28/11/19

INDIAN INSTITUTE OF ENGINEERING SCIENCE AND TECHNOLOGY, SHIBPUR
B.Tech-M.Tech Dual Degree (CST) 7^{th} Semester Examination, 2019
Compiler Design (CS - 701)

Time: 3 hours **Full Marks: 70**

(Answer Question No 1 and any five from the remaining)

1. (a) What is phrase level error recovery?
   (b) What is code motion?
   (c) Why is it necessary to generate intermediate code instead of generating target program directly?
   (d) What are the various methods of implementing three address statements?
   (e) What are the properties of optimizing compiler?

   $[2+2+2+2+2=10]$

2. (a) Write regular definition for the following language over {0,1} a string of 0's and 1's, which has 0 at the third position when counted from night.
   (b) Draw the transition diagram for the lexical analyzer that recognize the tokens like identifiers and relational operators. Use the following rules to form the identifier:
   - begins with an alphabet
   - consists of alphabets, digits and hyphen
   - should not end with an hyphen
   - not two hyphens appear together

   (c) What is a symbol table? List the functionalities of a symbol table.

   $[3+(3+3)+3=12]$

3. Consider the following Grammar

$$S \rightarrow X|ay$$
$$X \rightarrow xXy|Y$$
$$Y \rightarrow a$$

   where S is the start symbol, set of non-terminals are {a, x, y} and set of terminals are {5, 3, a, b}
   (a) Compute the set of SLR(1) items for this grammar
   (b) Is this grammar SLR(1)? Briefly explain why or why not.

   $[8+4=12]$

4. (a) Write an attribute grammar to count the number of digits to the right of a decimal point or a floating point value of a decimal number given by the following grammar: (Hint: Use a *count* attribute to count the number of digits to the right of the decimal point.)

$$dnum \rightarrow num.num$$
$$num \rightarrow num\ digit|\ digit$$
$$digit \rightarrow 0|1|2|3|4|5|6|7|8|9$$

   (b) Draw a flow diagram for the *FOR* loop and subsequently propose the semantic rules to generate three address code for the same.
   (c) Write the rules to partition a sequence of 3 address statements for forming basic blocks.

   $[4+4+4=12]$

5. Consider the following grammar G for arithmetic expressions:

$$E \rightarrow E + T \mid T$$
$$T \rightarrow T * F \mid F$$
$$F \rightarrow (E) \mid id$$

where E is the start symbol, set of non-terminals are $\{E, T, F\}$ and set of terminals are $\{$ *, (,), id $\}$.

(a) Compute $FIRST$ and $FOLLOW$ for each nonterminal in G.

(b) Construct a predictive parsing table for G.

(c) Show how your predictive parser processes the input string:

$$id + id * id$$

Draw the parse tree traced out by your parser.

$[6 + 3 + 3 = 12]$

6. (a) List the fields in an activation record. Write down the purpose of each of these fields in an activation record.

(b) Explain the sequence of stack allocation process for a function call using a suitable example.

(c) Why symbol-table is needed in various phases of compilers? How hashing can be used to design symbol-table.

$[4 + 4 + 4) = 12]$

7. (a) What are the different types of errors a program can contain? List out different error handling strategies.

(b) Explain the following code optimization techniques with examples: Constant propagation, Strength reduction and Code Motion.

(c) Discuss how induction variables can be detected and eliminated from the given intermediate code:

```
B2: i= i+1
    t1=4*j
    t2=a[t1]
    if t2<10 goto B2
```

$[3 + 6 + 3 = 12]$