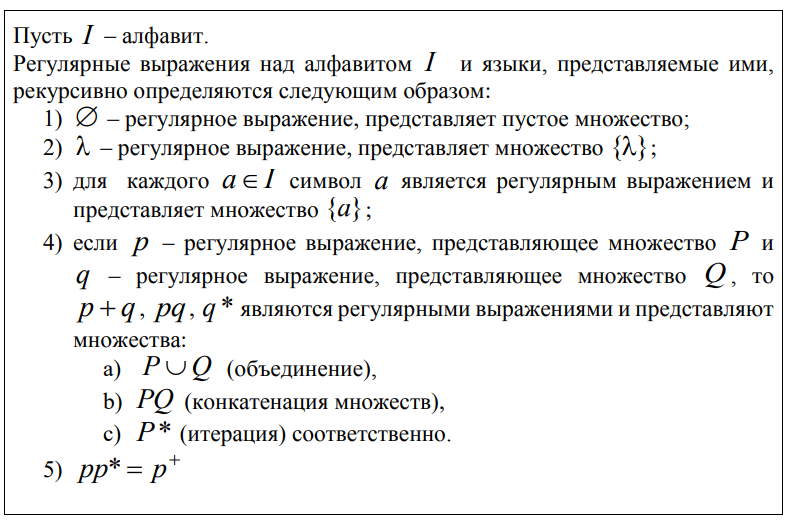
Регулярное выражение описывает множество цепочек – формальный язык. Для записи регулярного выражения используются метасимволы. Множество цепочек описанных регулярным выражением называется регулярным множеством (или регулярным языком).



Регулярное выражение: open(□)+((write+read+seek);) +□+close

1.open□write;□close

2.open□□□write;read;□□□close

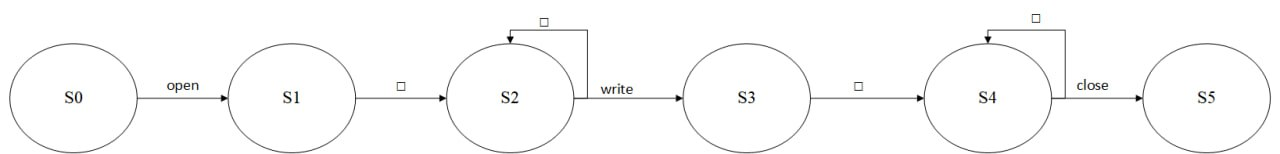
3.open□write;read;seek;□□□□close

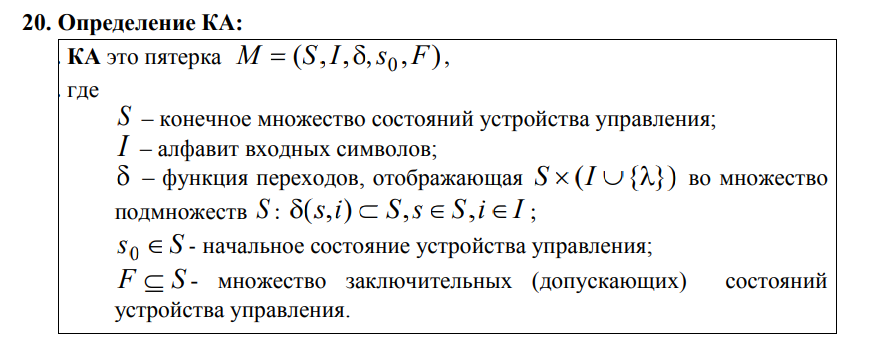
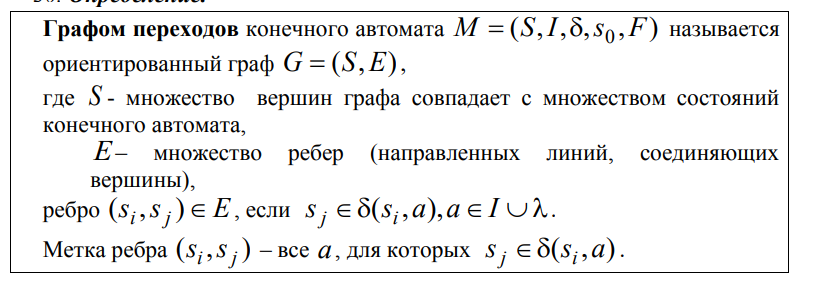
4.open□□□□□write;seek;□close 5.open□□□□write;read;seek;write;read;seek;□□□□close 6.open□write;seek;read;read;seek;□close

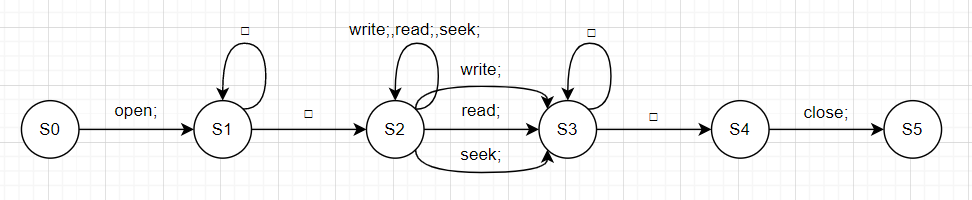
7.open□□□□write;read;seek;seek;seek;□□□□□□close

open□write;□close

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | open | □ | write | □ | close |
| S0 | S1 | ∅ | ∅ | ∅ | ∅ |
| S1 | ∅ | S2 | ∅ | ∅ | ∅ |
| S2 | ∅ | ∅ | S3 | ∅ | ∅ |
| S3 | ∅ | ∅ | ∅ | S4 | ∅ |
| S4 | ∅ | ∅ | ∅ | ∅ | S5 |
| S5 | ∅ | ∅ | ∅ | ∅ | ∅ |



open(□)+((write+read+seek);) +□+close



1. Что такое алфавит I?

o Алфавит — это конечное непустое множество допустимых символов языка (букв языка). Пример: 𝑉={𝑎,𝑏}V={a,b} — алфавит 𝑉V, состоящий из двух символов a и b.

2. Поясните обозначение λ, I, I+:\*

o λ (лямбда) обозначает пустую цепочку, которая не содержит ни одного символа.

o I\* — множество всех возможных цепочек, которые можно составить из символов алфавита 𝐼I, включая пустую цепочку.

o I+ — множество всех возможных непустых цепочек, составленных из символов алфавита 𝐼I.

3. Что такое язык L(I) над алфавитом I?

o Язык L(I) над алфавитом 𝐼I называется произвольное подмножество цепочек из 𝐼∗I∗ (множества всех возможных цепочек из символов алфавита 𝐼I).

4. Дайте определение формальной грамматике G:

o Формальная грамматика — это способ задания языка, который включает множество правил для порождения цепочек символов этого языка. Формальная грамматика обычно записывается как 𝐺=(𝑁,𝑇,𝑃,𝑆)G=(N,T,P,S), где:

 N — конечное множество нетерминальных символов;

 T — конечное множество терминальных символов;

 P — конечное множество правил вывода;

 S — стартовый символ, из которого начинаются выводы.

5. Поясните обозначение α ⇒ β и α ⇒ β:\*

o α ⇒ β — цепочка 𝛽β непосредственно выводима из цепочки 𝛼α по одному правилу грамматики.

o α ⇒ β\* — цепочка 𝛽β выводима из цепочки 𝛼α за несколько (возможно, ноль) шагов по правилам грамматики.

6. Что такое язык L(G), порождаемый грамматикой G?

o Язык L(G) — это множество всех терминальных цепочек, которые можно вывести из стартового символа 𝑆S грамматики 𝐺G с использованием правил этой грамматики. Формально это записывается как 𝐿(𝐺)={𝛼∈𝑇∗∣𝑆⇒∗𝛼}L(G)={α∈T∗∣S⇒∗α}, где 𝑇T — множество терминалов.

7. Что такое форма Бэкуса-Наура?

o Форма Бэкуса-Наура (BNF) — это нотация для формального описания синтаксиса языков программирования. Она используется для задания правил грамматики в виде производственных правил, которые описывают, как строки языка могут быть порождены из начального символа.

8. Что такое регулярная грамматика?

o Регулярная грамматика — это грамматика, в правилах вывода которой с правой стороны находится либо одиночный терминал, либо терминал, за которым следует нетерминал. Примеры правил регулярной грамматики: 𝐴→𝑎𝐵A→aB, 𝐵→𝑏B→b, 𝐶→𝜀C→ε.

9. Поясните понятие "регулярное множество":

o Регулярное множество — это множество, которое может быть описано регулярным выражением или распознано конечным автоматом. Это подмножество цепочек, которые удовлетворяют регулярному выражению.

10. Поясните понятие "регулярный язык":

o Регулярный язык — это язык, который может быть описан регулярной грамматикой, регулярным выражением или распознан конечным автоматом. Все регулярные языки являются регулярными множествами.

11. Что такое лексический анализ?

o Лексический анализ — это процесс разбора входного потока символов на токены (лексемы), которые являются смысловыми единицами языка программирования. Этот процесс выполняется лексическим анализатором.

12. Что такое лексический анализатор?

o Лексический анализатор (сканер) — это компонент компилятора, который выполняет лексический анализ, то есть преобразует входной поток символов в последовательность токенов.

13. Какая информация является входной для лексического анализатора, какая — выходной?

o Входная информация: исходный код программы в виде потока символов.

o Выходная информация: последовательность токенов, где каждый токен представляет собой отдельную смысловую единицу программы (ключевые слова, идентификаторы, операторы и т.д.).

14. Поясните различия между последовательным и параллельным лексическими анализаторами:

o Последовательный лексический анализатор обрабатывает входной поток символов линейно, один за другим.

o Параллельный лексический анализатор может обрабатывать несколько потоков символов одновременно, что может увеличить производительность, особенно в многопроцессорных системах.

15. Дайте определение регулярного выражения над алфавитом I:

o Регулярное выражение — это формальное средство для описания множества строк в алфавите 𝐼I. Регулярные выражения состоят из символов алфавита и специальных символов (метасимволов), которые обозначают операции объединения, конкатенации и замыкания (звезда Клини).

16. Дайте определение конечному автомату M = (S, I, δ, s0, F):

o Конечный автомат — это математическая модель, состоящая из:

 𝑆S — конечного множества состояний,

 𝐼I — конечного множества входных символов (алфавит),

 𝛿δ — функции переходов, 𝛿:𝑆×𝐼→𝑆δ:S×I→S,

 𝑠0s0 — начального состояния, 𝑠0∈𝑆s0∈S,

 𝐹F — множества конечных состояний, 𝐹⊆𝑆F⊆S.

17. В чем отличие между детерминированным и недетерминированным автоматом?

o Детерминированный конечный автомат (DFA): в каждом состоянии для каждого входного символа определен только один переход.

o Недетерминированный конечный автомат (NFA): в каждом состоянии для каждого входного символа может быть несколько возможных переходов, включая переходы без потребления символа (ε-переходы).

18. Определите понятие: мгновенное состояние конечного автомата:

o Мгновенное состояние конечного автомата — это пара (s, w), где 𝑠s — текущее состояние автомата, а 𝑤w — оставшаяся часть входной строки, которая еще не обработана.

19. Поясните обозначения (s, aw) ⟹ (s', w) и (sj, wi) ⟹ (sk, wk):\*

o (s, aw) ⟹ (s', w) — автомат переходит из состояния 𝑠s в состояние 𝑠′s′ при чтении символа 𝑎a из входной строки 𝑤w.

o (sj, wi) ⟹ (sk, wk)\* — автомат за несколько шагов переходит из состояния 𝑠𝑗sj в состояние 𝑠𝑘sk, обрабатывая строку 𝑤𝑖wi, и оставляя остаток строки 𝑤𝑘wk.

20. В каком соотношении находятся регулярная грамматика, регулярный язык, регулярное выражение, конечный автомат, граф состояний конечного автомата?

o Регулярная грамматика, регулярный язык, регулярное выражение и конечный автомат описывают один и тот же класс языков — регулярные языки. Регулярные выражения и регулярные грамматики могут быть преобразованы в конечные автоматы, которые могут быть представлены в виде графов состояний, описывающих их работу. Эти концепции взаимосвязаны и используются для описания и анализа регулярных языков.