

GUJARAT TECHNOLOGICAL UNIVERSITY**SEMESTER– 3 EXAMINATION – WINTER 2012****Subject code: 630005****Date: 04/01/2013****Subject Name: System Software****Time: 10:30 – 13:00****Total Marks: 70****Instructions:**

1. Attempt all questions.
2. Make suitable assumptions wherever necessary.
3. Figures to the right indicate full marks.

- Q.1 (a)** Explain the following terms in brief: **07**
1. Translator.
 2. System software.
 3. Non terminal symbol.
 4. DFA.
 5. Operand Descriptor.
 6. Structure Editor.
 7. Regular Expression.
- (b)** Answer the following questions:
1. What is grammar? Give classification of the grammar along with its example. **04**
 2. Explain the significance of search and allocation data structures. **03**
- Q.2 (a)** State whether following statement are true or false. Justify your answer: **07**
1. A preprocessor is a language processor that bridges execution gap by generating an object code.
 2. Syntax analysis is performed to check the validity of a production rule.
 3. A non terminal symbol always appears as a leaf node of AST.
 4. The term binding is an association of, a value of an entity with its attribute.
 5. Rehashing is a technique used for garbage collection in heaps.
 6. The mnemonic MOVER falls under the category of assembler directive.
 7. The term CSF in parsers refer to current sentential form.
- (b)** Differentiate between, character and block device driver. Draw block diagram of character device driver and explain it in detail. **07**
- OR**
- (b)** What is device driver? List and explain any four entry points used in block device driver. **07**
- Q.3 (a)** Discuss the problems encountered during the single pass assembly. **07**
- (b)** Draw a flow chart or write an algorithm for pass II of multi pass assembler. How many passes can a multi pass assembler have? **07**

OR

- Q.3 (a)** How does single pass assembler differ from multi pass assembler? Explain the significance of FRT, SRTAB_ARRAY and CRT. **07**
- (b)** Given the following assembly program, show the contents of all the tables generated after pass I. Also generate IC using Variant I. **07**

```

                START      300
B               DC         3
                MOVER      AREG, ='5'
                MOVEM      AREG, A
L1             MOVER      CREG, B
                ADD        CREG, ='1'
                COMP CREG, AREG
                BC         NE, NX
                LTORG
                ='5'
                ='1'
NX             SUB        AREG, ='1'
                COMP CREG, AREG
                BC         LT, LS
LS             STOP
                ORIGIN     L1 + 2
                MULT CREG, B
A              DS         1
                END
```

- Q.4 (a)** Draw the block diagram of SEGDEF, EXTDEF, LEDAT and FIXUP records of an object record. Explain the significance of each of them. **07**
- (b)** Discuss different types parameters used in macro with example. **07**

OR

- Q.4 (a)** Define the following terms: **07**
1. Translated time origin
 2. Linked origin
 3. Relocation factor
 4. Overlay
 5. Loader
 6. Object Module
 7. Binary Program
- Q.4 (b)** What is a macro? Given the following macro show all the data structure entries during macro definition and call. If call is made as CLEAR Arr,5. **07**

```

                MACRO
CLEAR          &X, &Y, &Z = AREG
LCL           &A
&A            SET      5
                MOVER    &Z, 0
.MORE         MOVEM    &Z, &X + &A
&A            SET      &A + 1
AIF           ( &A NE &Y) .MORE
```

MEND

- Q.5 (a)** Given a postfix string $| - a b c * + d e f ^ * + - |$ generate triples, indirect triples and quadruples. Explain the significance of these forms. **07**
- (b)** Explain the significance of parsing. Given a string $| - a - b * c + d - |$ draw operator precedence table for the symbols present in the string. Using this table check the validity of the string. Show all the intermediate steps of generating an AST. **07**

OR

- Q.5 (a)** What do you mean by code optimization? Explain the five optimizing transformation with example. **07**
- (b)** Given the following grammar, left factor the grammar and generate LL(1) parse table from it. Parse the string to $| - id + id * id - |$ to check whether it is valid or not using LL(1) parsing **07**
- $S ::= A + S \mid A$
 $A ::= B * A \mid B$
 $B ::= id$
