

ISLAMIC UNIVERSITY OF TECHNOLOGY (IUT)
ORGANISATION OF ISLAMIC COOPERATION (OIC)

Department of Computer Science and Engineering (CSE)

SEMESTER FINAL EXAMINATION

SUMMER SEMESTER, 2017-2018

DURATION: 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.

-
- | | | |
|-------|--|----|
| 1. a) | Draw the block diagram of a language processing system. Briefly discuss each of its components. | 10 |
| b) | Write a <i>Lex</i> program which can recognize presence of an even number of alphabetic strings followed by an odd number of integer numbers in a text file. Text file name will be supplied as an argument to the program. The <i>Lex</i> program will report start and end position of such sequence(s) present in the provided text file. | 10 |
| c) | Discuss the functions of a symbol table manager. | 5 |
| 2. a) | Discuss the reasons for separating the analysis phase of compiling into lexical and parsing during an ideal compiler construction. | 10 |
| b) | Write a program using <i>Lex</i> and <i>Yacc</i> which can convert a prefix expression into postfix expression. | 10 |
| c) | Which applications are known as cousins of a compiler? | 5 |
| 3. a) | List various compiler construction tools along with brief description. | 10 |
| b) | Compare LL parser and LR parser. What are the meaning of Ls and R here? | 10 |
| c) | Write regular expression for a float number supported in C++. | 5 |
| 4. a) | Consider the following context-free grammar:
$G \rightarrow L$
$L \rightarrow E ; L$
$L \rightarrow E$
$E \rightarrow E + T$
$E \rightarrow T$
$T \rightarrow id$
$T \rightarrow id ()$
$T \rightarrow id (L)$
The terminal symbols of this grammar are ; + () id | 20 |
| i. | Derive a leftmost derivation, for the string $x + y ; z (y ())$ and show the corresponding parse tree. | |
| ii. | Transform this grammar so that it can be used to construct a top-down predictive parser. | |
| b) | How syntax tree differs from parse tree? Classify attributes used in syntax tree. | 5 |

5. a) Consider the following context-free grammar and the parse table:

1. $G \rightarrow L$
2. $L \rightarrow L P$
3. $L \rightarrow P$
4. $P \rightarrow (P)$
5. $P \rightarrow ()$

state	action			goto	
	()	\$	L	P
0	s3			1	2
1	s3		accpt		4
2	r3		r3		
3	s6	s7			5
4	r2		r2		
5		s8			
6	s6	s10			9
7	r5		r5		
8	r4		r4		
9		s11			
10		r5			
11		r4			

Show, in full detail, the steps that an LR(1) parser would follow to parse the string $(()) ()$ using the above grammar. For each step, show the contents of the stack, what the next input is and the action that is taken.

b) Consider the context-free grammar:

$$S \rightarrow SS^* \mid SS^+ \mid a$$

- i. Show that the string $aa+a^*$ can be generated by this grammar.
- ii. Construct a parse tree for the string.

c) Briefly explain the concept of derivation.

6. a) Design a translation scheme for checking the *types* of the following statements:

- $$\begin{aligned}
 S &\rightarrow \text{id} := E \\
 S &\rightarrow \text{if } E \text{ then } S1 \\
 S &\rightarrow \text{do } S1 \text{ while } E \\
 S &\rightarrow S1 ; S2
 \end{aligned}$$

- b) Discuss various methods to implement three-address statements.
- c) Write short notes *viable prefix* and *handle pruning*.

7. a) Discuss on heap allocation strategy for activation records. Mention its advantages over other allocation methods.

b) Write the syntax-directed definition to generate intermediate code for the following statements:

- $$\begin{aligned}
 S &\rightarrow \text{if } E \text{ then } S1 \\
 S &\rightarrow \text{if } E \text{ then } S1 \text{ else } S2 \\
 S &\rightarrow \text{do } S1 \text{ while } E \\
 S &\rightarrow \text{while } E \text{ do } S1
 \end{aligned}$$

8. a) What is an activation Record? List the contents of an activation record along with their description.

b) Write a *Lex* program which will take a file name as an argument and count the number of uppercase and lowercase letters, digits, words, white spaces, lines and other symbols presented in the file.