

1. What is a SDD / Attribute Grammar? Give the SDD for the given grammar and construct the annotated parse tree for the strings:

(a) float x, y
 $decl \rightarrow type \text{ var-list}$
 $type \rightarrow int | float$
 $var-list \rightarrow id, var-list | id$

(b) $(34-3) * 42$
 $E \rightarrow E+T | E-T | T$
 $T \rightarrow T * F | F$
 $F \rightarrow (E) | id | num$

2. Give applications of SDT's with an example grammar and a tree.

3. Define S-attribed and L-attributed definitions, Inherited and Synthesized attributes and dependency graphs with suitable examples.

4. What is Intermediate Code? How can it be represented?

5. Construct a DAG and 3-address code for the given expressions.

(a) $((a-b) + (x+y) * (a+b)) - (a+b) * (x+y)$

(b) $x = y * z; w = p + y; y = y * z; p = w - x$

(c) $(a * b) + (c - d) * (a * b) + b$

6. How can 3-address code be represented? Give quadruples, triples and indirect triples for above (a)

(b) $-a * (b + c)$ (c) $x = -a * b + -a * b$

7. Explain 3-address code for control ^{flow} statements, switch, array, procedures and Backpatching with examples.

8. Discuss the following: (a) Access links, (b) Displays (c) Garbage collection (d) Heap Management (e) Type conversion, Type Equivalence and Type Coercion.

- 9. What is the structure of the symbol table? What are its contents? What are the data structures for implementing symbol tables? Give Symbol table organizations for Block Structured and Non Block Structured Languages.
- 10. Explain Static vs Dynamic Storage Allocation.
- 11. What is an activation record? Explain all fields in detail.
- 12. Discuss stack allocation with local & non local data.
- 13. What are the issues in the design of a code generator? Differentiate between Machine dependent and Machine Independent optimizations.
- 14. Convert the following code into 3-address code & optimize if required.

```

main ( )
{
  int a[10], i;
  for (i=0 ; i<10 ; i++)
    a[i] = i * 2
}

```

- 15. Explain briefly various parameter passing mechanisms.
- 16. What are the optimizations in Basic blocks? What is an induction variable? What is peephole optimization. Explain.
- 17. What is a Basic block? What is a flow graph? How are they constructed? Partition the given 3-address code into Basic blocks & flow graphs.

```

begin
  read x;
  if x > 0 then
    fact = 1;
    repeat
      fact = fact * x;
      x = x - 1;
    until x = 0;
  write fact;
end

```

- 18. What is data flow analysis? What are its properties? Describe in detail computation of data flow equations using live variable analysis by considering any flow graph.
- 19. Error Recovery in: Topdown/Bottom up parsing, lexical/syntax analysis.