KIIT Deemed to be University
Online End Semester Examination(Autumn Semester-2020)

Subject Name & Code: CD, CS 3008 **Applicable to Courses: B.Tech**
Full Marks=50 **Time:2 Hours**

SECTION-A(Answer All Questions. Each question carries 2 Marks)
Time:30 Minutes **(7×2=14 Marks)**

Question No	Question Type (MCQ /SAT)	Question	CO Mapping	Answer Key (For MCQ Questions only)
Q.No:1	MCQ	The part of a compiler that validates the derivation of the input string from the grammar of the language is, a. Semantic Analysis b. Syntax Analysis c. Data Flow Analysis d. Lexical Analysis	CO1	b
		The part of compiler that produces a stream of words with their syntactic categories from group of characters is, a. Semantic Analysis b. Data Flow Analysis c. Code Optimization d. Lexical Analysis	CO1	d
		The part of compiler that validates the variables used is consistent with their definitions and meaning is, a. Semantic Analysis b. Lexical Analysis c. Code Generation d. Syntax Analysis	CO1	a
		The part of compiler that tries to transform of IR to IR for improving the IR (IR means intermediate representation) is, a. Syntax Analysis b. Intermediate Code Generation c. Code Optimization d. Lexical Analysis	CO1	c
Q.No:2	MCQ	Consider the syntax directed translation scheme S –> x{print “3”}xW, S -> y{print “2”}, W -> S{print “1”}z. What will get printed during semantic processing of “xxxxyzz” using the above mentioned syntax directed translation scheme? a. 32131 b. 33211 c. 31312 d. 21313	CO4	b
		Consider the syntax directed translation scheme A -> xB{print “a”}x , A -> {print “b”}y , B -> z{print “c”}A. What will get printed during semantic processing of “xzxzyxx” using the above mentioned syntax directed translation scheme? a. bcaca b. acacb c. accba d. ccbaa	CO4	d

		Consider the syntax directed translation scheme P -> Q{print "2"}xx , P -> y{print "1"}, Q -> {print "3"}Pz. What will get printed during semantic processing of "yzxxzxx" using the above mentioned syntax directed translation scheme? a. 23231 b. 13232 c. 33122 d. 13322	CO4	c
		Consider the syntax directed translation scheme M -> {print "c"}xxN , M -> y{print "a"}, N -> z{print "b"}M. What will get printed during semantic processing of "xxzxxzy" using the above mentioned syntax directed translation scheme? a. cbcba b. ccabb c. acbcb d. cbcba	CO4	a
Q.No:3	**MCQ**	Consider the expression grammar A->A / X | A, X->T-X | X * T | T, T->T+F | F, F->a for generating arithmetic expressions over a. Which of the following is false? a. + is associative from left b. / is associative from left c. - is associative from left d. * is associative from left	CO3	c
		Consider the expression grammar E->E / B | E, B -> C - B | B * C | C, C->C+D | D, D->d for generating arithmetic expressions over d. Which of the following is true? a. + has highest precedence b. - has highest precedence c. * has highest precedence d. / has highest precedence	CO3	a
		Consider the expression grammar S-> S / Y | S, Y -> Z - Y | Y * Z | Z, Z -> Z+W | W, W->b for generating arithmetic expressions over b. Which of the following is false? a. Precedence of / is lesser than - b. Precedence of + is higher than - c. Precedence of / and * are same d. Precedence of - and * are same	CO3	c
		Consider the expression grammar P -> P / Q | P, Q -> R - Q | R * Q | R, R -> S + R | S, S->x for generating arithmetic expressions over x. Which of the following is false? a. Operator + is right associative b. Operator - is right associative c. Operator / is right associative d. Operator * is right associative	CO3	c
Q.No:4	**MCQ**	Which of the following is a valid optimized code after applying common sub-expression elimination and copy propagation to the code {a = x+y; b = a*z; c = x+y; d = c*z}? a. {u = x+y; b = u*z; d = u*z} b. {a = x+y; b = a*z; c = a; d = c*z} c. {u = x+y; v = u*z; b = v; d = v} d. {a = x+y; b = a*z; c = a; d = a}	CO5	c
		Which of the following is a valid optimized code after applying constant propagation and folding to the code {x=5; y=7; if y==7 goto L; z=x+5; L: x=10}? a. {x=5; y=7; if y==7 goto L; z=10; L: x=10} b. {x=5; y=7; z=a+5; x=10} c. {x=5; y=7;if y==7 goto L; z=15; L: x=10} d. {x=5; y=7; z=15; x=10}	CO5	d
		Which of the following is a valid optimized code after applying loop invariant code motion to the code {L: i=i+1; t1=a; t2=t1-4;	CO5	c

		t3=2*t2; t4=4*i; t5=t3+t4; goto L}? a. {t1=a; L: i=i+1; t2=t1-4; t3=2*t2; t4=4*i; t5=t3+t4; goto L} b. {t1=a; t2=t1-4; L: i=i+1; t3=2*t2; t4=4*i; t5=t3+t4; goto L} c. {t1=a; t2=t1-4; t3=2*t2; L: i=i+1; t4=4*i; t5=t3+t4; goto L} d. {t3=2*t2; L: i=i+1; t1=a; t2=t1-4; t4=4*i; t5=t3+t4; goto L}		
		Which of the following is a valid optimized code after applying strength reduction to the code {L: i=i+1; t=a; u=t-4; v=2+v; w=4*i; z=v+w; goto L}? a. {w=4; L: i=i+1; t=a; u=t-4; v=2+v; z=v+w; goto L} b. {w=4; L: i=i+1; t=a; u=t-4; v=2+v; w=i; z=v+w; goto L} c. {w=4; L: i=i+1; t=a; u=t-4; v=2+v; z=v+w; w=w+1; goto L} d. {w=4; L: i=i+1; t=a; u=t-4; v=2+v; z=v+w; w=w+4; goto L}	CO5	d
Q.No:5	**MCQ**	Suppose M is a LL(1) table for the grammar having productions A -> XY | b | ∈. Assuming FIRST(X) ={a,b} and FOLLOW(A) = {b,c}, how many total number of productions are present in M[A,b]? a. 2 b. 3 c. 4 d. 5	CO3	b
		Suppose N is a LL(1) table for the grammar having productions A -> Bx | y | ∈, B -> ∈. Assuming FOLLOW(A) = {x,$}, how many total number of productions are present in N[A,x]? a. 2 b. 3 c. 4 d. 5	CO3	a
		Suppose P is a LL(1) table having the productions {X -> YaZb, X -> ∈, X -> a} present in P[X,a]. Which of following is not possible for FOLLOW(X)? a. {a,b,$} b. {c,b,$} c. {a,c,$} d.{a,b,c}	CO3	b
		Suppose Q is a LL(1) table having the productions {A -> cMbd, A -> ∈, A -> Na} present in Q[A,c]. Which of following sets is valid for FOLLOW(A)? a. {a,b,d} b. {b,d,$} c. {a,d,c,$} d. {a,d,$}	CO3	c
Q.No:6	**MCQ**	Suppose the address descriptor for x and y contains itself. What will be the content of the address descriptor for x and y after generating the machine code for the instruction z=x+y assuming register A is used for x and register B is used for y and z respectively? a. {(x,A),(y,B)} b. {(x,A),(y)} c. {(x),(y,B)} d. {(x)(y)}	CO6	b
		Suppose the address descriptor for p and q contains itself. What will be the content of the address descriptor for p and q after generating the machine code for the instruction r=p*q assuming register X is used for q and register Y is used for p and r respectively? a. {(p,Y),(q,X)} b. {(p,Y),(q)} c. {(p),(q,X)} d. {(p)(q)}	CO6	c
		Suppose the address descriptor for m and n contains itself. What will be the content of the address descriptor for m and n after generating the machine code for the instruction t=m-n assuming register R is used for m and register S is used for n and register T used for t respectively? a.{(m,R),(n,S)} b.{(m,R),(n)} c.{(m),(n,S)} d.{(m)(n)}	CO6	a
		Suppose the address descriptor for c contains itself and address descriptor for d contains d and D respectively. What will be the content of the address descriptor for c and d after generating the machine code for the instruction y=c/d assuming register C is used for c and register D is used for d and y respectively? a.{(c,C),(d,D)} b.{(c,C),(d)} c.{(c),(d,D)} d.{(c)(d)}	CO6	b
Q.No:7	**MCQ**	A lexical analyzer uses the pattern 0?(1 | 2)*1 and 0?(0 | 2)*2 for recognizing the tokens A and B respectively. What will be the token sequence for the lexeme 12102021? a. ABA b. AAB c. BAA d. BBA	CO2	a

		A lexical analyzer uses the pattern 1?(0 | 2)*0 and 2?(1 | 0)*1 for recognizing the tokens C and D respectively. What will be the token sequence for the lexeme 20121120? a. DCD b. CCD c. DDC d. CDC	CO2	c
		A lexical analyzer uses the pattern x?(y | z)*y and y?(z | x)*z for recognizing the tokens M and N respectively. What will be the token sequence for the lexeme yzyxxzzy? a. NMM b. MMN c. MNN d. MNM	CO2	d
		A lexical analyzer uses the pattern x?(x | z)*y and x?(y | x)*z for recognizing the tokens P and Q respectively. What will be the token sequence for the lexeme xxyzxyzy? a. PQQ b. PQP c. QPP d. QPQ	CO2	c

SECTION-B(Answer Any Three Questions. Each Question carries 12 Marks)

Time: 1 Hour and 30 Minutes **(3×12=36 Marks)**

Question No	Question	CO Mapping (Each question should be from the same CO(s))
Q.No:8	Suppose a Lexical Analyzer admits the function names as letter followed by one or more occurrence of letters or digits but ending with a letter and also admits the operators **+, + =, −=, and −.** Design an appropriate regular definition for recognizing function names and the above said operators. Draw a transition diagram (DFA) that matches with the above designed regular definition. Consider a target machine X with two registers R_1 and R_2 that allows the LD, ST, ADD, SUB, MUL machine instructions. Assuming the size of each element of the arrays a and b as 8 bytes generate the target code of the statement **a[k] = b[i] + c * 2** for the target machine X.	CO2 CO6
	Suppose a Lexical Analyzer accepts the variable names as letter or $ followed by one or more occurrence of letters, digits but ending with a # and also accepts the operators **&&, &=, & and =.** Design an appropriate regular definition for recognizing variable names and the above said operators. Draw a transition diagram that matches with the above designed regular definition. Consider a target machine M with two registers P_1 and P_2 that allows the LD, ST, ADD, SUB, MUL machine instructions. Assuming the size of each element of the arrays x and y as 4 bytes generate the target code of the expression **n = x[k] * y[j] - c** for M.	
	Suppose a Lexical Analyzer admits the type names as letter followed by one or more occurrence of letters, digits but	

<table>
<tr>
<td></td>
<td>ending with a digit and the operators ||, !=, | and ==. Design an appropriate regular definition for recognizing type names and the above said operators. Draw a transition diagram that matches with the above designed regular definition. Consider a target machine N with two registers C_1 and C_2 that allows the LD, ST, ADD, SUB, MUL machine instructions. Assuming the size each element of the arrays p and r as 8 bytes generate the target code of the expression r[k] = p[j] * y - c for N.</td>
<td></td>
</tr>
<tr>
<td><u>Q.No:9</u></td>
<td>Consider the code,
for(i=1; i <= 8; i++)
{
 for(j = 0; j < 8; j++)
 {
   if(arr[i][j] > 2)
      arr[i][j] = 5;
 }
}
Considering the size of arr as 8X8 and size of each element as 4 bytes, find the three address code for the above code. Find the control flow graph for the above three address code. Represent the three address code of “x = a * y + b[i]” in quadruple and triple notation.</td>
<td>CO5</td>
</tr>
<tr>
<td></td>
<td>Consider the code,
m=0;
while(m<6)
{
 for(n = 1; n <= 6; n++)
 {
  brr[m][n] = 2;
 }
 if(m > 1)
   m = m+7;
}
Assuming the size of brr as 6X6 and size of each element as 8 bytes, find the three address code for the above code. Construct the control flow graph for the above three address code. Represent the three address code of “a[i] = x + b / c” in triple and quadruple notation.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Consider the code,
q = 1;
for(p = 0; p < 7; p++)
{
  while(q <= 7)
  {
    if(crr[p][q] < 5)
       crr[p][q] = 2;
    q = q + 1;
  }
}
Assuming the size of crr as 7X7 and size of each element as 4 bytes, find the three address code for the above code. Construct the control flow graph for the above three address code. Represent the three address code of “if ((a+x*y) < b) goto L” in quadruple and triple notation.</td>
<td></td>
</tr>
</table>

Q.No:10	Suppose the first two letters (in upper case) of your first name represented by FF, SF and the last two digits of your roll number is represented by SD,LD respectively (in case first name is a single letter take the last name into consideration). Further, assume the productions of the grammar have the fixed pattern {FF->SF SF SD, SF->SD SF | LD}. Obtain a grammar by replacing the characters and digits of your name and roll number in the above pattern. Construct a SLR(1) parsing table for your grammar. Further, provide the first three states of CLR automaton. Finally, design a DAG for the expression **(p/q+q)*(p-q/p)*(p-q).**	CO3
	Suppose the first two letters (in upper case) of your first name represented by FF, SF and the last two digits of your roll number is represented by SD,LD respectively (in case first name is a single letter take the last name into consideration). Further, assume the productions of the grammar have the fixed pattern {FF->SF SF SD, SF->SD SF | LD}. Obtain a grammar by replacing the characters and digits of your name and roll number in the above pattern. Construct a SLR(1) parsing table for your grammar. Further, provide the first three states of LR(1) automaton. Finally, draw a DAG for the expression **(x*y+x)-(x+y)*(x*y).**	
	Suppose the first two letters (in upper case) of your first name represented by FF, SF and the last two digits of your roll number is represented by SD,LD respectively (in case first name is a single letter take the last name into consideration). Further, assume that the productions of the grammar have the fixed pattern {FF->SF SF SD, SF->SD SF | LD}. Obtain a grammar by replacing the characters and digits of your name and roll number in the above pattern. Construct a SLR(1) parsing table for your grammar. Further, provide the first three states of LALR automaton. Finally, draw a DAG for the expression **(a-b)*(a*b)+a+a*b.**	
Q.No:11	Consider the following syntax directed definition (SDD) scheme, S -> S + A {S.v = A.v * S.v} S -> A {S.v = A.v ^ 2} A -> A * B {A.v = B.v * 2 + A.v} A -> B {A.v = B.v + 1} B -> digit {B.v = digit.lexval} where ^ represents the exponentiation operator. A. Draw the parse tree for the string “x * y + z” where x, y, and z are the last three digits of your roll number respectively. For example, for the roll number 1705234 the values of x, y, and z are 2, 3, and 4 respectively. B. Evaluate the above string using the above given SDD. C. Design a grammar to generate the strings of the language L = {$a^m b^n$: m, n > 0}. Add semantic actions to your grammar to check whether number of b’s is exactly one more than number of a’s in any string of L or not. D. Left factor the grammar P -> PqR | PQr | PQR	CO4

Consider the following syntax directed definition (SDD) scheme,

P -> P * Q {P.v = P.v || Q.v}
P -> Q {P.v = Q.v + 1}
Q -> Q - R {Q.v = Q.v + R.v * 2}
Q -> R {Q.v = R.v + 1}
R -> digit {R.v = digit.lexval}

where || represents the concatenation operator.

A. Draw the parse tree for the string "p - q * r" where p, q, and r are the last three digits of your roll number respectively. For example, for the roll number 1705789 the values of p, q, and r are 7, 8, and 9 respectively.
respectively.
B. Evaluate the above string using the above given SDD.
C. Design a grammar to generate the strings of the language M = $\{1^a0^b$: a, b > 0}. Add semantic actions to your grammar to check whether number of 1's is exactly one less than number of 0's in any string of M or not.
D. Left factor the grammar M -> aBX | abB | aBY

Consider the following syntax directed definition (SDD) scheme,

X -> X / Y {X.v = X.v + Y.v}
X -> Y {X.v = Y.v * 2}
Y -> Y * Z {Y.v = (Y.v || Z.v) + 2}
Y -> Z {Y.v = Z.v + 2}
Z -> digit {Z.v = digit.lexval}

where || represents the concatenation operator.

A. Draw the parse tree for the string "a / b * c" where a, b, and c are the last three digits of your roll number respectively. For example, for the roll number 1705567 the values of a, b, and c are 5, 6, and 7 respectively.
B. Evaluate the above string using the above given SDD.
C. Design a grammar to generate the strings of the language P = $\{x^cy^d$: c, d > 0}. Add semantic actions to your grammar to check whether number of y's is same as number of x's in any string of P or not.
D. Left factor the grammar A -> BXy | BBc | BXz