SPRING END SEMESTER EXAMINATION-2019

6th Semester B.Tech & B.Tech Dual Degree

COMPILER DESIGN

CS3002 / CS-601

(For 2017(L.E), 2016 & Previous Admitted Batches)

Time: 3 Hours　　　　Full Marks: 60

Answer any SIX questions.
Question paper consists of four sections-A, B, C, D.
Section A is compulsory.
Attempt minimum one question each from Sections B, C, D.
The figures in the margin indicate full marks.
Candidates are required to give their answers in their own words as far as practicable and all parts of a question should be answered at one place only.

SECTION-A

1. Briefly answer all questions. [2×10]

(a) Which phase in compiler is responsible for co-relating error messages with line numbers? How it is achieved during compilation?

(b) For a C program accessing X[i][j] in a two dimensional array of 8 rows, the following intermediate code is generated by a compiler.

$s = i * 128$

$t = j * 2$

$u = s + t$

$v = X[u]$

Assuming size of integer and character is of 4 and 2 bytes respectively, What will be the declaration of X corresponding to the above code? Justify.

(c) Suppose an arbitrary compiler produces the tokens A, B and C upon alphabet $\{x, y, z\}$ using the following patterns.

A: $x^*y?z$

B: $x?y^*z$

C: $xy?z^*$

If the string xyxxyzyz is processed by the analyzer, then what is the sequence of tokens produced as output?

(d) In an arbitrary language, the identifiers are of pattern "starting with # or @, and containing any combination of letter, digit, or $ in between". Write a regular definition to recognize identifiers of this language.

(e) Consider the syntax directed translation scheme,

$S \rightarrow ABC \quad \{S.u = B.u, B.u = A.v, A.v = A.v + 1\}$

How many synthesized and inherited attributes are present in the above scheme? Justify.

(f) Explain the changes need to reflected in address and register descriptor for the instruction **LD R, x**.

(g) Construct a LL(1) grammar with two non-terminals S, T such that $\text{FIRST}(S) = \text{FOLLOW}(T) = \{b, c, \$\}$. Justify your answer.

(h) Consider a Context-free grammar G without any ϵ-productions. For any two productions $A \rightarrow \alpha$ and $A \rightarrow \beta$ of G, First(α) and First(β) are disjoint. Justify that the grammar is LL(1).

(i) In LR(0) automaton, a state may suffer from either shift-reduce or reduce-reduce conflict. What conflicts the state may suffer produced from the canonical set of items $\{S \rightarrow T.+R, Q \rightarrow R.\}$ on input symbol + ? Justify your answer.

(j) How L-attributed definition is different from S-attributed definition? Provide at least two major differences.

SECTION-B

2. (a) Explain the following optimization techniques with appropriate examples. [4]

    i. Constant Propagation

    ii. Dead Code Elimination

    iii. Loop Invariant Code Motion

    iv. Common Subexpression Elimination

   (b) Consider the following grammar [4]

    $A \rightarrow BCB$

    $B \rightarrow Cd \mid cB \mid \in$

    $C \rightarrow eB$

    a. Compute FIRST and FOLLOW of each non-terminal.

    b. Check whether given grammar is LL(1) or not.

3. (a) Translate the intermediate code corresponding to the given expression "$\mathbf{x + x * (y - z) + (y - z) * t}$" in quadruple, triple and indirect triple respectively. [4]

   (b) What do you mean by "$handle\ pruning$"? Parse the string "$aabbbbdc$" using Shift-Reduce parsing technique over the grammar, [4]

    $A \rightarrow aaB \mid bbA$

    $A \rightarrow b \mid d$

    $B \rightarrow Ac$

SECTION-C

4. (a) Construct LR(0) parsing table for the following grammar, [4]

$D \rightarrow (D) \mid b$

(b) Consider the following grammar, [4]

$S \rightarrow iEtS \mid iEtSeS \mid c$

$E \rightarrow a$

a. Examine that the given grammar is ambiguous.

b. Develop an unambiguous grammar equivalent to the above grammar

5. (a) Discover the basic blocks present in the following three address code followed by the corresponding flow graph. [4]

(1) $i = m - 1$

(2) $k = n$

(3) $t1 = 4 * n$

(4) $v = a[t1]$

(5) $i = i + 1$

(6) $t2 = 4 * i$

(7) $t3 = a[t2]$

(8) if $t3 < v$ *goto* (5)

(9) $k = k - 1$

(10) $t4 = 4 * j$

(11) $t5 = a[t4]$

(12) if $t5 > v$ *goto* (9)

(b) Build a left recursion free grammar G′ for the following grammar G such that $L(G') = L(G)$. [4]

$S \rightarrow Aa \mid B$

$A \rightarrow Bb \mid Sc \mid \epsilon$

$B \rightarrow d$

6. (a) Suppose our objective is to build maximum number of compilers with M number frontends and N number of backends respectively. Then, compute the total number of frontends and backends need to be hardcoded in the following cases. [4]

a. Frontend and backend are independent

b. Frontend and backend are dependent

(b) An arbitrary compiler has the specifications, keywords = {if, for, func} and numbers consisting of digits that starts with 0x and can have optional fractional part. Construct a transition diagram to recognize keywords and numbers in this compiler. [4]

SECTION-D

7. (a) Consider the syntax directed translation scheme (SDTS) given in the following evaluating attributes in bottom up manner. [4]

$E \rightarrow E_1 * T \quad \{E.val = E_1.val * T.val\}$

$E \rightarrow T \quad \{E.val = T.val\}$

$T \rightarrow F - T_1 \quad \{T.val = F.val - T_1.val\}$

$T \rightarrow F \quad \{T.val = F.val\}$

$F \rightarrow 3 \quad \{F.val = 3\}$

$F \rightarrow 4 \quad \{F.val = 4\}$

Construct an annotated parse tree for the expression "$4 - 3 - 4 * 3$" and evaluate the expression using the parse tree.

(b) Use the synthesized attribute red and modify the above syntax directed translation scheme to compute number of reductions required while reducing an input expression to the start symbol E. [4]

8. (a) Prove that the following grammar is suitable for CLR parsing but not suitable for SLR parsing. [4]

$A \rightarrow CaCb \mid DbDa$

$C \rightarrow \in$

$D \rightarrow \in$

(b) Generate the machine code for the following three address code corresponding to a basic block. [4]

$t = b * c$

$u = d * e$

$v = u * t$

$x = v + f$

*****