INSTITUTE OF AERONAUTICAL ENGINEERING
(AUTONOMOUS)
Dundigal, Hyderabad - 500 043

INFORMATION TECHNOLOGY
ASSIGNMENT QUESTIONS

Course Name	AUTOMATA AND COMPILER DESIGN
Course Code	A50513
Class	III B. Tech I Semester
Branch	Information Technology
Year	2017-2018
Course Coordinator	Mr. D Rahul, Assistant Professor
Course Faculty	Mr. D Rahul, Assistant Professor

OBJECTIVES

To meet the challenge of ensuring excellence in engineering education, the issue of quality needs to be addressed, debated and taken forward in a systematic manner. Accreditation is the principal means of quality assurance in higher education. The major emphasis of accreditation process is to measure the outcomes of the program that is being accredited.

In line with this, Faculty of Institute of Aeronautical Engineering, Hyderabad has taken a lead in incorporating philosophy of outcome based education in the process of problem solving and career development. So, all students of the institute should understand the depth and approach of course to be taught through this question bank, which will enhance learner's learning process.

ASSIGNMENT-I

S. No	Question	Blooms Taxonomy Level	Course Outcome
	UNIT – I		
1	**Convert** 01*+1 regular expression to finite automata?	Remember	1
2	**Explain** phases of a compiler. Also write down the output for the following	Understand	1
3	**Explain** the general format of a LEX program with example?	Knowledge	1
4	**Define** regular expression? State the rules, which define regular	Knowledge	1
5	**Explain** the role lexical analyzer and issues of lexical analyzer?	Understand	1
6	**Explain** the specification of tokens?	Knowledge	1
7	**Define** Symbol table?	Knowledge	1
8	**Explain** lexeme? Define a regular set?	Remember	1
9	**Explain** the differences between pass and phase in detail and explain Boot strapping?	Understand	1
10	**Consider** the grammar S□0A|1B|0|1 A□0S|1B|1 B □ 0A|1S **Construct** left most derivations for parse trees for the sentence.	Knowledge	1

S. No	Question	Blooms Taxonomy Level	Course Outcome
11	**Write** FIRST & FOLLOW, construct predictive parsing table for the following grammar E → T E' E' →+TE' / Ɛ T → FT ' T ' → *FT ' /Ɛ F→ (E) / id	Knowledge	1
12	**Check** the following grammar is LL(1) or not and construct parsing table. S->AaAb/BbBa	Knowledge	1
13	**Explain** elimination of left recursion in the grammar E->E+T/T T->T*F/F F->(E) /id	Knowledge	1
14	**Explain** top down parsing methods with example?	Understand	1
15	**Analyze** whether the following grammar is LL(1) or not. Explain your answer with reasons. S → L, R S → R L → * R L → id R → L	Knowledge	1
16	**For** the operators given below, calculate the operator-precedence relations and operator precedence function. id, +, *, $	Knowledge	1
17	**Check** whether the following grammar is a LL(1)grammar S→iEtS|iEtSeS|a E →b Also define the FIRST and FOLLOW procedures.	Knowledge	1
18	**Define** the necessary conditions to be carried out before the construction of predictive parser?	Remember	1

19	**Prepare** the predictive parser for the following grammar:
	
	S → a|b|(T)
	T → T, S|S
	Write down the necessary algorithms and define FIRST and FOLLOW. Show the behavior of the parser in the sentences,
	   i. (a,(a,a))
	   ii. (((a,a),a,(a),a)
		Knowledge	1

20	Consider the following fragment of C
	  code: float i, j;
	  i = i*70+j+2;
	Write the output at all phases of the compiler for above „C" code.
		Knowledge	1

UNIT – II

S. No	Question	Blooms Taxonomy Level	Course Outcome
1	**Construct** SLR parsing table for S →CC C → aC/b	Knowledge	2

S. No	Question	Blooms Taxonomy Level	Course Outcome
2	**Construct** SLR parsing table for S→ CC C →aC / b	Knowledge	2
3	**Explain** Bottom up parsing method	Understand	2
4	Illustrate Brute-Force technique through a suitable example?	Knowledge	2
5	**Explain** the error recovery in parsing.	Understand	2
6	**State** shift-reduce parsing? Explain in detail the conflicts that may occur	Understand	2
7	**Prepare** a canonical parsing table for the grammar given below S →CC C →cC|d	Knowledge	2
8	For the grammar given below, **calculate** the operator precedence relation and the precedence functions	Understand	2
9	**Consider** the grammar given below. E→ E + T E →T T→ T * F T →F F→ (E) F→ id Prepare LR parsing table for the above grammar .Give the moves of LR parser on id * id + id.	Knowledge	2
10	**Analyze** whether the following grammar is SLR(1) or not. Explain your answer with reasons. S → L,R S → R L → * R L → id R → L	Knowledge	2

UNIT – III

S. No	Question	Blooms Taxonomy Level	Course Outcome
1	**State** L – attributed grammars and S- attributed grammars with an example?	Knowledge	3
2	**Define** triple, Indirect triple, quadruples with examples?	Remember	3
3	**Explain** Intermediate code representations?	Understand	3
4	**Brief** about Syntax Directed Translator?	Knowledge	3
5	**Explain** Abstract syntax trees with an example?	Understand	3
6	**Define** type expression? Explain the equivalence of type expressions with an appropriate example?	Knowledge	3

7	**Generate** the three-address code for the following C program fragment while(a > b)
	{
	if (c < d)
	x = y +
	z; else
	x = y - z;
	}
		Understand	3

8	**Explain** Intermediate code generation for Basic block, Control Flow and Boolean Expressions?	Knowledge	3

S. No	Question	Blooms Taxonomy Level	Course Outcome
9	**Explain** how declaration is done in a procedure using syntax directed translation?	Knowledge	3
10	**List** the various ways of calling the procedures? Explain in detail?	Knowledge	3
11	**Explain** type expression, type system, simple type checker?	Understand	3
12	**List** different data structures used for symbol table?	Remember	3
13	**State** general activation record?	Understand	3
14	**Explain** type checking for different expressions?	Understand	3
15	a. **Explain** static and stack storage allocations? b. **Explain** the limitations of static allocation?	Understand	3
16	**Write** short notes on the specification of a simple type checker?	Understand	3
17	a. **Compare** three different storage allocation strategies? b. **Explain** symbol table organization using hashing?	Understand	3
18	a. **List** the various attributes of a symbol table? b. **Explain** symbol table organization using trees?	Understand	3
19	**Describe** various forms of target programs?	Remember	3
20	**Explain** heap storage allocation and static storage allocation?	Understand	3

UNIT – IV

S. No	Question	Blooms Taxonomy Level	Course Outcome
1	**Describe** 3 areas of code optimization?	Understand	4
2	**Define** constant folding?	Understand	4
3	**List** the advantages of the organization of code optimizer?	Understand	4
4	**Explain** Local optimization and loop optimization in detail.	Understand	4
5	**Define** Reduction in strength?	Understand	4
6	**Define** Common Sub expressions?	Understand	4
7	**Explain** runtime memory divisions?	Understand	4
8	**Explain** peephole optimization?	Understand	4
9	**Explain** in the DAG representation of the basic block with example.	Understand	4
10	a. **Explain** copy propagation and Dead code elimination? b. **What** is live variable?	Remember	4
11	a. **Explain** local and global common sub expression elimination? b. **Define** a flow graph. Explain how flow graph can be constructed for a given program?	Remember	4
12	a. **Explain** code hoisting and elimination of loop invariant statements? b. **Explain** how? Redundant sub expression elimination? can be done at global level in a given program?	Understand	4
13	a. **Describe** local optimization? b. **Explain** any three principal sources of code optimization?	Understand	4
14	a. **Explain** strength reduction and code movement? b. **Define** basic block? Write an algorithm for partitioning into blocks?	Understand	4
15	a. **Describe** peephole optimizations? b. **Explain** about loops in flow graphs?	Understand	4

16	a. **Explain** loop optimizations? b. **Describe** elimination of common sub expression and elimination of dead Code?	Understand	4
17	a. **Explain** natural loops and inner loops of a flow graph with an example. b. **State** purpose of data flow analysis? Explain available expression and reaching definition?	Understand	4
18	a.**Explain** strength reduction and code movement? b.**Define** basic block? Write an algorithm for partitioning into blocks?	Understand	4
19	a. **Describe** peephole optimizations? b. **Explain** about loops in flow graphs?	Understand	4
20	**Explain** in detail the optimization technique "Strength Reduction"?	Understand	4

UNIT – V

S. No	Question	Blooms Taxonomy Level	Course Outcome
1	**Explain** register allocation and assignment?	Knowledge	5
2	**Show** the code sequence generated by the simple code generation algorithm u := a − c v := t + u d := v + u//d live at the end	Understand	5
3	**Explain** object code forms, generic code algorithm?	Understand	5
4	**Explain** machine dependent and machine independent optimization?	Understand	5
5	**List** different data flow properties? Define get reg( ) function?	Knowledge	5
6	**Explain** about code generation?	Understand	5
7	**List** various machine dependent code optimization techniques?	Understand	5
8	**Explain** the different issues in the design of a code generator?	Understand	5
9	a. **Describe various** register allocation optimization techniques with an example. b. **generate** code sequence for the following expression using code generation algorithm K :=(a-b) + (a-c) + (a-c)	Knowledge	5
10	a. **Explain** about directed acyclic graph (DAG) for register allocation? b. **Discuss** various forms of object code?	Knowledge	5

Prepared by: Mr. D Rahul, Assistant Professor

Date: 07/07/2017   **HOD, IT**