SASTRA DEEMED TO BE UNIVERSITY	**SCHOOL OF COMPUTING** **CIA – III Exam: Nov 2024** **Course Code: CSE320** **Course Name: Compiler Design** **Duration: 90minutes** **Max. Marks: 50**

PART A

ANSWER ALL THE QUESTIONS **10 MARKS**

1. What do you understand from the code generated by a Cross Compiler. (2)
2. Give the structure of an activation record. (2)
3. When does dangling reference occur? (2)
4. Discuss the properties of an optimizing compilers. (2)
5. For the given expression, write the TAC statements and build DAG for the sequence. d=(a-b) + (a-c) + (a-c). (2)

PART B

ANSWER ALL THE QUESTIONS **25 MARKS**

6. (a) Construct CLR Parsing table for the following grammar and parse the input string "**abde**".

A → aB | Ad
B → bBC | d
C → e (10)

OR

(b) Show how the minimized DFA is obtained directly from an augmented regular expression r=(0|1) 0* (0|1) (10)

7. (a) Write the Syntax directed translation scheme to generate the TAC statement for flow of control statements in C++. Illustrate your SDT with the following snippet of code

```
res[i][j] = mergeArrays (arr1, arr2);
  for (int i = 0; i < n1 + n2; i++)
    printArray (res[i][i+2]);
```
(10)

OR

(b) (i) Elaborate the issues in the design of code generator. (5)

(ii) Write short notes on register allocation and register assignment during code generation phase. (5)

8.(a) Explain in detail the various transformations used in peephole optimization to improve the efficiency of target code. (5)

OR

(b) Describe the various Storage allocation strategies used to manage the Run-time memory allocated for the program. (5)

PART C

ANSWER ALL THE QUESTION **15 MARKS**

9. Examine the sequence of Three Address Code statements given below and perform the following tasks.

```
(1)   i := m-1              (16)   t7 := 4*i
(2)   j := n                (17)   t8 := 4*j
(3)   t1 := 4*n             (18)   t9 := a[t8]
(4)   v := a[t1]            (19)   a[t7] := t9
(5)   i := i+1              (20)   t10 := 4*j
(6)   t2 := 4*i             (21)   a[t10] := x
(7)   t3 := a[t2]           (22)   goto (5)
(8)   if t3 < v goto (5)    (23)   t11 := 4*i
(9)   j := j-1              (24)   x := a[t11]
(10)  t4 := 4*j             (25)   t12 := 4*i
(11)  t5 := a[t4]           (26)   t13 := 4*n
(12)  if t5 > v goto (9)    (27)   t14 := a[t13]
(13)  if i >= j goto (23)   (28)   a[t12] := t14
(14)  t6 := 4*i             (29)   t15 := 4*n
(15)  x := a[t6]            (30)   a[t15] := x
```

(a) The Basic blocks of Instructions. (3)

(b) Construct a flowgraph, specify the loops. (3)

(c) For each variable, record its next-use and liveliness at the end of each basic block. (3)

(d) Apply the code improving transformations wherever possible and generate the optimized TAC sequence. (6)

SASTRA DEEMED TO BE UNIVERSITY THINK MERIT | THINK TRANSPARENCY | THINK SASTRA	**SCHOOL OF COMPUTING** **CIA – III Exam: Nov 2024** **Course Code: CSE320** **Course Name: Compiler Design** **Duration: 90minutes Max. Marks: 50**
CO – RBT	

PART A

ANSWER ALL THE QUESTIONS **10 MARKS**

No	QUESTION	CO	RBT
1	What do you understand from the code generated by a Cross Compiler.	CO2	L2
2	Give the structure of an activation record.	CO4	L1
3	When does dangling reference occur?	CO3	L2
4	Discuss the properties of an optimizing compilers.	CO5	L2
5	For the given expression, write the TAC statements and build DAG for the sequence. d=(a-b) + (a-c) + (a-c).	CO4 CO5	L2

PART B

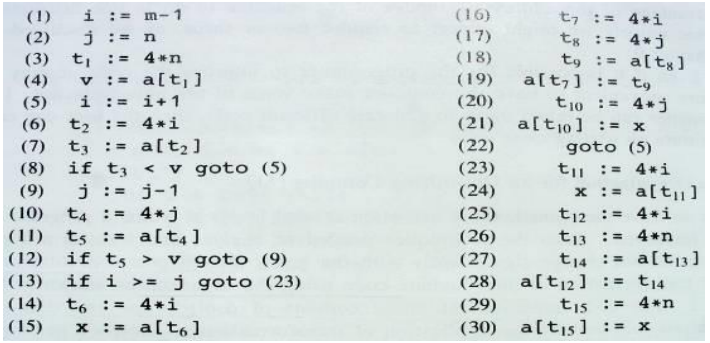
ANSWER ALL THE QUESTIONS **25 MARKS**

No	QUESTION	CO	RBT
6.a	Construct CLR Parsing table for the following grammar and parse the input string "**abde**". **A → aB | Ad** **B → bBC | d** **C → e** (10)	CO2	L3
	OR		
6.b	Show how the minimized DFA is obtained directly from an augmented regular expression r=(0|1) 0* (0|1).(10)	CO1	L3
7.a	Write the Syntax directed translation scheme to generate the TAC statement for flow of control statements in C++. Illustrate your SDT with the following snippet of code **res[i][j] = mergeArrays (arr1, arr2);** **for (int i = 0; i < n1 + n2; i++)** **printArray (res[i][i+2]);** (10)	CO4	L3
	OR		
7.b	(i) Elaborate the issues in the design of code generator.	CO6	L2
	(ii) Write short notes on register allocation and register assignment during code generation phase.	CO6	L2
8.a	Explain in detail the various transformations used in peephole optimization to improve the efficiency of target code. (5)	CO5	L2
	OR		
8.b	Describe the various Storage allocation strategies used to manage the Run-time memory allocated for the program. (5)	CO3	L2

PART C

ANSWER ALL THE QUESTION **15 MARKS**

No	QUESTION	CO	RBT
9	Examine the sequence of Three Address Code statements given below and perform the following tasks. (a) The Basic blocks of Instructions. (3) (b) Construct a flowgraph, specify the loops.(3) (c) For each variable, record its next-use and liveliness at the end of each basic block. (3) (d) Apply the code improving transformations and generate the optimized code. (6)	CO5 & CO6	L3



--- END ---

SASTRA DEEMED TO BE UNIVERSITY	**SCHOOL OF COMPUTING** **CIA – III Exam: Nov 2024** **Course Code: CSE320** **Course Name: Compiler Design** **Duration: 90minutes** **Max. Marks: 50**

PART A

ANSWER ALL THE QUESTIONS **10 MARKS**

1. ***What do you understand from the code generated by a Cross Compiler***.
   - The Cross compiler is a type of compiler that generates executable code for a platform different from the one on which the compiler is running.
   - This code is not directly executable on the machine where the compiler is running but is intended to run on the target system, which may have a different instruction set architecture (ISA) or operating system.

2. ***Give the structure of an activation record.*** ***(2)***

Return Values
Actual Parameters
Control Link
Access Link
Saved Machine Status
Local variables
Temporaries

3. ***When does dangling reference occur?*** ***(2)***
   - Activation records are data structures that are used in runtime memory management to support function calls and help manage the call / control stack.
   - When a function returns a pointer to a local variable, the activation record for that function is popped off the stack once the function returns. The local variable's memory is reclaimed, leaving the pointer dangling.

4. ***Discuss the criteria for an optimizing compiler.*** ***(2)***
   - **Correctness**: The compiler must preserve the semantics of the original program.
   - **Improve the Efficiency of target Code**: The optimization must produce code that executes faster, uses less memory, or optimally utilizes system resources.

5. ***For the given expression, write the TAC statements and build DAG for the sequence. d=(a-b) + (a-c) + (a-c).*** ***(2)***

   t1 = a – b; t2 = a – c ; t3 = a – c ; t4 = t1 + t2; t5 = t4 + t3; d = t5 (1mark)

   Correct DAG formation (1mark)

PART B

ANSWER ALL THE QUESTIONS **25 MARKS**

6. (a) Construct CLR Parsing table for the following grammar and parse the input string "abde".

A → aB | Ad

B → bBC | d

C → e ***(10)***

Augmented Grammar	1
CLOUSRE (A' → A) & Initial Set of LR items	1
Sets of LR Items using GOTO	4
CLR Parsing Table Construction	2
CLR Parsing	2
Total Marks	**10**

OR

(b) Show how the minimized DFA is obtained directly from an augmented regular expression r=(0|1) 0* (0|1) ***(10)***

Construction of Syntax Tree	1
Firstpos, Lastpos Computation	1+1=2
Followpos Computation	2
D_tran & D_states formation	3
Minimized DFA	2
Total Marks	**10**

7. (a) Write the Syntax directed translation scheme to generate the TAC statement for flow of control statements in C++. Illustrate your SDT with the following snippet of code

```
res[i][j] = mergeArrays (arr1, arr2);
  for (int i = 0; i < n1 + n2; i++)
    printArray (res[i][i+2]);
```

(10)

SDT scheme for simple if statement	2
SDT scheme for if-else statement	2
SDT scheme for while statement	2
Three Address Code Representation	4
Total Marks	**10**

OR

(b) ***(i) Elaborate the issues in the design of code generator.*** ***(5)***

5 Design Issues – Explanation of each carries 1 mark.

(ii) Write short notes on register allocation and register assignment during code generation phase. ***(5)***

Register Allocation – Definition, 2 techniques	2
Register Assignment – Definition, 3 techniques	3
Total Marks	**5**

8.(a) Explain in detail the various transformations used in peephole optimization to improve the efficiency of target code. ***(5)***

Definition of Peephole, Peephole window operations	2
5 Transformations with suitable example	3
Total Marks	**5**

OR

(b) Describe the various Storage allocation strategies used to manage the Run-time memory allocated for the program. ***(5)***

Static allocation; Stack allocation; Heap allocation	5
Definition, advantages & disadvantages of each	
Total Marks	**5**

PART C

ANSWER ALL THE QUESTION **15 MARKS**

9. Examine the sequence of Three Address Code statements given below and perform the following tasks.

(a) The Basic blocks of Instructions. ***(3)***

Identifying the Leaders	1
Formation of Basic Blocks	2
Total Marks	**3**

```
(1)  i := m-1          (16)  t7 := 4*i
(2)  j := n            (17)  t8 := 4*j
(3)  t1 := 4*n         (18)  t9 := a[t8]
(4)  v := a[t1]        (19)  a[t7] := t9
(5)  i := i+1          (20)  t10 := 4*j
(6)  t2 := 4*i         (21)  a[t10] := x
(7)  t3 := a[t2]       (22)  goto (5)
(8)  if t3 < v goto (5)(23)  t11 := 4*i
(9)  j := j-1          (24)  x := a[t11]
(10) t4 := 4*j         (25)  t12 := 4*i
(11) t5 := a[t4]       (26)  t13 := 4*n
(12) if t5 > v goto (9)(27)  t14 := a[t13]
(13) if i >= j goto (23)(28) a[t12] := t14
(14) t6 := 4*i         (29)  t15 := 4*n
(15) x := a[t6]        (30)  a[t15] := x
```

(b) Construct a flowgraph, specify the loops. ***(3)***

Flow graph nodes with predecessor node & link, loops flow	1
Representing each block with sequence of Instructions	2
Total Marks	**3**

(c) For each variable, record its next-use and liveliness at the end of each basic block. ***(3)***

List of Next use information for each variable in every block	1.5
Marking of live on Entry & Live on exit at every block	1.5
Total Marks	**3**

(d) Apply the code improving transformations wherever possible and generate the optimized TAC sequence. (6)

Local – CSE, Copy Propagation & renaming of temp.	2
Global – CSE, Copy Propagation & renaming of temp.	4
Total Marks	**15**

--- END ---