Variant 1

AF=(Q, Σ, δ, q_0, F),
Q = {q_0, q_1, q_2 , q_3},
Σ = {a, c, b}, F = {q_2}.
δ (q_0, a) = q_0,
δ (q_0, a) = q_1 ,
δ (q_1, c) = q_1,
δ (q_1, b) = q_2,
δ (q_2, b) = q_3,
δ (q_3, a) = q_1 .

Theme : Finite Automata

1. Present the automaton in form of graph. Is this automaton deterministic or no? Why?
2. Convert a Finite Automaton to the Regular Grammar.
3. Transform nondeterministic finite automaton (NFA) into a deterministic automaton (DFA). Present the DFA in form of graph.
4. Present 1 string based on the input alphabet Σ that will be not accepted by the automaton. For showing use the Multiple Run option from menu Input from JFLAP.
5. For the finite automaton AF=(Q, Σ, δ, q_0, F) present five strings that will be accepted. The length of strings should be more then n+2, where n is the number of stares from Q. For showing use the Multiple Run option from menu Input from JFLAP. Elaborate the program that for the obtained DFA will generate the strings by the given lenght.
6. For each string x, present the configuration sequence for the acceptance of this string. To show the acceptance use the option Step with Closure and Fast Run from menu Input from JFLAP.
7. For each obrtained 5 strings build the decompozition x=uvw , using the pumping lemma.
8. Obtained DFA minimize.

Variant 2

AF=(Q, Σ, δ, q_0, F),
Q = { q_0, q_1, q_2 , q_3, q_4 },
Σ = { a, b, c }, F = { q_4 }.
δ (q_0, a) = q_1 ,
δ (q_1, b) = q_2 ,
δ (q_1, b) = q_3,
δ (q_2, c) = q_3,
δ (q_3, a) = q_3,
δ (q_3, b) = q_4 .

Theme : Finite Automata

1. Present the automaton in form of graph. Is this automaton deterministic or no? Why?
2. Convert a Finite Automaton to the Regular Grammar.
3. Transform nondeterministic finite automaton (NFA) into a deterministic automaton (DFA). Present the DFA in form of graph.
4. Present 1 string based on the input alphabet Σ that will be not accepted by the automaton. For showing use the Multiple Run option from menu Input from JFLAP.
5. For the finite automaton AF=(Q, Σ, δ, q_0, F) present five strings that will be accepted. The length of strings should be more then n+2, where n is the number of stares from Q. For showing use the Multiple Run option from menu Input from JFLAP. Elaborate the program that for the obtained DFA will generate the strings by the given lenght.
6. For each string x, present the configuration sequence for the acceptance of this string. To show the acceptance use the option Step with Closure and Fast Run from menu Input from JFLAP.
7. For each obrtained 5 strings build the decompozition x=uvw , using the pumping lemma.
8. Obtained DFA minimize.

Variant 3

AF=(Q, Σ, δ, q_0, F),
Q = { q_0, q_1, q_2 , q_3, q_4 },
Σ = { a, b }, F = { q_4 }.
δ (q_0, a) = q_1 ,
δ (q_1, b) = q_1 ,
δ (q_1, a) = q_2,
δ (q_2, b) = q_2,
δ (q_2, b) = q_3,
δ (q_3, b) = q_4,
δ (q_3, a) = q_1.

Theme : Finite Automata

1. Present the automaton in form of graph. Is this automaton deterministic or no? Why?
2. Convert a Finite Automaton to the Regular Grammar.
3. Transform nondeterministic finite automaton (NFA) into a deterministic automaton (DFA). Present the DFA in form of graph.
4. Present 1 string based on the input alphabet Σ that will be not accepted by the automaton. For showing use the Multiple Run option from menu Input from JFLAP.
5. For the finite automaton AF=(Q, Σ, δ, q_0, F) present five strings that will be accepted. The length of strings should be more then n+2, where n is the number of stares from Q. For showing use the Multiple Run option from menu Input from JFLAP. Elaborate the program that for the obtained DFA will verify if string is accepted or no.
6. For each string x, present the configuration sequence for the acceptance of this string. To show the acceptance use the option Step with Closure and Fast Run from menu Input from JFLAP.
7. For each obrtained 5 strings build the decompozition x=uvw , using the pumping lemma.
8. Obtained DFA minimize.

Variant 4

AF=(Q, Σ, δ, q_0, F),
Q = { q_0, q_1, q_2 , q_3 },
Σ = { a, b }, F = { q_3 }.
δ (q_0, a) = q_1 ,
δ (q_0, a) = q_2 ,
δ (q_1, b) = q_1,
δ (q_1, a) = q_2,
δ (q_2, a) = q_1,
δ (q_2, b) = q_3.

Theme : Finite Automata

1. Present the automaton in form of graph. Is this automaton deterministic or no? Why?
2. Convert a Finite Automaton to the Regular Grammar.
3. Transform nondeterministic finite automaton (NFA) into a deterministic automaton (DFA). Present the DFA in form of graph.
4. Present 1 string based on the input alphabet Σ that will be not accepted by the automaton. For showing use the Multiple Run option from menu Input from JFLAP.
5. For the finite automaton AF=(Q, Σ, δ, q_0, F) present five strings that will be accepted. The length of strings should be more then n+2, where n is the number of stares from Q. For showing use the Multiple Run option from menu Input from JFLAP. Elaborate the program that for the obtained DFA will verify if string is accepted or no.
6. For each string x, present the configuration sequence for the acceptance of this string. To show the acceptance use the option Step with Closure and Fast Run from menu Input from JFLAP.
7. For each obrtained 5 strings build the decompozition x=uvw , using the pumping lemma.
8. Obtained DFA minimize.

Variant 5

AF=(Q, Σ, δ, q_0, F),
Q = { q_0, q_1, q_2 , q_3 },
Σ = { a, b }, F = { q_3 }.
δ (q_0, a) = q_1 ,
δ (q_0, b) = q_0 ,
δ (q_1, a) = q_2,
δ (q_1, a) = q_3,
δ (q_2, a) = q_3,
δ (q_2, b) = q_0.

Theme : Finite Automata

1. Present the automaton in form of graph. Is this automaton deterministic or no? Why?
2. Convert a Finite Automaton to the Regular Grammar.
3. Transform nondeterministic finite automaton (NFA) into a deterministic automaton (DFA). Present the DFA in form of graph.
4. Present 1 string based on the input alphabet Σ that will be not accepted by the automaton. For showing use the Multiple Run option from menu Input from JFLAP.
5. For the finite automaton AF=(Q, Σ, δ, q_0, F) present five strings that will be accepted. The length of strings should be more then n+2, where n is the number of stares from Q. For showing use the Multiple Run option from menu Input from JFLAP. Elaborate the program that for the obtained DFA will generate the strings by the given lenght.
6. For each string x, present the configuration sequence for the acceptance of this string. To show the acceptance use the option Step with Closure and Fast Run from menu Input from JFLAP.
7. For each obrtained 5 strings build the decompozition x=uvw , using the pumping lemma.
8. Obtained DFA minimize.

Variant 6

AF=(Q, Σ, δ, q_0, F),
Q = { q_0, q_1, q_2 , q_3, q_4 },
Σ = { a, b }, F = { q_4 }.
δ (q_0, a) = q_1 ,
δ (q_1, b) = q_1 ,
δ (q_1, b) = q_2,
δ (q_2, b) = q_3,
δ (q_3, a) = q_1,
δ (q_2, a) = q_4.

Theme : Finite Automata

1. Present the automaton in form of graph. Is this automaton deterministic or no? Why?
2. Convert a Finite Automaton to the Regular Grammar.
3. Transform nondeterministic finite automaton (NFA) into a deterministic automaton (DFA). Present the DFA in form of graph.
4. Present 1 string based on the input alphabet Σ that will be not accepted by the automaton. For showing use the Multiple Run option from menu Input from JFLAP.
5. For the finite automaton AF=(Q, Σ, δ, q_0, F) present five strings that will be accepted. The length of strings should be more then n+2, where n is the number of stares from Q. For showing use the Multiple Run option from menu Input from JFLAP. Elaborate the program that for the obtained DFA will generate the strings by the given length.
6. For each string x, present the configuration sequence for the acceptance of this string. To show the acceptance use the option Step with Closure and Fast Run from menu Input from JFLAP.
7. For each obrtained 5 strings build the decompozition x=uvw , using the pumping lemma.
8. Obtained DFA minimize.

Variant 7

AF=(Q, Σ, δ, q_0, F),
Q = { q_0, q_1, q_2 , q_3 },
Σ = { a, b }, F = { q_3 }.
δ (q_0, a) = q_1,
δ (q_1, b) = q_2,
δ (q_2, b) = q_3,
δ (q_3, a) = q_1,
δ (q_2, b) = q_2,
δ (q_1, a) = q_1 .

Theme : Finite Automata

1. Present the automaton in form of graph. Is this automaton deterministic or no? Why?
2. Convert a Finite Automaton to the Regular Grammar.
3. Transform nondeterministic finite automaton (NFA) into a deterministic automaton (DFA). Present the DFA in form of graph.
4. Present 1 string based on the input alphabet Σ that will be not accepted by the automaton. For showing use the Multiple Run option from menu Input from JFLAP.
5. For the finite automaton AF=(Q, Σ, δ, q_0, F) present five strings that will be accepted. The length of strings should be more then n+2, where n is the number of stares from Q. For showing use the Multiple Run option from menu Input from JFLAP. Elaborate the program that for the obtained DFA will verify if string is accepted or no.
6. For each string x, present the configuration sequence for the acceptance of this string. To show the acceptance use the option Step with Closure and Fast Run from menu Input from JFLAP.
7. For each obrtained 5 strings build the decompozition x=uvw , using the pumping lemma.
8. Obtained DFA minimize.

Variant 8

AF=(Q, Σ, δ, q_0, F),
Q = { q_0, q_1, q_2 , q_3 , q_4 },
Σ = { a, b}, F = { q_3 }.
δ (q_0, a) = q_1,
δ (q_1, b) = q_2 ,
δ (q_2, b) = q_0 ,
δ (q_3, a) = q_4,
δ (q_4, a) = q_0,
δ (q_2, a) = q_3,
δ (q_1, b) = q_1.

Theme : Finite Automata

1. Present the automaton in form of graph. Is this automaton deterministic or no? Why?
2. Convert a Finite Automaton to the Regular Grammar.
3. Transform nondeterministic finite automaton (NFA) into a deterministic automaton (DFA). Present the DFA in form of graph.
4. Present 1 string based on the input alphabet Σ that will be not accepted by the automaton. For showing use the Multiple Run option from menu Input from JFLAP.
5. For the finite automaton AF=(Q, Σ, δ, q_0, F) present five strings that will be accepted. The length of strings should be more then n+2, where n is the number of stares from Q. For showing use the Multiple Run option from menu Input from JFLAP. Elaborate the program that for the obtained DFA will verify if string is accepted or no.
6. For each string x, present the configuration sequence for the acceptance of this string. To show the acceptance use the option Step with Closure and Fast Run from menu Input from JFLAP.
7. For each obrtained 5 strings build the decompozition x=uvw , using the pumping lemma.
8. Obtained DFA minimize.

Variant 9

AF=(Q, Σ, δ, q_0, F),
Q = { q_0, q_1, q_2 , q_3 , q_4 },
Σ = { a, b, c }, F = { q_4 }.
δ (q_0, a) = q_1,
δ (q_1, b) = q_2,
δ (q_2, c) = q_0 ,
δ (q_1, b) = q_3,
δ (q_3, a) = q_4,
δ (q_3, b) = q_0.

Theme : Finite Automata

1. Present the automaton in form of graph. Is this automaton deterministic or no? Why?
2. Convert a Finite Automaton to the Regular Grammar.
3. Transform nondeterministic finite automaton (NFA) into a deterministic automaton (DFA). Present the DFA in form of graph.
4. Present 1 string based on the input alphabet Σ that will be not accepted by the automaton. For showing use the Multiple Run option from menu Input from JFLAP.
5. For the finite automaton AF=(Q, Σ, δ, q_0, F) present five strings that will be accepted. The length of strings should be more then n+2, where n is the number of stares from Q. For showing use the Multiple Run option from menu Input from JFLAP. Elaborate the program that for the obtained DFA will generate the strings by the given lenght.
6. For each string x, present the configuration sequence for the acceptance of this string. To show the acceptance use the option Step with Closure and Fast Run from menu Input from JFLAP.
7. For each obrtained 5 strings build the decompozition x=uvw , using the pumping lemma.
8. Obtained DFA minimize.

Variant 10

AF=(Q, Σ, δ, q_0, F),
Q = { q_0, q_1, q_2 , q_3 },
Σ = { a, b, c }, F = { q_3 }.
δ (q_0, a) = q_1 ,
δ (q_1, b) = q_2 ,
δ (q_2, c) = q_3,
δ (q_3, a) = q_1,
δ (q_1, b) = q_1,
δ (q_0, b) = q_2.

Theme : Finite Automata

1. Present the automaton in form of graph. Is this automaton deterministic or no? Why?
2. Convert a Finite Automaton to the Regular Grammar.
3. Transform nondeterministic finite automaton (NFA) into a deterministic automaton (DFA). Present the DFA in form of graph.
4. Present 1 string based on the input alphabet Σ that will be not accepted by the automaton. For showing use the Multiple Run option from menu Input from JFLAP.
5. For the finite automaton AF=(Q, Σ, δ, q_0, F) present five strings that will be accepted. The length of strings should be more then n+2, where n is the number of stares from Q. For showing use the Multiple Run option from menu Input from JFLAP. Elaborate the program that for the obtained DFA will verify if string is accepted or no.
6. For each string x, present the configuration sequence for the acceptance of this string. To show the acceptance use the option Step with Closure and Fast Run from menu Input from JFLAP.
7. For each obrtained 5 strings build the decompozition x=uvw , using the pumping lemma.
8. Obtained DFA minimize.

Variant 11

AF=(Q, Σ, δ, q_0, F),
Q = { q_0, q_1, q_2 , q_3 },
Σ = { a, b, c }, F = { q_3 }.
δ (q_0, a) = q_1 ,
δ (q_1, b) = q_2,
δ (q_2, c) = q_0 ,
δ (q_1, a) = q_3,
δ (q_0, b) = q_2,
δ (q_2, c) = q_3.

Theme : Finite Automata

1. Present the automaton in form of graph. Is this automaton deterministic or no? Why?
2. Convert a Finite Automaton to the Regular Grammar.
3. Transform nondeterministic finite automaton (NFA) into a deterministic automaton (DFA). Present the DFA in form of graph.
4. Present 1 string based on the input alphabet Σ that will be not accepted by the automaton. For showing use the Multiple Run option from menu Input from JFLAP.
5. For the finite automaton AF=(Q, Σ, δ, q_0, F) present five strings that will be accepted. The length of strings should be more then n+2, where n is the number of stares from Q. For showing use the Multiple Run option from menu Input from JFLAP. Elaborate the program that for the obtained DFA will generate the strings by the given lenght.
6. For each string x, present the configuration sequence for the acceptance of this string. To show the acceptance use the option Step with Closure and Fast Run from menu Input from JFLAP.
7. For each obrtained 5 strings build the decompozition x=uvw , using the pumping lemma.
8. Obtained DFA minimize.

Variant 12

AF=(Q, Σ, δ, q_0, F),
Q = { q_0, q_1, q_2, q_3 },
Σ = { a, b, c }, F = { q_2 }.
δ (q_0, b) = q_0 ,
δ (q_0, a) = q_1 ,
δ (q_1, c) = q_1,
δ (q_1, a) = q_2,
δ (q_3, a) = q_1,
δ (q_3, a) = q_3,
δ (q_2, a) = q_3.

Theme : Finite Automata

1. Present the automaton in form of graph. Is this automaton deterministic or no? Why?
2. Convert a Finite Automaton to the Regular Grammar.
3. Transform nondeterministic finite automaton (NFA) into a deterministic automaton (DFA). Present the DFA in form of graph.
4. Present 1 string based on the input alphabet Σ that will be not accepted by the automaton. For showing use the Multiple Run option from menu Input from JFLAP.
5. For the finite automaton AF=(Q, Σ, δ, q_0, F) present five strings that will be accepted. The length of strings should be more then n+2, where n is the number of stares from Q. For showing use the Multiple Run option from menu Input from JFLAP. Elaborate the program that for the obtained DFA will verify if string is accepted or no.
6. For each string x, present the configuration sequence for the acceptance of this string. To show the acceptance use the option Step with Closure and Fast Run from menu Input from JFLAP.
7. For each obrtained 5 strings build the decompozition x=uvw , using the pumping lemma.
8. Obtained DFA minimize.

Theme : Finite Automata

Variant 13

$AF=(Q, \Sigma, \delta, q_0, F)$,
$Q = \{ q_0, q_1, q_2, q_3 \}$,
$\Sigma = \{ a, b \}$, $F = \{ q_3 \}$.
$\delta(q_0, a) = q_0$,
$\delta(q_0, b) = q_1$,
$\delta(q_1, a) = q_1$,
$\delta(q_1, a) = q_2$,
$\delta(q_1, b) = q_3$,
$\delta(q_2, a) = q_2$,
$\delta(q_2, b) = q_3$.

1. Present the automaton in form of graph. Is this automaton deterministic or no? Why?
2. Convert a Finite Automaton to the Regular Grammar.
3. Transform nondeterministic finite automaton (NFA) into a deterministic automaton (DFA). Present the DFA in form of graph.
4. Present 1 string based on the input alphabet Σ that will be not accepted by the automaton. For showing use the Multiple Run option from menu Input from JFLAP.
5. For the finite automaton $AF=(Q, \Sigma, \delta, q_0, F)$ present five strings that will be accepted. The length of strings should be more then n+2, where n is the number of stares from Q. For showing use the Multiple Run option from menu Input from JFLAP.
   Elaborate the program that for the obtained DFA will generate the strings by the given lenght.
6. For each string x, present the configuration sequence for the acceptance of this string. To show the acceptance use the option Step with Closure and Fast Run from menu Input from JFLAP.
7. For each obrtained 5 strings build the decompozition x=uvw , using the pumping lemma.
8. Obtained DFA minimize.

Theme : Finite Automata

Variant 14

$AF=(Q, \Sigma, \delta, q_0, F)$,
$Q = \{ q_0, q_1, q_2 \}$,
$\Sigma = \{ a, b, c \}$, $F = \{ q_2 \}$.
$\delta(q_0, a) = q_0$,
$\delta(q_0, b) = q_1$,
$\delta(q_1, c) = q_1$,
$\delta(q_1, c) = q_2$,
$\delta(q_2, a) = q_0$,
$\delta(q_1, a) = q_1$.

1. Present the automaton in form of graph. Is this automaton deterministic or no? Why?
2. Convert a Finite Automaton to the Regular Grammar.
3. Transform nondeterministic finite automaton (NFA) into a deterministic automaton (DFA). Present the DFA in form of graph.
4. Present 1 string based on the input alphabet Σ that will be not accepted by the automaton. For showing use the Multiple Run option from menu Input from JFLAP.
5. For the finite automaton $AF=(Q, \Sigma, \delta, q_0, F)$ present five strings that will be accepted. The length of strings should be more then n+2, where n is the number of stares from Q. For showing use the Multiple Run option from menu Input from JFLAP.
   Elaborate the program that for the obtained DFA will verify if string is accepted or no.
6. For each string x, present the configuration sequence for the acceptance of this string. To show the acceptance use the option Step with Closure and Fast Run from menu Input from JFLAP.
7. For each obrtained 5 strings build the decompozition x=uvw , using the pumping lemma.
8. Obtained DFA minimize.

Theme : Finite Automata

Variant 15

$AF=(Q, \Sigma, \delta, q_0, F)$,
$Q = \{ q_0, q_1, q_2, q_3 \}$,
$\Sigma = \{ a, b, c \}$, $F = \{ q_3 \}$.
$\delta(q_0, a) = q_0$,
$\delta(q_1, b) = q_2$,
$\delta(q_0, a) = q_1$,
$\delta(q_2, a) = q_2$,
$\delta(q_2, b) = q_3$,
$\delta(q_2, c) = q_0$.

1. Present the automaton in form of graph. Is this automaton deterministic or no? Why?
2. Convert a Finite Automaton to the Regular Grammar.
3. Transform nondeterministic finite automaton (NFA) into a deterministic automaton (DFA). Present the DFA in form of graph.
4. Present 1 string based on the input alphabet Σ that will be not accepted by the automaton. For showing use the Multiple Run option from menu Input from JFLAP.
5. For the finite automaton $AF=(Q, \Sigma, \delta, q_0, F)$ present five strings that will be accepted. The length of strings should be more then n+2, where n is the number of stares from Q. For showing use the Multiple Run option from menu Input from JFLAP.
   Elaborate the program that for the obtained DFA will generate the strings by the given lenght.
6. For each string x, present the configuration sequence for the acceptance of this string. To show the acceptance use the option Step with Closure and Fast Run from menu Input from JFLAP.
7. For each obrtained 5 strings build the decompozition x=uvw , using the pumping lemma.
8. Obtained DFA minimize.

Variant 16

Theme : Finite Automata

$AF=(Q, \Sigma, \delta, q_0, F)$,
$Q = \{ q_0, q_1, q_2, q_3 \}$,
$\Sigma = \{ a, b \}$, $F = \{ q_3 \}$.
$\delta(q_0, a) = q_1$,
$\delta(q_1, b) = q_1$,
$\delta(q_1, b) = q_2$,
$\delta(q_2, a) = q_2$,
$\delta(q_2, b) = q_3$,
$\delta(q_0, b) = q_0$.

1. Present the automaton in form of graph. Is this automaton deterministic or no? Why?
2. Convert a Finite Automaton to the Regular Grammar.
3. Transform nondeterministic finite automaton (NFA) into a deterministic automaton (DFA). Present the DFA in form of graph.
4. Present 1 string based on the input alphabet Σ that will be not accepted by the automaton. For showing use the Multiple Run option from menu Input from JFLAP.
5. For the finite automaton $AF=(Q, \Sigma, \delta, q_0, F)$ present five strings that will be accepted. The length of strings should be more then n+2, where n is the number of stares from Q. For showing use the Multiple Run option from menu Input from JFLAP.
   Elaborate the program that for the obtained DFA will verify if string is accepted or no.
6. For each string x, present the configuration sequence for the acceptance of this string. To show the acceptance use the option Step with Closure and Fast Run from menu Input from JFLAP.
7. For each obrtained 5 strings build the decompozition x=uvw , using the pumping lemma.
8. Obtained DFA minimize.

Variant 17

AF=(Q, Σ, δ, q_0, F),
Q = { q_0, q_1, q_2, q_3 },
Σ = { a, b, c}, F = { q_3 }.
δ (q_0, a) = q_0 ,
δ (q_0, a) = q_1 ,
δ (q_1, b) = q_1 ,
δ (q_2, b) = q_3,
δ (q_1, a) = q_2,
δ (q_2, a) = q_0.

Theme : Finite Automata

1. Present the automaton in form of graph. Is this automaton deterministic or no? Why?
2. Convert a Finite Automaton to the Regular Grammar.
3. Transform nondeterministic finite automaton (NFA) into a deterministic automaton (DFA). Present the DFA in form of graph.
4. Present 1 string based on the input alphabet Σ that will be not accepted by the automaton. For showing use the Multiple Run option from menu Input from JFLAP.
5. For the finite automaton AF=(Q, Σ, δ, q_0, F) present five strings that will be accepted. The length of strings should be more then n+2, where n is the number of stares from Q. For showing use the Multiple Run option from menu Input from JFLAP.
   Elaborate the program that for the obtained DFA will generate the strings by the given lenght.
6. For each string x, present the configuration sequence for the acceptance of this string. To show the acceptance use the option Step with Closure and Fast Run from menu Input from JFLAP.
7. For each obrtained 5 strings build the decompozition x=uvw , using the pumping lemma.
8. Obtained DFA minimize.

Variant 18

AF=(Q, Σ, δ, q_0, F),
Q = { q_0, q_1, q_2, q_3 },
Σ = { a, b, c }, F = { q_3}.
δ (q_0, a) = q_0 ,
δ (q_0, a) = q_1 ,
δ (q_1, b) = q_2 ,
δ (q_2, a) = q_2,
δ (q_3, a) = q_3,
δ (q_2, b) = q_3.

Theme : Finite Automata

1. Present the automaton in form of graph. Is this automaton deterministic or no? Why?
2. Convert a Finite Automaton to the Regular Grammar.
3. Transform nondeterministic finite automaton (NFA) into a deterministic automaton (DFA). Present the DFA in form of graph.
4. Present 1 string based on the input alphabet Σ that will be not accepted by the automaton. For showing use the Multiple Run option from menu Input from JFLAP.
5. For the finite automaton AF=(Q, Σ, δ, q_0, F) present five strings that will be accepted. The length of strings should be more then n+2, where n is the number of stares from Q. For showing use the Multiple Run option from menu Input from JFLAP.
   Elaborate the program that for the obtained DFA will verify if string is accepted or no.
6. For each string x, present the configuration sequence for the acceptance of this string. To show the acceptance use the option Step with Closure and Fast Run from menu Input from JFLAP.
7. For each obrtained 5 strings build the decompozition x=uvw , using the pumping lemma.
8. Obtained DFA minimize.

Variant 19

AF=(Q, Σ, δ, q_0, F),
Q = { q_0, q_1, q_2 },
Σ = { a, b }, F = { q_2}.
δ (q_0, a) = q_1 ,
δ (q_0, a) = q_0 ,
δ (q_1, b) = q_2 ,
δ (q_0, b) = q_0,
δ (q_1, b) = q_1,
δ (q_2, b) = q_2.

Theme : Finite Automata

1. Present the automaton in form of graph. Is this automaton deterministic or no? Why?
2. Convert a Finite Automaton to the Regular Grammar.
3. Transform nondeterministic finite automaton (NFA) into a deterministic automaton (DFA). Present the DFA in form of graph.
4. Present 1 string based on the input alphabet Σ that will be not accepted by the automaton. For showing use the Multiple Run option from menu Input from JFLAP.
5. For the finite automaton AF=(Q, Σ, δ, q_0, F) present five strings that will be accepted. The length of strings should be more then n+2, where n is the number of stares from Q. For showing use the Multiple Run option from menu Input from JFLAP.
   Elaborate the program that for the obtained DFA will generate the strings by the given lenght.
6. For each string x, present the configuration sequence for the acceptance of this string. To show the acceptance use the option Step with Closure and Fast Run from menu Input from JFLAP.
7. For each obrtained 5 strings build the decompozition x=uvw , using the pumping lemma.
8. Obtained DFA minimize.

Variant 20

AF=(Q, Σ, δ, q_0, F),
Q = { q_0, q_1, q_2, q_3 },
Σ = { a, b, c }, F = { q_3}.
δ (q_0, a) = q_0,
δ (q_0, a) = q_1,
δ (q_2, a) = q_2,
δ (q_1, b) = q_2,
δ (q_2, c) = q_3,
δ (q_3, c) = q_3 .

Theme : Finite Automata

1. Present the automaton in form of graph. Is this automaton deterministic or no? Why?
2. Convert a Finite Automaton to the Regular Grammar.
3. Transform nondeterministic finite automaton (NFA) into a deterministic automaton (DFA). Present the DFA in form of graph.
4. Present 1 string based on the input alphabet Σ that will be not accepted by the automaton. For showing use the Multiple Run option from menu Input from JFLAP.
5. For the finite automaton AF=(Q, Σ, δ, q_0, F) present five strings that will be accepted. The length of strings should be more then n+2, where n is the number of stares from Q. For showing use the Multiple Run option from menu Input from JFLAP.
   Elaborate the program that for the obtained DFA will verify if string is accepted or no.
6. For each string x, present the configuration sequence for the acceptance of this string. To show the acceptance use the option Step with Closure and Fast Run from menu Input from JFLAP.
7. For each obrtained 5 strings build the decompozition x=uvw , using the pumping lemma.
8. Obtained DFA minimize.

Variant 21

AF=(Q, Σ, δ, q_0, F),
Q = { q_0, q_1, q_2, q_3 },
Σ = { a, b, c }, F = { q_3}.
δ (q_0, a) = q_0 ,
δ (q_0, a) = q_1 ,
δ (q_1, b) = q_2 ,
δ (q_2, c) = q_3 ,
δ (q_3, c) = q_3,
δ (q_2, a) = q_2.

Theme : Finite Automata

1. Present the automaton in form of graph. Is this automaton deterministic or no? Why?
2. Convert a Finite Automaton to the Regular Grammar.
3. Transform nondeterministic finite automaton (NFA) into a deterministic automaton (DFA). Present the DFA in form of graph.
4. Present 1 string based on the input alphabet Σ that will be not accepted by the automaton. For showing use the Multiple Run option from menu Input from JFLAP.
5. For the finite automaton AF=(Q, Σ, δ, q_0, F) present five strings that will be accepted. The length of strings should be more then n+2, where n is the number of stares from Q. For showing use the Multiple Run option from menu Input from JFLAP. Elaborate the program that for the obtained DFA will generate the strings by the given lenght.
6. For each string x, present the configuration sequence for the acceptance of this string. To show the acceptance use the option Step with Closure and Fast Run from menu Input from JFLAP.
7. For each obrtained 5 strings build the decompozition x=uvw , using the pumping lemma.
8. Obtained DFA minimize.

Variant 22

AF=(Q, Σ, δ, q_0, F),
Q = { q_0, q_1, q_2 },
Σ = { a, b }, F = { q_2}.
δ (q_0, a) = q_0 ,
δ (q_1, b) = q_1,
δ (q_1, b) = q_2,
δ (q_0, b) = q_1 ,
δ (q_1, a) = q_0,
δ (q_2, b) = q_1.

Theme : Finite Automata

1. Present the automaton in form of graph. Is this automaton deterministic or no? Why?
2. Convert a Finite Automaton to the Regular Grammar.
3. Transform nondeterministic finite automaton (NFA) into a deterministic automaton (DFA). Present the DFA in form of graph.
4. Present 1 string based on the input alphabet Σ that will be not accepted by the automaton. For showing use the Multiple Run option from menu Input from JFLAP.
5. For the finite automaton AF=(Q, Σ, δ, q_0, F) present five strings that will be accepted. The length of strings should be more then n+2, where n is the number of stares from Q. For showing use the Multiple Run option from menu Input from JFLAP. Elaborate the program that for the obtained DFA will verify if string is accepted or no.
6. For each string x, present the configuration sequence for the acceptance of this string. To show the acceptance use the option Step with Closure and Fast Run from menu Input from JFLAP.
7. For each obrtained 5 strings build the decompozition x=uvw , using the pumping lemma.
8. Obtained DFA minimize

Variant 23

AF=(Q, Σ, δ, q_0, F),
Q = { q_0, q_1, q_2 },
Σ = { a, b }, F = { q_2}.
δ (q_0, a) = q_0 ,
δ (q_0, a) = q_1 ,
δ (q_1, b) = q_2 ,
δ (q_0, b) = q_0,
δ (q_2, b) = q_2,
δ (q_1, a) = q_0.

Theme : Finite Automata

1. Present the automaton in form of graph. Is this automaton deterministic or no? Why?
2. Convert a Finite Automaton to the Regular Grammar.
3. Transform nondeterministic finite automaton (NFA) into a deterministic automaton (DFA). Present the DFA in form of graph.
4. Present 1 string based on the input alphabet Σ that will be not accepted by the automaton. For showing use the Multiple Run option from menu Input from JFLAP.
5. For the finite automaton AF=(Q, Σ, δ, q_0, F) present five strings that will be accepted. The length of strings should be more then n+2, where n is the number of stares from Q. For showing use the Multiple Run option from menu Input from JFLAP. Elaborate the program that for the obtained DFA will generate the strings by the given lenght.
6. For each string x, present the configuration sequence for the acceptance of this string. To show the acceptance use the option Step with Closure and Fast Run from menu Input from JFLAP.
7. For each obrtained 5 strings build the decompozition x=uvw , using the pumping lemma.
8. Obtained DFA minimize

Variant 24

AF=(Q, Σ, δ, q_0, F),
Q = { q_0, q_1, q_2 },
Σ = { a, b }, F = { q_2}.
δ (q_0, b) = q_0 ,
δ (q_0, b) = q_1 ,
δ (q_1, b) = q_2 ,
δ (q_0, a) = q_0,

δ (q_2, a) = q_2,

Theme : Finite Automata

1. Present the automaton in form of graph. Is this automaton deterministic or no? Why?
2. Convert a Finite Automaton to the Regular Grammar.
3. Transform nondeterministic finite automaton (NFA) into a deterministic automaton (DFA). Present the DFA in form of graph.
4. Present 1 string based on the input alphabet Σ that will be not accepted by the automaton. For showing use the Multiple Run option from menu Input from JFLAP.
5. For the finite automaton AF=(Q, Σ, δ, q_0, F) present five strings that will be accepted. The length of strings should be more then n+2, where n is the number of stares from Q. For showing use the Multiple Run option from menu Input from JFLAP. Elaborate the program that for the obtained DFA will verify if string is accepted or no.
6. For each string x, present the configuration sequence for the acceptance of this string. To show the acceptance use the option Step with Closure and Fast Run from menu Input from JFLAP.
7. For each obrtained 5 strings build the decompozition x=uvw , using the pumping lemma.
8. Obtained DFA minimize

δ (q_1, a) = q_1.

Variant 25

AF=$(Q, \Sigma, \delta, q_0, F)$,
$Q = \{ q_0, q_1, q_2, q_3 \}$,
$\Sigma = \{ a, b \}$, $F = \{ q_2\}$.
$\delta (q_0, a) = q_0$,
$\delta (q_0, a) = q_1$,
$\delta (q_1, a) = q_2$,
$\delta (q_1, b) = q_1$,
$\delta (q_2, a) = q_3$,
$\delta (q_3, a) = q_1$.

Theme : Finite Automata

1. Present the automaton in form of graph. Is this automaton deterministic or no? Why?
2. Convert a Finite Automaton to the Regular Grammar.
3. Transform nondeterministic finite automaton (NFA) into a deterministic automaton (DFA). Present the DFA in form of graph.
4. Present 1 string based on the input alphabet Σ that will be not accepted by the automaton. For showing use the Multiple Run option from menu Input from JFLAP.
5. For the finite automaton AF=(Q, Σ, δ, q_0, F) present five strings that will be accepted. The length of strings should be more then n+2, where n is the number of stares from Q. For showing use the Multiple Run option from menu Input from JFLAP.
   Elaborate the program that for the obtained DFA will generate the strings by the given lenght.
6. For each string x, present the configuration sequence for the acceptance of this string. To show the acceptance use the option Step with Closure and Fast Run from menu Input from JFLAP.
7. For each obrtained 5 strings build the decompozition x=uvw , using the pumping lemma.
8. Obtained DFA minimize

Variant 26

AF=$(Q, \Sigma, \delta, q_0, F)$,
$Q = \{ q_0, q_1, q_2 , q_3\}$,
$\Sigma = \{ a, b, c \}$, $F = \{ q_3\}$.
$\delta (q_0, a) = q_1$,
$\delta (q_1, b) = q_1$,
$\delta (q_1, a) = q_2$,
$\delta (q_0, a) = q_0$,
$\delta (q_2, c) = q_3$,
$\delta (q_3, c) = q_3$

Theme : Finite Automata

1. Present the automaton in form of graph. Is this automaton deterministic or no? Why?
2. Convert a Finite Automaton to the Regular Grammar.
3. Transform nondeterministic finite automaton (NFA) into a deterministic automaton (DFA). Present the DFA in form of graph.
4. Present 1 string based on the input alphabet Σ that will be not accepted by the automaton. For showing use the Multiple Run option from menu Input from JFLAP.
5. For the finite automaton AF=(Q, Σ, δ, q_0, F) present five strings that will be accepted. The length of strings should be more then n+2, where n is the number of stares from Q. For showing use the Multiple Run option from menu Input from JFLAP.
   Elaborate the program that for the obtained DFA will verify if string is accepted or no.
6. For each string x, present the configuration sequence for the acceptance of this string. To show the acceptance use the option Step with Closure and Fast Run from menu Input from JFLAP.
7. For each obrtained 5 strings build the decompozition x=uvw , using the pumping lemma.
8. Obtained DFA minimize

Variant 27

AF=$(Q, \Sigma, \delta, q_0, F)$,
$Q = \{ q_0, q_1, q_2 , q_3\}$,
$\Sigma = \{ a, b \}$, $F = \{ q_3\}$.
$\delta (q_0, a) = q_1$,
$\delta (q_1, b) = q_2$,
$\delta (q_1, a) = q_3$,
$\delta (q_0, b) = q_2$,
$\delta (q_2, b) = q_3$,
$\delta (q_1 , a) = q_1$.

Theme : Finite Automata

1. Present the automaton in form of graph. Is this automaton deterministic or no? Why?
2. Convert a Finite Automaton to the Regular Grammar.
3. Transform nondeterministic finite automaton (NFA) into a deterministic automaton (DFA). Present the DFA in form of graph.
4. Present 1 string based on the input alphabet Σ that will be not accepted by the automaton. For showing use the Multiple Run option from menu Input from JFLAP.
5. For the finite automaton AF=(Q, Σ, δ, q_0, F) present five strings that will be accepted. The length of strings should be more then n+2, where n is the number of stares from Q. For showing use the Multiple Run option from menu Input from JFLAP.
   Elaborate the program that for the obtained DFA will generate the strings by the given lenght.
6. For each string x, present the configuration sequence for the acceptance of this string. To show the acceptance use the option Step with Closure and Fast Run from menu Input from JFLAP.
7. For each obrtained 5 strings build the decompozition x=uvw , using the pumping lemma.
8. Obtained DFA minimize

Variant 28

AF=$(Q, \Sigma, \delta, q_0, F)$,
$Q = \{ q_0, q_1, q_2, q_3 \}$,
$\Sigma = \{ a, b, c \}$, $F = \{ q_3\}$.
$\delta (q_0, a) = q_0$,
$\delta (q_0, a) = q_1$,
$\delta (q_1, a) = q_1$,
$\delta (q_1, c) = q_2$,
$\delta (q_1, b) = q_3$,
$\delta (q_0, b) = q_2$,
$\delta (q_2, b) = q_3$.

Theme : Finite Automata

1. Present the automaton in form of graph. Is this automaton deterministic or no? Why?
2. Convert a Finite Automaton to the Regular Grammar.
3. Transform nondeterministic finite automaton (NFA) into a deterministic automaton (DFA). Present the DFA in form of graph.
4. Present 1 string based on the input alphabet Σ that will be not accepted by the automaton. For showing use the Multiple Run option from menu Input from JFLAP.
5. For the finite automaton AF=(Q, Σ, δ, q_0, F) present five strings that will be accepted. The length of strings should be more then n+2, where n is the number of stares from Q. For showing use the Multiple Run option from menu Input from JFLAP.
   Elaborate the program that for the obtained DFA will verify if string is accepted or no.
6. For each string x, present the configuration sequence for the acceptance of this string. To show the acceptance use the option Step with Closure and Fast Run from menu Input from JFLAP.
7. For each obrtained 5 strings build the decompozition x=uvw , using the pumping lemma.
8. Obtained DFA minimize

Variant 29

AF=(Q, $\sum$, δ, q_0, F),

Q = { q_0, q_1, q_2},

$\sum$ = { a, b }, F = { q_2}.

δ (q_0, a) = q_1 ,

δ (q_0, a) = q_0 ,

δ (q_1, b) = q_1,

δ (q_1, a) = q_2,

δ (q_2 , b) = q_2 ,

δ (q_2 , a) = q_0.

Variant 30

AF=(Q, $\sum$, δ, q_0, F),

Q = { q_0, q_1, q_2 },

$\sum$ = { a, b, c }, F = { q_2}.

δ (q_0, a) = q_1 ,

δ (q_1, c) = q_1 ,

δ (q_1, b) = q_2,

δ (q_2 , a) = q_2 ,

δ (q_0 , a) = q_0,

δ (q_1, a) = q_0.

Variant 31

AF=(Q, $\sum$, δ, q_0, F),

Q = { q_0, q_1, q_2, q_3 },

$\sum$ = { a, b }, F = { q_3}.

δ (q_0, a) = q_0 ,

δ (q_0, a) = q_1,

δ (q_1, a) = q_1,

δ (q_1, b) = q_2,

δ (q_2, b) = q_3 ,

δ (q_3, b) = q_3.

Variant 32

AF=(Q, $\sum$, δ, q_0, F),

Q = { q_0, q_1, q_2 },

$\sum$ = { a, b, c }, F = { q_2 }.

δ (q_0, a) = q_0 ,

δ (q_0, a) = q_1 ,

δ (q_1, c) = q_0,

δ (q_1, b) = q_1,

δ (q_1, a) = q_2,

δ (q_2, a) = q_2.