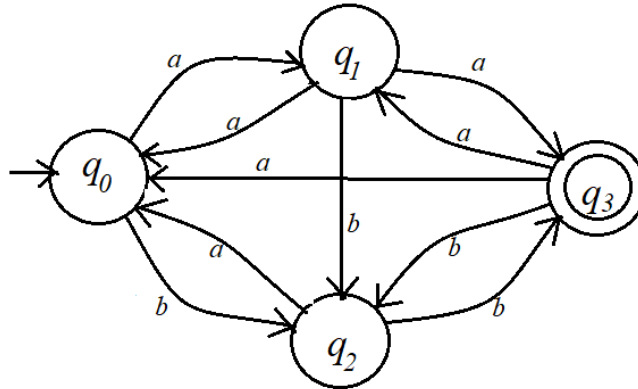
Лексический анализатор превращает все идентификаторы, отличные от ключевых слов языка, в символ `id`, а все целые числа – в `num`. Переводы строк, пробелы и табуляции

игнорируются. Постройте дерево разбора указанного текста:

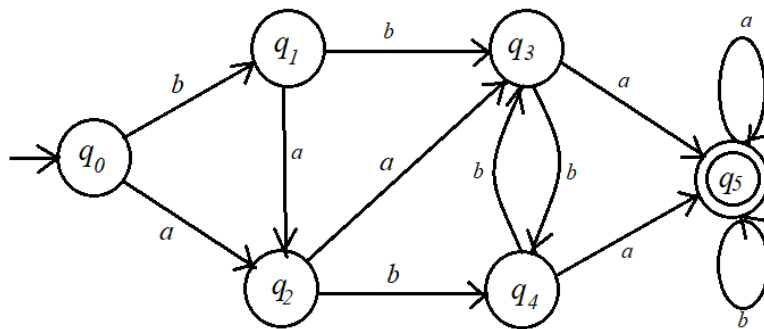
```
begin
  if a < b
    then begin
      res := 0;
      while res < b do
        res := res + 2;
        res := res + b
      end
    else res := a + 1
  end.
```

## Вариант 11

1. Построить конечный автомат по заданному регулярному выражению:  $(aa|bb|ab)^*a$ .
2. Построить ДКА, эквивалентный заданному НКА



3. Минимизировать заданный конечный автомат:



4. Дана грамматика ( $E$  – стартовый символ):

$E \rightarrow E \& T \mid T$

$T \rightarrow T \vee F \mid F$

$F \rightarrow \neg F \mid (E) \mid a$

Построить вывод слова  $\neg(\neg a \vee (\neg a \& \neg a))$ .

5. Дана грамматика ( $P$  – стартовый символ):

$P \rightarrow S$ .

$E \rightarrow E+E \mid E<E \mid \text{id} \mid \text{num}$

$S \rightarrow \text{id} := E$

$S \rightarrow \text{if } E \text{ then } S \text{ else } S$

$S \rightarrow \text{while } E \text{ do } S$

$S \rightarrow \text{begin } L \text{ end}$

$L \rightarrow S$

$L \rightarrow S; L$

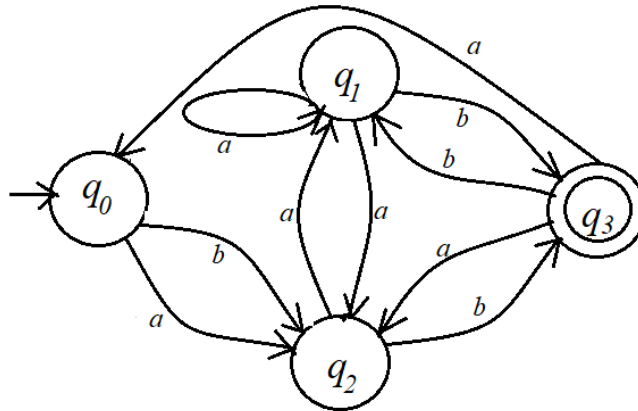
Лексический анализатор превращает все идентификаторы, отличные от ключевых слов языка, в символ `id`, а все целые числа – в `num`. Переводы строк, пробелы и табуляции

игнорируются. Постройте дерево разбора указанного текста:

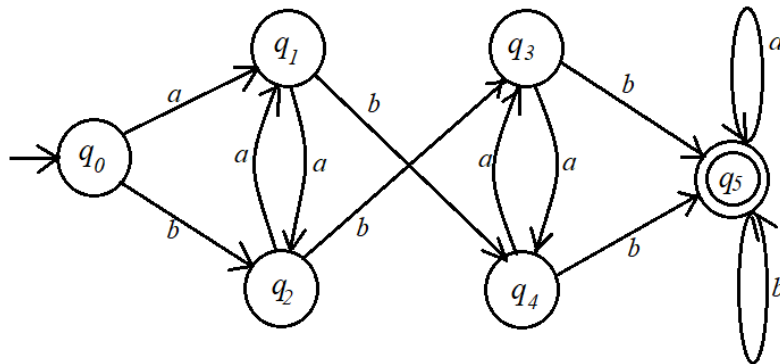
```
begin
  if a < b
    then res := a + 1
    else begin
      res := 0;
      while res < b do
        res := res + 2;
        res := res + b
      end
    end
  end.
```

## Вариант 12

1. Построить конечный автомат по заданному регулярному выражению:  $(aa|(bb)^*|ab)^*$ .
2. Построить ДКА, эквивалентный заданному НКА



3. Минимизировать заданный конечный автомат:



4. Дана грамматика (E – стартовый символ):

$E \rightarrow E \& T \mid T$

$T \rightarrow T \vee F \mid F$

$F \rightarrow \neg F \mid (E) \mid a$

Построить вывод слова  $\neg((\neg a \& \neg a) \vee \neg a)$ .

5. Дана грамматика (P – стартовый символ):

$P \rightarrow S$ .

$E \rightarrow E+E \mid E<E \mid \text{id} \mid \text{num}$

$S \rightarrow \text{id} := E$

$S \rightarrow \text{if } E \text{ then } S \text{ else } S$

$S \rightarrow \text{while } E \text{ do } S$

$S \rightarrow \text{begin } L \text{ end}$

$L \rightarrow S$

$L \rightarrow S; L$

Лексический анализатор превращает все идентификаторы, отличные от ключевых слов языка, в символ `id`, а все целые числа – в `num`. Переводы строк, пробелы и табуляции

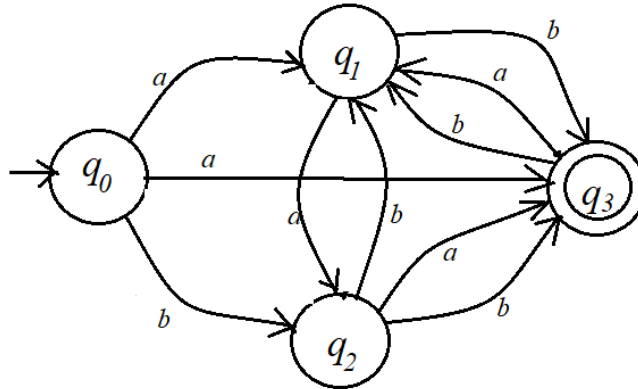
игнорируются. Постройте дерево разбора указанного текста:

```
begin
  i := 0;
  j := 0;
  while i < 10 do begin
    if i < 5
      then j := j+2
      else j := i+j;
    i := i+1
  end
end.
```

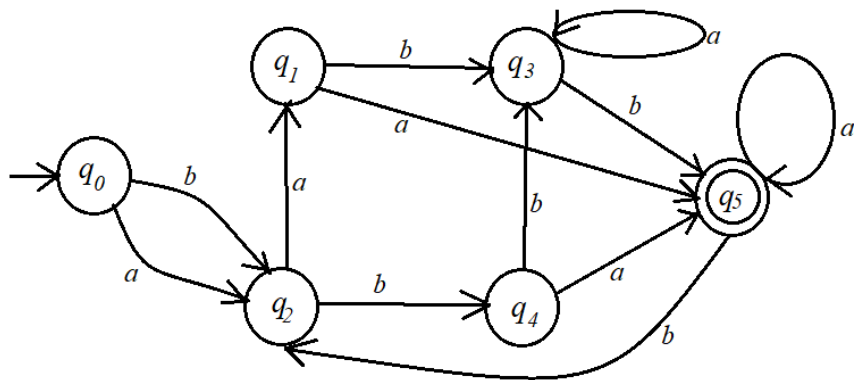


## Вариант 13

1. Построить конечный автомат по заданному регулярному выражению:  $(aa)^*|(bb)^*|(ab)^*$ .
2. Построить ДКА, эквивалентный заданному НКА



3. Минимизировать заданный конечный автомат:



4. Дана грамматика (E – стартовый символ):

$E \rightarrow E \& T \mid T$

$T \rightarrow T \vee F \mid F$

$F \rightarrow \neg F \mid (E) \mid a$

Построить вывод слова  $\neg((\neg a \vee \neg a) \& \neg a)$ .

5. Дана грамматика (P – стартовый символ):

$P \rightarrow S$ .

$E \rightarrow E+E \mid E<E \mid \text{id} \mid \text{num}$

$S \rightarrow \text{id} := E$

$S \rightarrow \text{if } E \text{ then } S \text{ else } S$

$S \rightarrow \text{while } E \text{ do } S$

$S \rightarrow \text{begin } L \text{ end}$

$L \rightarrow S$

$L \rightarrow S; L$

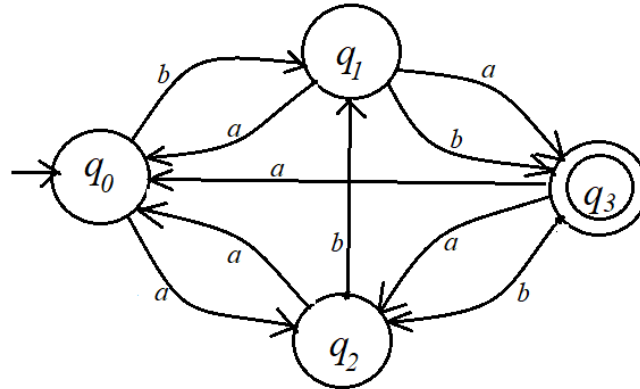
Лексический анализатор превращает все идентификаторы, отличные от ключевых слов языка, в символ `id`, а все целые числа – в `num`. Переводы строк, пробелы и табуляции

игнорируются. Постройте дерево разбора указанного текста:

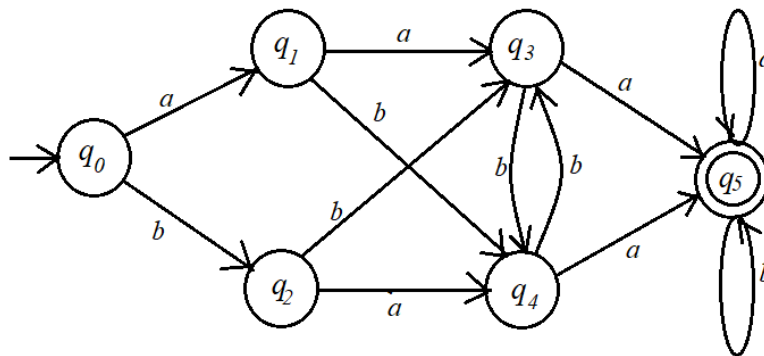
```
begin
  res := 0;
  while res < b do
    res := res + 2;
  if a < b
    then res := res + 1
    else begin
      a := res + b;
      b := a;
    end
  end
end.
```

## Вариант 14

1. Построить конечный автомат по заданному регулярному выражению:  $((aa)^*|bb)^*$ .
2. Построить ДКА, эквивалентный заданному НКА



3. Минимизировать заданный конечный автомат:



4. Дана грамматика ( $E$  – стартовый символ):

$E \rightarrow E \& T \mid T$

$T \rightarrow T \vee F \mid F$

$F \rightarrow \neg F \mid (E) \mid a$

Построить вывод слова  $((\neg a \vee \neg a) \& \neg a) \& \neg a$ .

5. Дана грамматика ( $P$  – стартовый символ):

$P \rightarrow S$ .

$E \rightarrow E+E \mid E<E \mid \text{id} \mid \text{num}$

$S \rightarrow \text{id} := E$

$S \rightarrow \text{if } E \text{ then } S \text{ else } S$

$S \rightarrow \text{while } E \text{ do } S$

$S \rightarrow \text{begin } L \text{ end}$

$L \rightarrow S$

$L \rightarrow S; L$

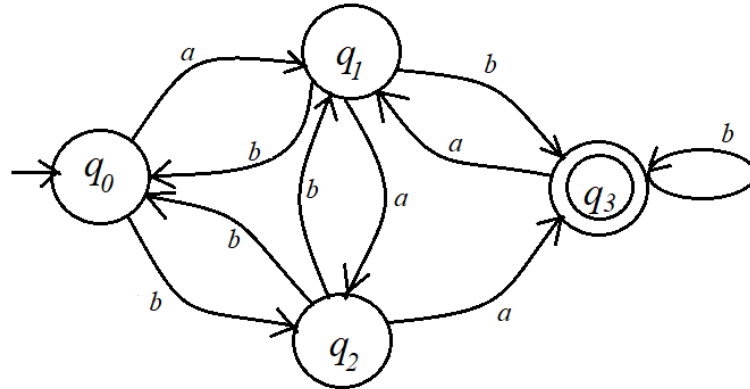
Лексический анализатор превращает все идентификаторы, отличные от ключевых слов языка, в символ `id`, а все целые числа – в `num`. Переводы строк, пробелы и табуляции

игнорируются. Постройте дерево разбора указанного текста:

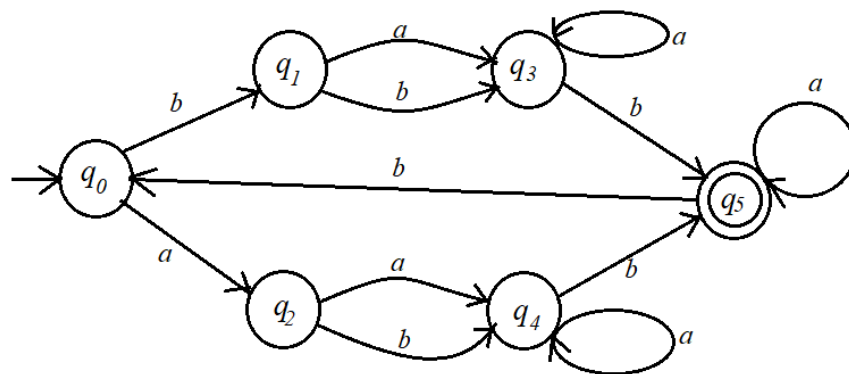
```
begin
  if a < b
    then begin
      res := 0;
      while res < b do
        res := res + 2;
        res := res + b
      end
    else res := a + 1
  end.
```

## Вариант 15

1. Построить конечный автомат по заданному регулярному выражению:  $(aa)^*|(bb)^*$ .
2. Построить ДКА, эквивалентный заданному НКА



3. Минимизировать заданный конечный автомат:



4. Дана грамматика (E – стартовый символ):

$E \rightarrow E \& T \mid T$

$T \rightarrow T \vee F \mid F$

$F \rightarrow \neg F \mid (E) \mid a$

Построить вывод слова  $\neg a \vee ((\neg a \vee \neg a) \& \neg a)$ .

5. Дана грамматика (P – стартовый символ):

$P \rightarrow S$ .

$E \rightarrow E+E \mid E<E \mid \text{id} \mid \text{num}$

$S \rightarrow \text{id} := E$

$S \rightarrow \text{if } E \text{ then } S \text{ else } S$

$S \rightarrow \text{while } E \text{ do } S$

$S \rightarrow \text{begin } L \text{ end}$

$L \rightarrow S$

$L \rightarrow S; L$

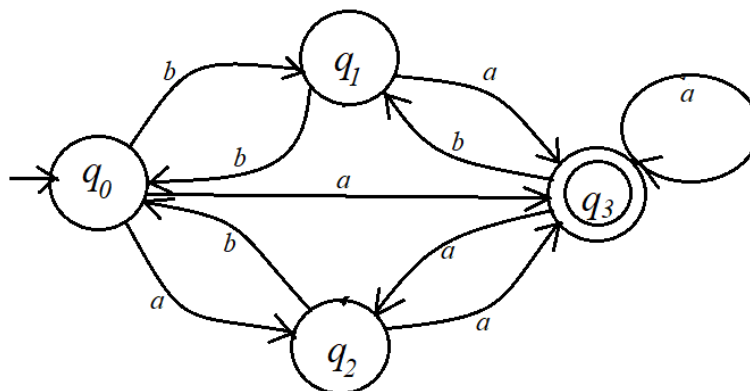
Лексический анализатор превращает все идентификаторы, отличные от ключевых слов языка, в символ `id`, а все целые числа – в `num`. Переводы строк, пробелы и табуляции

игнорируются. Постройте дерево разбора указанного текста:

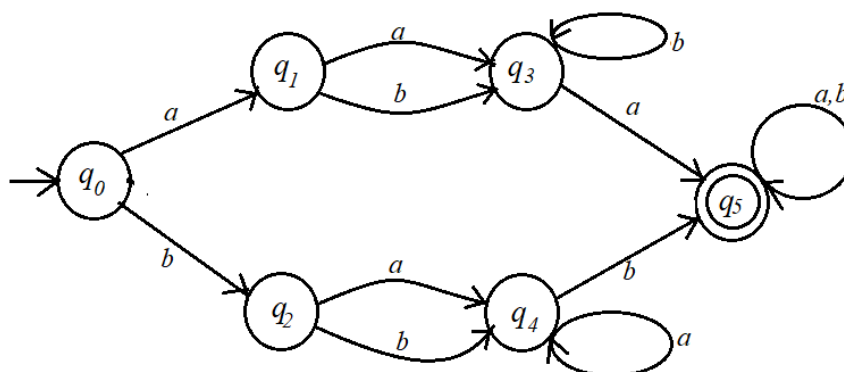
```
begin
  if a < b
    then res := a + 1
    else begin
      res := 0;
      while res < b do
        res := res + 2;
        res := res + b
      end
    end
  end.
```

## Вариант 16

1. Построить конечный автомат по заданному регулярному выражению:  $(aaa|bb|abab)^*$ .
2. Построить ДКА, эквивалентный заданному НКА



3. Минимизировать заданный конечный автомат:



4. Дана грамматика (E – стартовый символ):

$E \rightarrow E \& T \mid T$

$T \rightarrow T \vee F \mid F$

$F \rightarrow \neg F \mid (E) \mid a$

Построить вывод слова  $\neg a \& ((a \vee \neg a) \& \neg a)$ .

5. Дана грамматика (P – стартовый символ):

$P \rightarrow S$ .

$E \rightarrow E + E \mid E < E \mid \text{id} \mid \text{num}$

$S \rightarrow \text{id} := E$

$S \rightarrow \text{if } E \text{ then } S \text{ else } S$

$S \rightarrow \text{while } E \text{ do } S$

$S \rightarrow \text{begin } L \text{ end}$

$L \rightarrow S$

$L \rightarrow S; L$

Лексический анализатор превращает все идентификаторы, отличные от ключевых слов языка, в символ `id`, а все целые числа – в `num`. Переводы строк, пробелы и табуляции

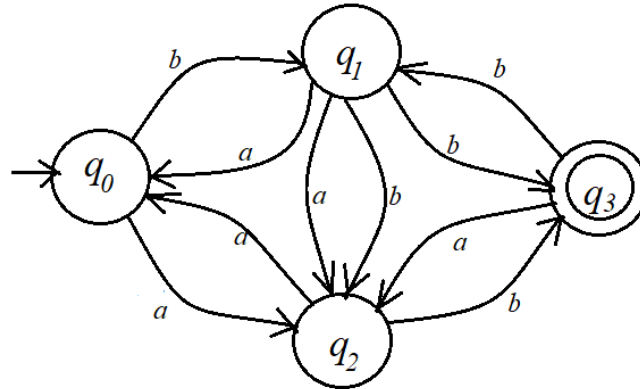
игнорируются. Постройте дерево разбора указанного текста:

```
begin
  i := 0;
  j := 0;
  while i < 10 do begin
    if i < 5
      then j := j+2
      else j := i+j;
    i := i+1
  end
end.
```

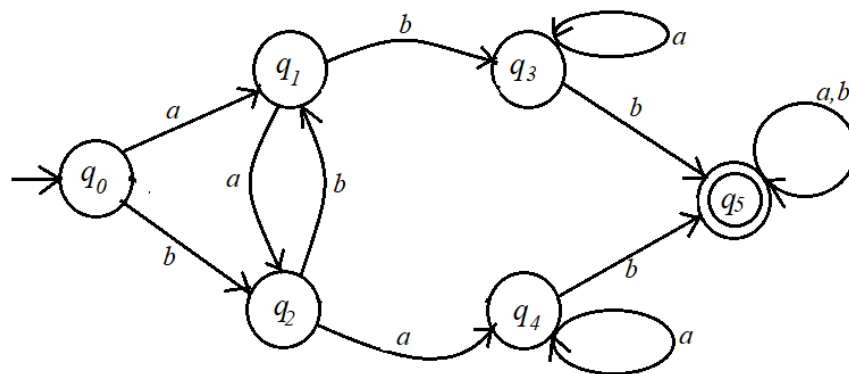


## Вариант 17

1. Построить конечный автомат по заданному регулярному выражению:  $(aa|bb)^*|a^*$ .
2. Построить ДКА, эквивалентный заданному НКА



3. Минимизировать заданный конечный автомат:



4. Дана грамматика (E – стартовый символ):

$E \rightarrow E \& T \mid T$

$T \rightarrow T \vee F \mid F$

$F \rightarrow \neg F \mid (E) \mid a$

Построить вывод слова  $(\neg a \vee a) \& \neg(a \vee \neg a)$ .

5. Дана грамматика (P – стартовый символ):

$P \rightarrow S$ .

$E \rightarrow E+E \mid E<E \mid \text{id} \mid \text{num}$

$S \rightarrow \text{id} := E$

$S \rightarrow \text{if } E \text{ then } S \text{ else } S$

$S \rightarrow \text{while } E \text{ do } S$

$S \rightarrow \text{begin } L \text{ end}$

$L \rightarrow S$

$L \rightarrow S; L$

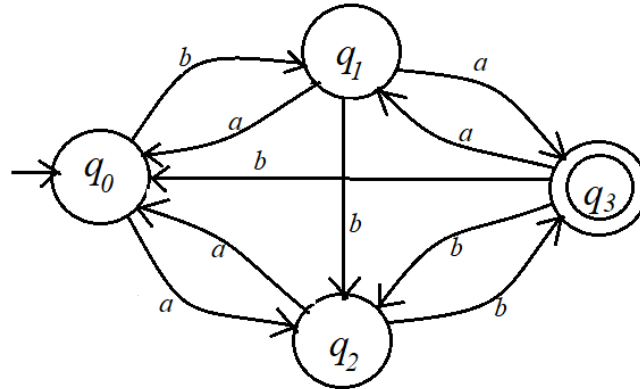
Лексический анализатор превращает все идентификаторы, отличные от ключевых слов языка, в символ `id`, а все целые числа – в `num`. Переводы строк, пробелы и табуляции

игнорируются. Постройте дерево разбора указанного текста:

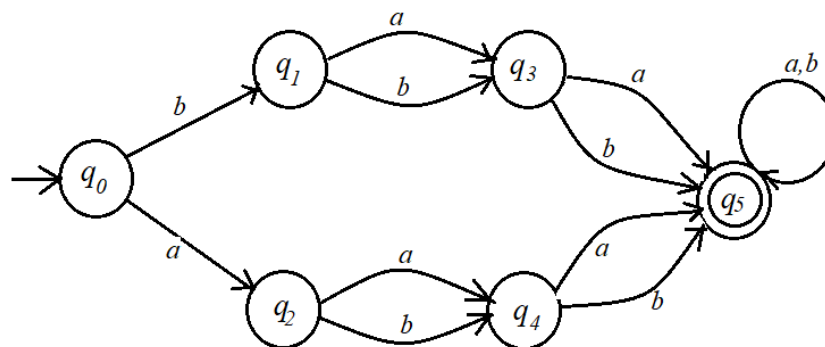
```
begin
  if a < b
    then res := a + 1
    else begin
      res := 0;
      while res < b do
        res := res + 2;
        res := res + b
      end
    end
  end.
```

## Вариант 18

1. Построить конечный автомат по заданному регулярному выражению:  $(a^*|bb)^*$ .
2. Построить ДКА, эквивалентный заданному НКА



3. Минимизировать заданный конечный автомат:



4. Дана грамматика (E – стартовый символ):

$E \rightarrow E \& T \mid T$

$T \rightarrow T \vee F \mid F$

$F \rightarrow \neg F \mid (E) \mid a$

Построить вывод слова  $\neg(a \vee a) \& \neg(a \vee \neg a)$ .

5. Дана грамматика (P – стартовый символ):

$P \rightarrow S$ .

$E \rightarrow E+E \mid E<E \mid \text{id} \mid \text{num}$

$S \rightarrow \text{id} := E$

$S \rightarrow \text{if } E \text{ then } S \text{ else } S$

$S \rightarrow \text{while } E \text{ do } S$

$S \rightarrow \text{begin } L \text{ end}$

$L \rightarrow S$

$L \rightarrow S; L$

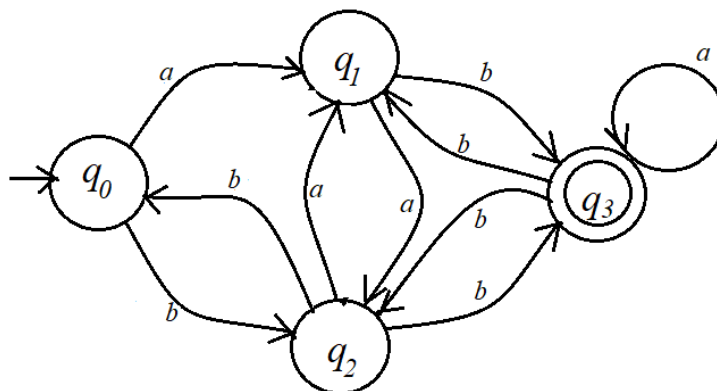
Лексический анализатор превращает все идентификаторы, отличные от ключевых слов языка, в символ `id`, а все целые числа – в `num`. Переводы строк, пробелы и табуляции

игнорируются. Постройте дерево разбора указанного текста:

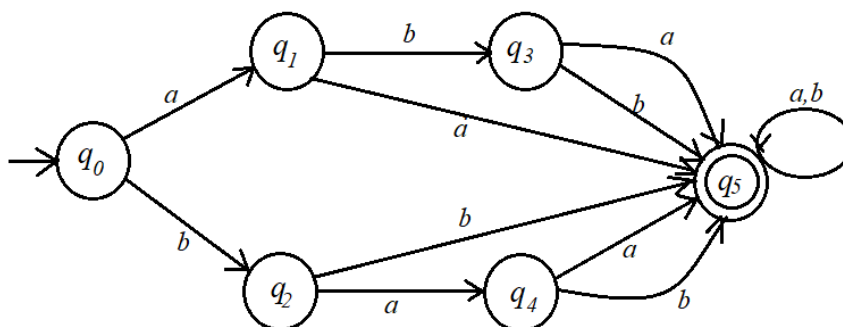
```
begin
  i := 0;
  j := 0;
  while i < 10 do begin
    if i < 5
      then j := j+2
      else j := i+j;
    i := i+1
  end
end.
```

## Вариант 19

1. Построить конечный автомат по заданному регулярному выражению:  $a^*|b^*|(ab)^*$ .
2. Построить ДКА, эквивалентный заданному НКА



3. Минимизировать заданный конечный автомат:



4. Дана грамматика (E – стартовый символ):

$E \rightarrow E \& T \mid T$

$T \rightarrow T \vee F \mid F$

$F \rightarrow \neg F \mid (E) \mid a$

Построить вывод слова  $\neg(a \vee a \& a) \vee (a \vee \neg a)$ .

5. Дана грамматика (P – стартовый символ):

$P \rightarrow S$ .

$E \rightarrow E+E \mid E<E \mid \text{id} \mid \text{num}$

$S \rightarrow \text{id} := E$

$S \rightarrow \text{if } E \text{ then } S \text{ else } S$

$S \rightarrow \text{while } E \text{ do } S$

$S \rightarrow \text{begin } L \text{ end}$

$L \rightarrow S$

$L \rightarrow S; L$

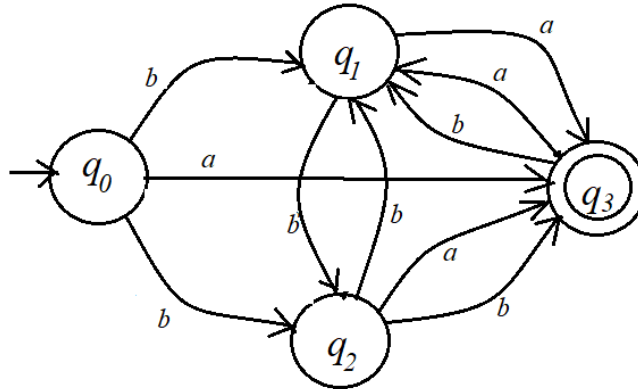
Лексический анализатор превращает все идентификаторы, отличные от ключевых слов языка, в символ `id`, а все целые числа – в `num`. Переводы строк, пробелы и табуляции

игнорируются. Постройте дерево разбора указанного текста:

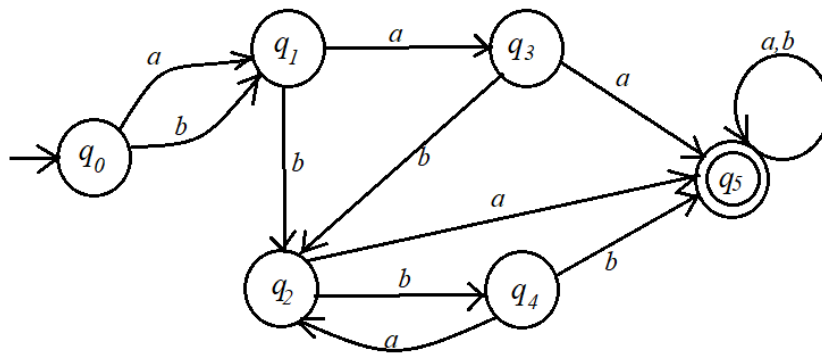
```
begin
  res := 0;
  while res < b do
    res := res + 2;
  if a < b
    then res := res + 1
    else begin
      a := res + b;
      b := a;
    end
  end
end.
```

## Вариант 20

1. Построить конечный автомат по заданному регулярному выражению:  $abab|baba|a^*$ .
2. Построить ДКА, эквивалентный заданному НКА



3. Минимизировать заданный конечный автомат:



4. Дана грамматика ( $E$  – стартовый символ):

$E \rightarrow E \& T \mid T$

$T \rightarrow T \vee F \mid F$

$F \rightarrow \neg F \mid (E) \mid a$

Построить вывод слова  $\neg a \vee (a \& a) \vee (a \vee \neg a)$ .

5. Дана грамматика ( $P$  – стартовый символ):

$P \rightarrow S$ .

$E \rightarrow E + E \mid E < E \mid \text{id} \mid \text{num}$

$S \rightarrow \text{id} := E$

$S \rightarrow \text{if } E \text{ then } S \text{ else } S$

$S \rightarrow \text{while } E \text{ do } S$

$S \rightarrow \text{begin } L \text{ end}$

$L \rightarrow S$

$L \rightarrow S; L$

Лексический анализатор превращает все идентификаторы, отличные от ключевых слов языка, в символ `id`, а все целые числа – в `num`. Переводы строк, пробелы и табуляции

игнорируются. Постройте дерево разбора указанного текста:

```
begin
  if a < b
    then res := a + 1
    else begin
      res := 0;
      while res < b do
        res := res + 2;
        res := res + b
      end
    end
  end.
```