Code: 20A05601T

R20

B.Tech III Year II Semester (R20) Regular Examinations August 2023

COMPILER DESIGN

(Computer Science & Engineering)

Time: 3 hours Max. Marks: 70

PART – A

(Compulsory Question)

*****

1 Answer the following: (10 X 02 = 20 Marks)

(a) What is the role of an input buffer in Lexical analyser? 2M

(b) What is bootstrapping? Give an example. 2M

(c) What is Context free grammar? Give an example. 2M

(d) State the need for Parser generator? 2M

(e) List any four three address code statements. 2M

(f) Recall the applications of Syntax directed translations. 2M

(g) What is register allocation in code generation phase? 2M

(h) List the issues in code generation phase. 2M

(i) What is a flow graph? 2M

(j) What is Constant Propagation? 2M

PART – B

(Answer all the questions: 05 X 10 = 50 Marks)

2 (a) Show the design process of a Lexical analyser generator. 5M

(b) Explain how to convert a given Regular expression to equivalent Automata? 5M

OR

3 (a) What are the phases of a compiler? Explain them? 5M

(b) Exhibit working of Optimization of DFA-Based pattern matchers. 5M

4 (a) Show the working of Bottom-Up Parser. 5M

(b) Explain about parsing ambiguous grammars by an SLR parser. 5M

OR

5 Construct SLR parsing for the following grammar: 10M

$\mathrm{E} \rightarrow \mathrm{E} + \mathrm{T}/\mathrm{T}$

$\mathrm{T} \rightarrow \mathrm{T} * \mathrm{F}/\mathrm{F}$

$\mathrm{F} \rightarrow \mathrm{id}/(\mathrm{E})$

Show the moves of parser for the input $\mathrm{id} + \mathrm{id} * \mathrm{id}$.

6 Explain how to implement L attributed SDD's with an example. 10M

OR

7 Why do we need Type checking? Give a procedure to implement the same. 10M

8 (a) Explain the procedure to access Non local data on the Stack? 5M

(b) Explain the functioning of a simple code generator. 5M

OR

9 (a) Explain the issues in the design of Code generator. 5M

(b) Write short notes on garbage collection. 5M

10 What is direct acyclic graph? Explain how this is useful for dataflow analysis. 10M

OR

11 Explain how to perform Partial redundancy elimination task? Write about flow graphs applications. 10M

*****

Code: 20A05601T

R20

B.Tech III Year II Semester (R20) Supplementary Examinations January 2024

COMPILER DESIGN

(Information Technology)

Time: 3 hours Max. Marks: 70

PART – A

(Compulsory Question)

*****

1		Answer the following: (10 X 02 = 20 Marks)	
	(a)	Define linker and loader.	2M
	(b)	List the two types of assemblers.	2M
	(c)	Compare SLR, CLR and LACR.	2M
	(d)	Define coercion.	2M
	(e)	Discuss the types of Intermediate Code.	2M
	(f)	Give the format of symbol table.	2M
	(g)	What is a basic block? Give an example.	2M
	(h)	How do you define common sub expression elimination?	2M
	(i)	What is the use of algebraic identities in optimization of basic blocks?	2M
	(j)	What is partial redundancy elimination?	2M

PART – B

(Answer all the questions: 05 X 10 = 50 Marks)

2	Explain various phases in the construction of compiler with a neat sketch.	10M
	OR	
3	Discuss compiler construction tools.	10M
4	Eliminate left recursion in the following: E->E+T|T, T->T*F|F, F->(E)|id.	10M
	OR	
5	Explain the concept of LR parsing algorithm with neat diagram.	10M
6	Explain translation schema for array elements.	10M
	OR	
7	Convert the following expression to Reverse Polish: (1 + 2) * (3 / 4) ^ (5 + 6) Show the stack contents at each step.	10M
8	Explain the concept of static VS dynamic storage allocation.	10M
	OR	
9	Define reference counting. What is the role of reference counting in garbage collection?	10M
10	Give the directed acyclic Graph Representation of Basic Blocks.	10M
	OR	
11	Explain constant propagation with example.	10M

*****