



SF-8331

B. E. - III (Sem. - VI) (Computer) Examination
May/June - 2011
System Programming

Time : Hours]

[Total Marks : 75

Instructions :

(1)

नीचे दशांशवैल निशानीवाणी विगतो उत्तरवही पर अवश्य लपवी. Fillup strictly the details of signs on your answer book.	Seat No. :
Name of the Examination :	<input type="text"/>
<input type="text" value="B. E. - 3 (SEM. - 6) (COMPUTER)"/>	<input type="text"/>
Name of the Subject :	<input type="text"/>
<input type="text" value="SYSTEM PROGRAMMING"/>	<input type="text"/>
Subject Code No. : <input type="text" value="8"/> <input type="text" value="3"/> <input type="text" value="3"/> <input type="text" value="1"/>	Section No. (1, 2,.....): <input type="text" value="Nil"/>
Student's Signature	

- (2) Make assumptions whenever required.
(3) Numbers on the **right** indicate marks.

- 1 Attempt any **five** : 5
- (i) Define : Token
 - (ii) Define : compiler
 - (iii) Define : context free grammar
 - (iv) Define : Parse tree
 - (v) Which phase of compiler is optional phase ?
 - (vi) What is need of intermediate code ?
 - (vii) Top down parser can never work with left recursive grammar. - State True/False.

- 2 (a) Construct an optimized DFA : 7
- aa^*bc^*

OR

- (a) Show that following regular expressions are equivalent 7
by constructing optimized DFA.
- (i) $(a/b)^*$
 - (ii) $(a^*/b)^*$
- (b) Construct LL (1) parsing table for following grammar. 4
- $S \rightarrow A$
 $A \rightarrow aB \mid aC \mid Ad \mid Ae$
 $B \rightarrow bBc \mid f$
 $C \rightarrow g$

- (C) Eliminate left recursion from following grammar. 3
 $S \rightarrow ABC$
 $A \rightarrow Aa \mid @$
 $B \rightarrow Bb \mid @$
 $C \rightarrow Cc \mid @$
- 3** Attempt the following : 16
- (i) What is code optimization ? Explain two methods for code optimization in detail.
 - (ii) What is three address code ? Explain triples and quadruples in detail.
 - (iii) State different storage allocation strategies. Explain static allocation and stack allocation in detail.
 - (iv) What is main task of semantic analysis phase ? Define : inherited attribute and synthesized attribute.
- 4** Attempt any **five** : 5
- (i) Define : Symbol Table
 - (ii) Define : Backpatching
 - (iii) Define : Loaders
 - (iv) Define : Macro Assembler
 - (v) Define : Backward References
 - (vi) State True or False : “Symbol table is used to store mnemonics and opcodes.”
 - (vii) State True or False : “Single pass assemblers cannot handle forward references.”
- 5** (a) Explain in brief design of a Two Pass Assembler. 8
(b) Explain in brief design the linker. 8
- OR**
- (b) Explain in brief the design of a macro assembler. 8
(c) Write and explain the algorithm for macro expansion. 4
- 6** Attempt the following : (any **three**) 15
- (i) Explain how forward references are handled in single pass assembler.
 - (ii) Explain in brief self relocating programs.
 - (iii) Explain and compare two variants of intermediate code.
 - (iv) Compare and contrast macro preprocessor and macro assembler.