ISLAMIC UNIVERSITY OF TECHNOLOGY (IUT)
ORGANISATION OF ISLAMIC COOPERATION (OIC)
Department of Computer Science and Engineering (CSE)

EMESTER FINAL EXAMINATION — SUMMER SEMESTER, 2013-2014

URATION: 3 Hours — FULL MARKS: 150

CSE 4801: Compiler Design

Programmable calculators are not allowed. Do not write anything on the question paper.

There are **8 (eight)** questions. Answer any **6 (six)** of them.

Figures in the right margin indicate marks.

) Discuss the role of a *Symbol Table Manager* during compilation. 6
) Explain the ways to implement *Symbol Table* in context of data structure. 12
) State the cousins of compiler along with their functions. 7

) Compilation process can be decomposed into several phases. State the typical phases of a compiler. Explain the importance of implement compiler in separate phases. 4+6
) Classify Grammars, and define each of the class. 7
) Explain the position and role of a *syntax analyzer* in a multi-phase compilation model. 8

) Discuss on various error recovery strategies to recover from syntactic errors. 12
) The following grammar for **if-then-else** statements is proposed to remedy the **dangling-else** ambiguity: 8

stmt → if *expr* then *stmt*
| *matched_stmt*
matched_stmt → if *expr* then *matched_stmt* else *stmt*
| other

Show that the grammar is still ambiguous.
) Explain when and why the **elimination of left recursion** is necessary. 5

) Define L-Attributed definitions. Explain how L-Attributed definitions can be evaluated by depth first order method. 10
) Explain the bottom-up evaluation of S-Attributed Definitions using parser stack. 10
) What is annotated parse tree? Discuss how a dependency graph can be constructed from an annotated parse tree to help evaluate the attributes. 5

a) Consider the following grammar: 15

$E \rightarrow E + T \mid T$
$T \rightarrow T\,F \mid F$
$F \rightarrow F^* \mid (S) \mid a$

Construct the SLR parser table for this grammar.
b) Give the *Translation Scheme* for checking the type of following statements: 10

$S \rightarrow id = E$
$S \rightarrow if\ E\ then\ S_1$
$S \rightarrow while\ E\ do\ S_1$
$S \rightarrow S_1; S_2$

320

6. a) Write short notes on the followings: 3x
   i. Left factoring ii. Handle pruning iii. Error handler
   b) Consider the grammar

$E \rightarrow E+T \mid E-T \mid T$
$T \rightarrow (E) \mid id \mid num$

Give the syntax-directed definition for constructing a syntax tree for an expression. Use the functions *mknode(op, left, right), mkleaf(id.entry), mkleaf(num.entry)* to create nodes of the syntax tree.

c) Find the set of FIRST and FOLLOW for the non-terminals of the following grammar: 7

$E \rightarrow TE'$
$E' \rightarrow +TE' \mid -TE' \mid \varepsilon$
$T \rightarrow FT'$
$T' \rightarrow *FT' \mid \varepsilon$
$F \rightarrow -E \mid (E) \mid id \mid num$

7. a) Design ***syntax-directed definitions*** to generate three-address codes for the following productions- 6×2
   i. $S \rightarrow$ *if E then* S_1 *else* S_2
   ii. $S \rightarrow$ *for*$(i=1;\ i<n;\ i++)\ S_1$
   b) Explain various ways to implement three-address statements. 13

8. a) A string contains a random sequence of numbers and words. Write a *Lex* program to read the string and produce sum of each consecutive numbers and concatenate each consecutive words. 10

Sample input: Dhaka 50 60 70 Winter Summer 10
Output: Dhaka 180 Winter Summer 10

b) Write a *Lex* program which will take a file name as an argument and count and print the number of vowels, consonants, words and other symbols presented in the file. 15