1. Generate code for the following expression using the code generator algorithm X := (a + b) \* (c - d)
2. About the issues in the design of a code generator.
3. Generate code for the following three-address statements.

x=a[i]

y=b[x]

a[i]=y

1. Translate the arithmetic expression a + -(b + c) into quadruples, triples and indirect triples.
2. About code generator and the three primary task of code generator.
3. About the
4. Quadruples. (ii) Triples (iii) Indirect triples
5. Syntax tree for the expression a+a\*(b-c)+(b-c)\*d
6. DAG for the expression a+a\*(b-c)+(b-c)\*d. Show the steps for constructing the same.
7. About Control Sections and Program Linking
8. About Bootstrap Loader.
9. Explain how multi-pass assembler handles the following forward reference:

1 HALFSZ EQU MAXLEN/2

2 MAXLEN EQU BUFEND-BUFFER

3 PREVBT EQU BUFFER-1

4 BUFFER RESB 4096

5 BUFEND EQU \*

Assume that the starting address of first instruction is 1000H

1. About the

a) Header record

b)Text record

c)End record

d)Modification record

1. About an Absolute Loader.
2. About multi pass assembler.
3. About program relocation
4. About the PASS-1 of two pass assembler and explain with an example.
5. Generate the target address for the following machine instruction:

i) 032600H ii) 0310C303H iii) 03C300H iv) 003600H

if (B)= 006000, (Pc):003000, (X) = 000090.

1. About symbol definition statements
2. About System software. Bring out the difference between System software and Application software.
3. About instruction format and addressing modes of SIC/XE architecture with examples.
4. Define 2 Pass Assembler. Generate the object code for the following SIC/XE source program.

SUM START 4000

FIRST LDX ZERO

LDA ZERO

LOOP ADD TABLE,X

TIX COUNT

JLT LOOP

STA TOTAL

RSUB

TABLE RESW 2000

COUNT RESW 1

ZERO WORD 0

TOTAL RESW 1

END FIRST

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Mnemonic | LDA | LDX | STA | ADD | TIX | JLT | RSUB |
| Opcode | 00 | 04 | 0C | 18 | 2C | 38 | 4C |