Computer

COLLEGE OF ENGINEERING ,PUNE.

(An Autonomous Institute of Government of Maharashtra, Pune - 411005)

End Semester Examination

CT 401 – COMPILER CONSTRUCTION

Year : B.Tech
Academic Year : 2012-13
Duration: 3 hr

Branch : Computer Engg.
Date : 23 /11/2012
Max. Marks : 50

Instructions:

i) Answer all questions .
ii) Figures to the right indicate full marks .

Q.1 A) Explain the different phases in the design of a Compiler , with help of diagram and examples ? 05

B) Explain the role of finite automata in the design of compiler ? 03

Q.2 A) Consider the following grammar : 06

1. $S \rightarrow G\#$
2. $G \rightarrow E = E$
3. $G \rightarrow f$
4. $E \rightarrow T$
5. $E \rightarrow E + T$
6. $T \rightarrow f$
7. $T \rightarrow T * f$

i) Construct the LR(0) itemsets of the given grammar .
ii) Are there any shift/reduce or reduce/reduce conflicts. Specify in which state the conflict occurs and reason of conflict? Resolve the conflicts, if possible .
iii) Whether SLR(1) technique resolves the conflict? Justify your answer with reasons.

B) i) Whether the above grammar is LR(1) grammar ? 06
ii) If so, give detail analysis of conflicts and look-aheads ?
iv) Check whether the grammar is LALR(1) ? Construct the LR(1) items and parsing table for the same.

Q.3 A) Give the attribute grammar along with semantic rules for type checking of the grammar given below : 05

program → var-decls ; stmts
var-decls → var-decls ; var-decl | var-decl
var-decl → id : type-exp ;
type-exp → int | bool
stmts → stmts ; stmt |stmt
stmt → id := exp

B) Consider following translation scheme : 05

$E \rightarrow E^1$ or M E^2

{

backPatch(E^1.false, M.quad)

E.true = merge(E^1.true, E^2.true)

E.false = E^2.false

}

$E \rightarrow E^1$ and M E^2

{

backPatch(E^1.true, M.quad)

E.true = E^2.true

E.false = merge(E^1.false, E^2.false)

}

$E \rightarrow id^1$ relop id^2

{

E.true = makeList(nextQuad)

E.false = makeList(nextQuad+1)

gen(if id^1.place relop id^2.place goto _)

gen(goto _)

}

M-> ε

{

M.quad = nextQuad

}

S->if E then M^1 S^1 N else M^2 S^2

{

backPatch(E.true, M^1.quad)

backPatch(E.false, M^2.quad)

S.Next= merge(S^1.next, N.next, S^2.next)

}

N-> ε

{

gen(goto _)

}

Assume additional data if necessary.

Using this translation scheme, translate following statements to three address code:

```
if A < B or C < D then
        X = X + 1
else
        Y = Y + 1
```

Q.4 A) What is a basic block with respect to data flow analysis. Consider the following program : 08

```
while((a>b)&&(a>c))
{
   z = 3;
  if (a>0)
 {
  w = x + 1 ;
  y = x + 1 ;
  }
else
 {
  x = 1 ;
  y = 2 ;
  z = x + y ;
 }
a = x + y ;
b = a * z ;
}
```

1) Design the data flow graph for the above.
2) Which of the local and global optimization could be applied to the flow graph , considering all possible orders of applying different optimizations? Assume that a, b, c, z, w, x, y are live at entry of basic block 1.
3) Which of the local optimizations will change the output of constant propagation only when used in conjunction with other optimization.
4) Which expressions are available at the entry and exit of every basic block.
5) Which variables are live at the entry and exit of every basic block.

B) Illustrate the working of Sethi-Ullman algorithm for the following example : 05

$$x * (l - m) + m / (n - s)$$

Assume you have two available registers : r0 and r1.

OR

B) Consider a machine model with two general purpose registers and instructions shown below with their costs : 05

$R_i \leftarrow R_i$ op R_j	Cost -2
$R \leftarrow c$	Cost - 1
$R \leftarrow m$	Cost – 1
$R \leftarrow$ ind(R)	Cost – 1
$R \leftarrow$ ind(R+m)	Cost – 4
$m \leftarrow R$	Cost – 1

Now Consider the expression :

$$c * x[i] + e * (ind(f + g))$$

Generate optimal code for this machine using Aho Johnson's Algorithm?

Q5 A) Write Short Note on Meet Over Paths Solutions (MOP) in data flow analysis ? 03

B) Give the run-time organization of activation record for a single activation of a function of C language on any hypothetical machine ? 04

--- Best of Luck ----