Unique Paper Code	: 32347501
Name of Course	: B.Sc. Hons. Computer Science
Name of the Paper	: System Programming
Semester	: V
Year of Admission	: 2019 onwards

Duration: 3 Hours Maximum Marks: 75 Marks

Instructions for Candidates:

1. Attempt any **FOUR** out of **SIX** questions.
2. All questions carry equal marks.

1. Give a single regular expression that representsboth decimal numbers and hexadecimal numbers (precede with 0x prefix). Draw a transition diagram that recognizes the language described by the above regular expression. Write a lex program that converts a hexadecimal number into its equivalent binary number in a file.

2. Consider the program given below.Draw the activation tree to search a key value 10 in `arr`. Also show the contents of the activation record for each recursive call.

```
int arr[] = { 2, 3, 4, 10, 40, 55, 67 };//global
int binarySearch(int l, int r, int x)
{
    if (r >= l) {
        int mid = l + (r - l) / 2;
         if (arr[mid] == x)
            return mid;
if (arr[mid] > x)
            return binarySearch(l, mid - 1, x);
        return binarySearch(mid + 1, r, x);
    }

    return -1;
}

int main(void)
{
    int x = 10;
    int n = sizeof(arr) / sizeof(arr[0]);
    int result = binarySearch(0, n - 1, x);
    (result == -1) ?cout<< "Element is not present in array"
                   : cout<< "Element is present at index " <<result;
    return 0;
}
```

3. Consider the following SDD:

$R \rightarrow SR'\{R'.a = S.val, R.val = R'.b\}$
$R' \rightarrow * SR'_1\{R'_1.a = R'.a \times S.val, R'.b = R'_1.b\}$
$R' \rightarrow \in \{R'.b = R'.a\}$
$S \rightarrow digit\ \{S.val = digit.lexval\}$

Draw the annotated parse tree to evaluate the expression `6*7*8`. Also constructthe corresponding dependency graph. Is the above SDD L-Attributed or S-Attributed? Justify your answer.

Construct an S-attributed Syntax Directed Definition (SDD) that takes binary number as an input and produces its decimal equivalent as output.

4. Consider the following grammar:

$$X \rightarrow X + p \mid X * p \mid X - p \mid p$$

Using bottom-up parser, give the stepwise construction of parse tree for the input string p + p – p * p + p. Also show handles generated during the parse of above input string. Construct the CLR(1) parsing table for the given grammar.Identify the conflicts, if any.

5. Consider a hypothetical machine having four registers (R1, R2, S1, and S2) and two general purpose registers (P and Q) that supports load, store, move and arithmetic operations (assume the mnemonics used in instructions). Addition and subtraction operationsmust utilizeregisters R1and R2 to store their intermediate operands. Likewise, multiplication and division operationsmust utilizeregisters S1 and S2 to store their intermediate operands. Generate the machine code for the following three address instructions. Also, compute the cost of each instruction, stating any assumptionsyou make.

```
t1 = x*x
t2 = y * y
t3 = t2 * x
z  = t1 + t3
x  = z/2
y  = x- z
```

6. Consider the following section tables for the four object files, namely, a.obj, b.obj, c.obj, and d.obj. Construct the combined section table (CST) generated during the Pass I of linking process showing all intermediate steps. Also draw the layout of final executable file generated after pass II of linking process (show all the intermediate steps in CST).

Name	Size	Align
.SegA	50	8
.SegB	105	4
.SegC	200	8

Section table for a.obj

Name	Size	Align
.SegA	70	8
.SegB	315	4
.SegC	79	8

Section table for b.obj

Name	Size	Align
.SegB	150	4
.SegC	80	8
.SegD	55	16

Section table for c.obj

Name	Size	Align
.SegB	110	4
.SegD	35	16
.SegE	45	8

Section table for d.obj