

UNIVERSITY OF PUNE
[4363]-265
T. E.(IT)Examination - 2013
SYSTEM SOFTWARE PROGRAMMING
(2008 Pattern)

[Total No. of Questions:12]
[Time: 3 Hours]

[Total No. of Printed Pages :2]
[Max. Marks: 100]

Instructions :

- (1) Answer Q1 or Q2, Q3 or Q4, Q5 or Q6 from section I and Q7 or Q8, Q9 or Q10, Q11 or Q12 from section II.*
 - (2) Answers to the two sections should be written in separate answer-books.*
 - (3) Black figures to the right indicate full marks.*
 - (4) Neat diagrams must be drawn wherever necessary.*
 - (5) Assume suitable data, if necessary.*
-

SECTION-I

- Q1 a) What feature of Assembly makes to design pass II of two pass Assembler? Explain with suitable example. [6]
b) Describe different types of errors that are handled by Pass I & Pass II of two pass Assembler. [6]
c) Enlist different Data Structures created during Pass I of two pass Assembler. Explain with suitable example. [6]

OR

- Q2 a) Write an algorithm for Pass-II of two pass Assembler and explain with suitable example. [10]
b) What do you mean assembler directives? Explain how assembler directives LTORG, ORIGIN and EQU are processed in pass-I of two pass assembler. [8]
- Q3 a) Describe Macro definition within Macro definition with suitable Examples. [8]
b) What are the different ways in which we can specify the arguments to a Macro call? Explain with suitable examples. [8]

OR

- Q4 a) Write an algorithm for Pass-II of two pass Macro Processor with suitable example. [8]

- b) Differentiate between Macro and Subroutine with respect to execution [4]
speed and memory space.
- c) Write a short note on C-Preprocessor. [4]
- Q5 a) Explain Shift Reduce Parser with example also enlist what are major [8]
problems with Shift Reduce Parser.
- b) Differentiate between top down and bottom up parser. [6]
- c) What are the basic tasks of scanner? [2]

OR

- Q6 a) Explain the problem of left factoring in Top down Parser. [6]
- b) Write short note on Lexical Analysis. [6]
- c) What is ambiguous grammar? Explain with example. [4]

SECTION-II

- Q7 a) Explain Machine dependent code optimization technique with suitable [8]
example.
- b) For the statement given below, generate intermediate code in the [8]
format. i) Quadruple, ii) Triple, iii) Parse tree, iv) Postfix
notation, $A = -P * (-Q + R)$

OR

- Q8 a) What are the issues in code generation [8]
- b) Explain the importance of intermediate code generation in compiler. [4]
- c) Write a short note on activation record. [4]
- Q9 a) Explain Binary Symbolic subroutines (BSS) loading scheme with [10]
Example. Also discuss how allocation, relocation, linking & loading is done.
- b) Explain with the help of Flow chart the design of an absolute loader. [8]

OR

- Q10 a) Explain the following [6]
i) Overlay Structure ii) Linkage editor.
- b) What is loader? Enlist basic functions of a loader? [6]
- c) Compare: i) Dynamic loading Vs Dynamic linking [6]
- Q11 a) Describe various types of editors? Explain with the help of the block [12]
diagram of Typical Editor structure.
- b) Differentiate between Line and Screen editor. [4]

OR

- Q12 a) Write a short note on Debug Monitor. [4]
- b) With the help of diagram, Explain user interface & its use in software [6]
application.
- c) Explain following language processing tools. [6]
i) LEX ii) YACC