

Bangladesh Army University of Engineering and Technology (BAUET)
Department of Computer Science and Engineering (CSE)
(CSE9th Batch)
Compiler Assignment List

Sl .	Student ID	Assignment Question
1	19104001	1. Find out First and Follow set from the following grammar i. $X \rightarrow XSb / Sa / b$ $S \rightarrow Sb / Xa / a$ ii. $S \rightarrow Aa / b$ $A \rightarrow Ac / Sd / \epsilon$ iii. $+ \rightarrow +BC/+fhjhesk/+jckfhgiu/thgiurshg$ $C \rightarrow +nkdsngk/Cdiovjdioj/unhguid$
2	19104003	1. Give 5 example for each optimization technique. a. Dead code elimination b. Variable elimination c. Code motion d. Reduction in strength
3	19104004	1. Write a code to find out X is a perfect number or not. For that code find out leader, design basic block and control flow graph. Here X = your id.
4	19104005	1. Consider the following program code: Prod=0; I=1; Do{ Prod=prod+a[i]*b[i]; I=i+1; }while (i<=10); a. Partition in into blocks b. Construct the flow graph 2. What is code optimization? Explain machine dependent and independent code optimization.
5	19104006	1. Write a short note with example to optimize the code: a. Dead code elimination b. Variable elimination c. Code motion d. Reduction in strength 2. What is control and data flow analysis? Explain with example.
6	19104007	1. Develop a DAG and three address code for the expression. $x = ((a + b) / (b - c)) - (a + b) * (b - c) + f.$ 2. Create DAG and three – address code for the following C program. i = 1; s = 0; while (i <= 10) { s = s + a[i] [i]; i = i + 1; }
7	19104008	1. Write a code to find out X is a perfect number or not. For that code design a symbol table and write down token. Here X = your id.
8	19104010	1. Write a code to find out X is a perfect number or not. For that code design three address code, triples. Here X = your id.
9	19104011	1. For the statement given, write three address statements and construct DAG. a. $a + a * (b - c) + (b - c) * d + g(g + h)$ b. $s\{d + g\} / g + f(t + g) + g_g + g * h$
10	19104012	1. Write a code to find out X is a perfect number or not. For that code design a symbol table and write down token. Here X = your id.
11	19104013	1. Show the following grammar is LR(1) but not LALR(1).

		$S \rightarrow Aa bAc Bc bBa$ $A \rightarrow d$ $B \rightarrow d$
		2. Write the comparison among SLR Parser, LALR parser and Canonical LR Parser
12	19104014	1. When does dangling references occur 2. Mention the two rules for type checking. 3. Mention and explain the applications of DAGs. List the advantages of DAG.
13	19104015	1. Write a code to find out X is an Armstrong number or not. For that code find out leader, design basic block and control flow graph. Here X = your id.
14	19104016	1. What are the various phases of a compiler? Explain each phase in detail by using the input "a=(b+c)*(b+c)*2"
15	19104018	1. Write the algorithm for FIRST and FOLLOW in parser. 2. What is dangling reference? Explain.
16	19104019	1. Solve the following grammar is ambiguous: $S \rightarrow aSbS / bSaS / \epsilon$ 2. Explain Top- Down parsing and Bottom up Parsing.
17	19104020	1. Eliminate Left recursion in the following grammar <div style="margin-left: 40px;"> i. $X \rightarrow XSb / Sa / b$ $S \rightarrow Sb / Xa / a$ </div> <div style="margin-left: 40px;"> ii. $S \rightarrow Aa / b$ $A \rightarrow Ac / Sd / \epsilon$ </div> <div style="margin-left: 40px;"> iii. $+ \rightarrow +BC/+fhjhesk/+jckfhgiu/thgiurshg$ $C \rightarrow +nkdsngk/Cdiovjdioj/unhguid$ </div>
18	19104022	1. Write a code to find out X is a perfect number or not. For that code design a symbol table and write down token. Here X = your id.
19	19104023	1. Write about the Error handling in different phases. (OR) Explain various Error encountered in different phases of compiler. 2. Explain language processing system with neat diagram.
20	19104024	1. Eliminate Left recursion in the following grammar <div style="margin-left: 40px;"> i. $K \rightarrow kldgdg/Kd/()ghK/K=&/c$ $G \rightarrow Ghjl/Kjjhsj/Gjdvb/llk$ </div> <div style="margin-left: 40px;"> ii. $D \rightarrow CA/H$ $C \rightarrow Ca / sd / \epsilon$ </div> <div style="margin-left: 40px;"> iii. $\& \rightarrow jhfdhjk/jsfha/elirhe/eorhjif$ $Kha \rightarrow fehnnf / jfbaj /)(*)$ </div>
21	19104025	1. Do Left factoring in the following grammar <div style="margin-left: 40px;"> i. $S \rightarrow aAd / aB$ $A \rightarrow a / ab$ $B \rightarrow ccd / ddc$ </div> <div style="margin-left: 40px;"> ii. $S \rightarrow SaS'$ $S' \rightarrow b / bc / bcd / \epsilon$ </div> <div style="margin-left: 40px;"> iii. $S \rightarrow aSSbS / aSaSb / abb / assssssbghr / asasdff / asabdgf / b$ </div>
22	19104026	1. Do Left factoring in the following grammar <div style="margin-left: 40px;"> i. $S \rightarrow S aAd / aB$ $A \rightarrow A+ a / +abb / +(),: / +scd / +scdhdfggfhd / +scd()nv / A+scd$ $B \rightarrow ccd / ddc$ </div> <div style="margin-left: 40px;"> ii. $S \rightarrow SaS'/adfe/adf/adfgre/adfeg/adhtej$ </div>

		$S' \rightarrow bcfe / bc / bcd / \epsilon$ iii. $S \rightarrow vdeS' / ++kfjn / +=jdnfd / += vde$
23	19104027	1. Explain the various Compiler Construction Tools. 2. Differentiate NFA and DFA with proper example.
24	19104028	1. Draw the syntax tree and DAG and also implement value number method for the following expression: a. $(a*b)+(c+d)-(a+b+c+d)$ b. $(a+(b*c)^d-e)/(f+g)$ c. $(a*b)+(c-d)*(a*b)+b$
25	19104029	1. Eliminate Left recursion in the following grammar i. $* \rightarrow * +) \mid)$ $) \rightarrow) * \{ \mid \{$ $\{ \rightarrow (*) \mid \}$ ii. $temp \rightarrow std$ $ter \rightarrow ter , d / ter \% e / ; ? / ac$ $L \rightarrow L B c / f / : > ?$ iii. $= \rightarrow B a / = , a / c$ $B \rightarrow B b / A '' b / d$
26	19104030	1. Write short notes a) Application of compiler technology b) Specification of Tokens c) Define LL(1) d) Differences between SLR, CLR, LALR parsers? e) Problems in Top Down Parsing?
27	19104031	1. Write three address quadruples, triples and indirect triples for the expression: d. $(a*b)+(c+d)-(a+b+c+d)$ e. $(a+(b*c)^d-e)/(f+g)$ f. $(a*b)+(c-d)*(a*b)+b$
28	19104032	1. Find out First and Follow list from the following grammar. i. $S \rightarrow S a A d / a B$ $A \rightarrow A + a / + a b b / + () , : / + s c d / + s c d h d f g g f h d / + s c d (n v / A + s c d$ $B \rightarrow c c d / d d c$ ii. $* \rightarrow * +) \mid)$ $) \rightarrow) * \{ \mid \{$ $\{ \rightarrow (*) \mid \}$ iii. $X \rightarrow X S b / S a / b$ $S \rightarrow S b / X a / a$
29	19104033	1. Write a code to find out X is an Armstrong number or not. For that code design a symbol table and write down token. Here X = your id.
30	19104034	1. Write a code to find out X is a prime number or not. For that code design each phase of compiler. Here X = your id.
31	19104035	1. Write a code to find out X is an Armstrong number or not. For that code design each phase of compiler. Here X = your id.
32	19104037	1. Find out First and Follow list from the following grammar. i. $S \rightarrow a B D h$ $B \rightarrow c C$ $C \rightarrow b C / \epsilon$ $D \rightarrow E F$ $E \rightarrow g / \epsilon$ $F \rightarrow f / \epsilon$ ii. $S \rightarrow A$ $A \rightarrow a B / A d$ $B \rightarrow b$

		$e = d \times c$ $b = e + a$ $f = b + c$ $g = f + d + b$
39	18204014	<p>1. Draw DAG for the following basic block and also implement Value number method</p> <p>i. $(a + b) \times (a + b + c)$</p> <p>ii. $((a + a) + (a + a)) + ((a + a) + (a + a))$</p> <p>iii. B10:</p> $S1 = 4 \times I$ $S2 = \text{addr}(A) - 4$ $S3 = S2[S1]$ $S4 = 4 \times I$ $S5 = \text{addr}(B) - 4$ $S6 = S5[S4]$ $S7 = S3 \times S6$ $S8 = \text{PROD} + S7$ $\text{PROD} = S8$ $S9 = I + 1$ $I = S9$ $\text{If } I \leq 20 \text{ goto L10}$