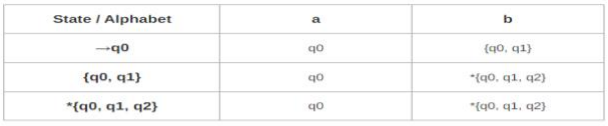
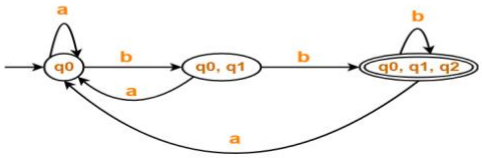
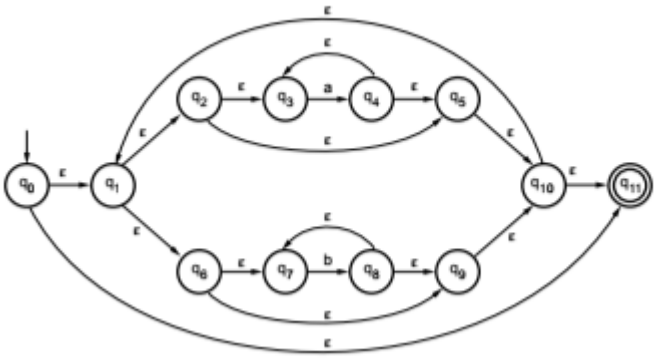
SRM Institute of Science and Technology
College of Engineering and Technology
SCHOOL OF COMPUTING- ANSWER KEY

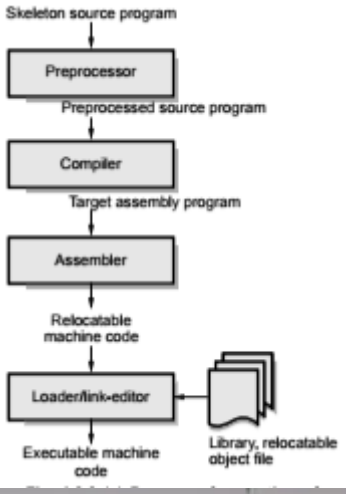
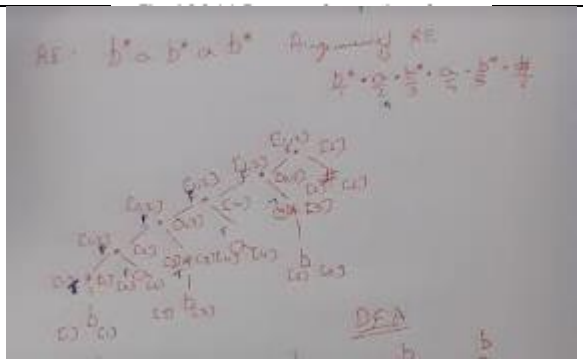
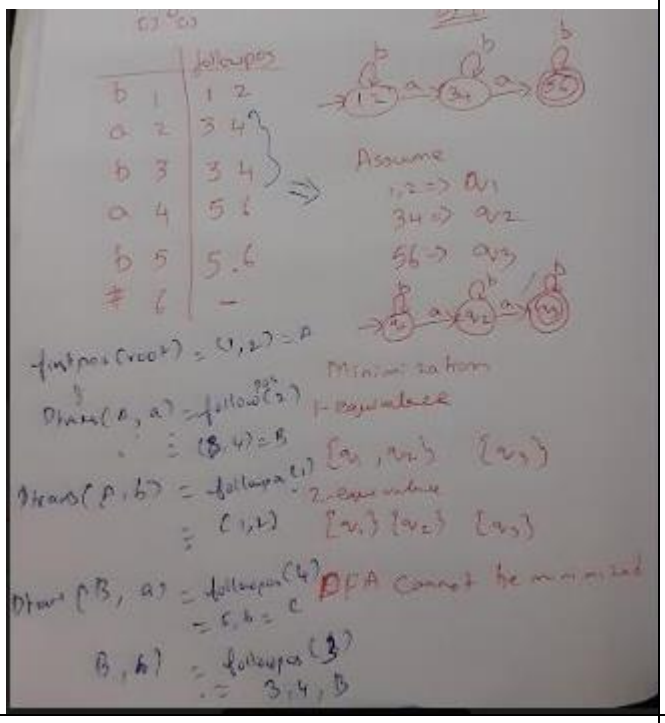
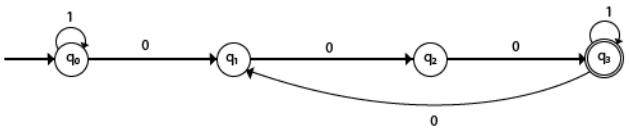
SRM Nagar, Kattankulathur – 603203, Chengalpattu District, Tamilnadu

Mode of Exam
OFFLINE

Academic Year: 2022-23 (EVEN) SET B – ANSWER KEY

Test: CLAT-1 **Date: 17.2.2022**
Course Code & Title: 18CSC304J COMPILER DESIGN **Duration: 1 HOUR**
Year & Sem: III & V **Max. Marks: 25**

Q. No	Question	Marks
1	b) {0,1,3,7} and {3}	1
2	d) %%	1
3	c.) 26	1
4	**d) 12**	1
5	**a)aaabbbab**	1
6	 Now, Deterministic Finite Automata (DFA) may be drawn as- 	4
7	The NFA with ε for $(a*/b*)*$ will be - 	4

8		**4**
11	 	**10**
12	i) DFA with three consecutive zero's  ii) Various operations of regular lanaguages Let L, L_1, L_2 be subsets of Σ^* Concatenation: $L_1L_2 = \{xy \mid x \text{ is in } L_1 \text{ and } y \text{ is in } L_2\}$	**10**

Union is set union of L1 and L2

Kleene Closure: $L^* = L^i = L^0 \cup L^1 \cup L^2 \cup \ldots$

Positive Closure: $L^+ = L^i = L^1 \cup L^2 \cup \ldots$

iii)

Front end back end

Different phases of compiler can be grouped together to form a **front end** and **back end**. The front end consists of those phases that primarily dependent on the source language and independent on the target language. The front end consists of analysis part. Typically it includes lexical analysis, syntax analysis, and semantic analysis. Some

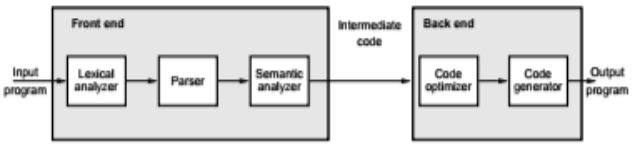


Fig. 1.7.1 Front end back end model

amount of code optimization can also be done at front end. The back end consists of those phases that are totally dependent upon the target language and independent on the source language. It includes code generation and code optimization. The front end back end model of the compiler is very much advantageous because of following reasons -

1. By **keeping the same front end** and attaching **different back ends** one can produce **a compiler** for same source language **on different machines**.
2. By keeping **different front ends** and **same back end** one can compile **several different languages** on the same machine.

Machine Dependent and Machine Independent Phases

The machine dependent phases are code generation and code optimization phases. The machine independent phases are lexical analyzers, syntax analyzers, semantic analyzers.