Roll No.														

PRESIDENCY UNIVERSITY
BENGALURU

SCHOOL OF ENGINEERING

TEST 1

Sem & AY: Odd Sem 2019-20 | **Date**: 01.10.2019

Course Code: CSE 214 | **Time**: 2.30 to 3.30 PM

Course Name: PRINCIPLES OF PROGRAMMING LANGUAGES | **Max Marks**: 40

Program & Sem: B.Tech (CSE) & V | **Weightage**: 20%

Instructions:

(i) Read the question properly and answer accordingly.

(ii) Question paper consists of 3 parts.

Part A [Memory Recall Questions]

Answer both the Questions. Each Question carries five marks. (2Qx5M=10M)

1. Ramesh wants to create a software in the area of system programming. He chosen a particular language for implementing that software. But the language is not suited for implementing that software. List and explain the advantages for learning different programming languages to solve the above problem.

(C.O.NO.1) [Knowledge]

2. List and explain the various programming domains. (C.O.NO.1) [Knowledge]

Part B [Thought Provoking Questions]

Answer both the Questions. Each Question carries five marks. (2Qx5M=10M)

3. Consider the following grammar (C.O.NO.1) [Comprehension]

E→E-E | E+E

E→E/E | E*E

E→(E)

E→id

i. Check whether the above grammar is ambiguous or unambiguous.

ii. Translate the above grammar in to an unambiguous grammar

4. Consider the following declaration of a two-dimensional array in C:

float a[10][20];

Assuming that the main memory is byte-addressable and that the array is stored starting from memory address 1000, predict the address of a[4][5].

(C.O.NO.2) [Comprehension]

Part C [Problem Solving Questions]

Answer both the Questions. Each Question carries ten marks. (2Qx10M=20M)

5. Consider the grammar of a Programming Language: The non-terminals are enclosed within<NT> brackets, others are terminals, and <program > is the starting symbol.

The Productions rules are given below. (C.O.NO.1) [Comprehension]

<program> → begin <stmt_list> end

<stmt_list> → <stmt>

| <stmt> ; <stmt_list>

<stmt> → <var> = <expression>

<var> → A | B | C

<expression> → <var> * <var>

| <var> / <var>

| <var>

Consider the following program. Prove that the program belongs to the grammar.
Hint: Show the derivation.

begin A = B * C ; B = C end

6. Translate the following statement into Target Code. Assume all variables are float type

res=a*b+c*120

(C.O.NO.1) [Application]

PRESIDENCY UNIVERSITY
BENGALURU

SCHOOL OF ENGINEERING

Odd Semester: 2019-2020

Course Code: CSE 214

Course Name: Principles of Programming Languages

Program &Sem: B.Tech. 5th sem

Date: 1st Oct 2019

Time: 1 Hour

Max Marks: 40

Weightage: 20%

Q.NO.	C.O.NO	Unit/Module Number/Unit /Module Title	Memory recall type [Marks allotted] Bloom's Levels			Thought provoking type [Marks allotted] Bloom's Levels			Problem Solving type [Marks allotted]			Total Marks
			K			C			A			
1	CO1	Module 1	5	L1								5
2	CO1	Module 1	5	L1								5
3	CO1	Module 1				5	L2					5
4	CO2	Module 2				5	L2					5
5	CO1	Module 1				10	L2					10
6	CO1	Module 1							10	L3		10
	Total Marks		10			40			10			40

K =Knowledge Level C = Comprehension Level, A = Application Level

Note: While setting all types of questions the general guideline is that about 60%

Of the questions must be such that even a below average students must be able to attempt, About 20% of the questions must be such that only above average students must be able to attempt and finally 20% of the questions must be such that only the bright students must be able to attempt.

[I hereby certify that All the questions are set as per the above guide lines. Mr. Prasad P S]

PRESIDENCY UNIVERSITY
BENGALURU

SCHOOL OF ENGINEERING

Odd Semester: 2019-2020

Course Code: CSE 214

Course Name: Principles of Programming Languages

Program &Sem: B.Tech, 5th sem

Date: 01-10-19

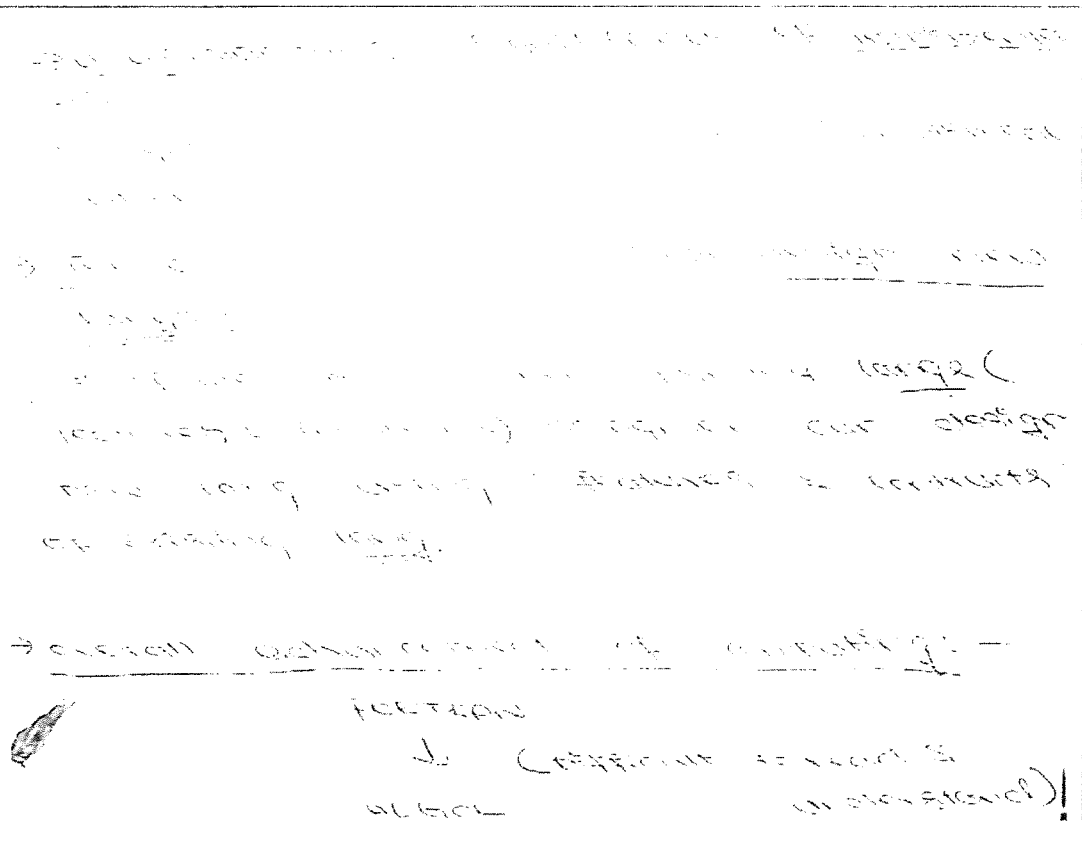
Time: 1 Hour

Max Marks: 40

Weightage: 20%

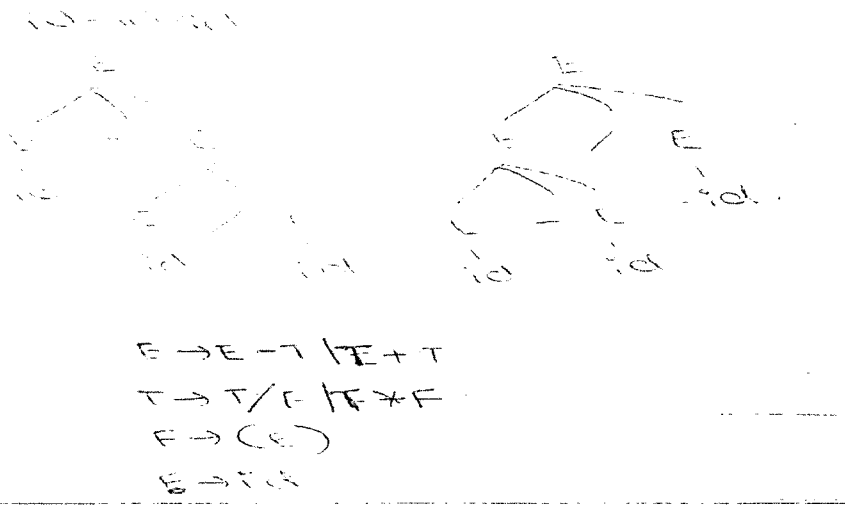
Part A (2Q x5 M = 10 Marks)

Q No	Solution	Scheme of Marking	Max Tim requi for ea Quest

1		5x1=5	10 Minut
			
2	Scientific applications Business applications	5x1=5	8 Minu

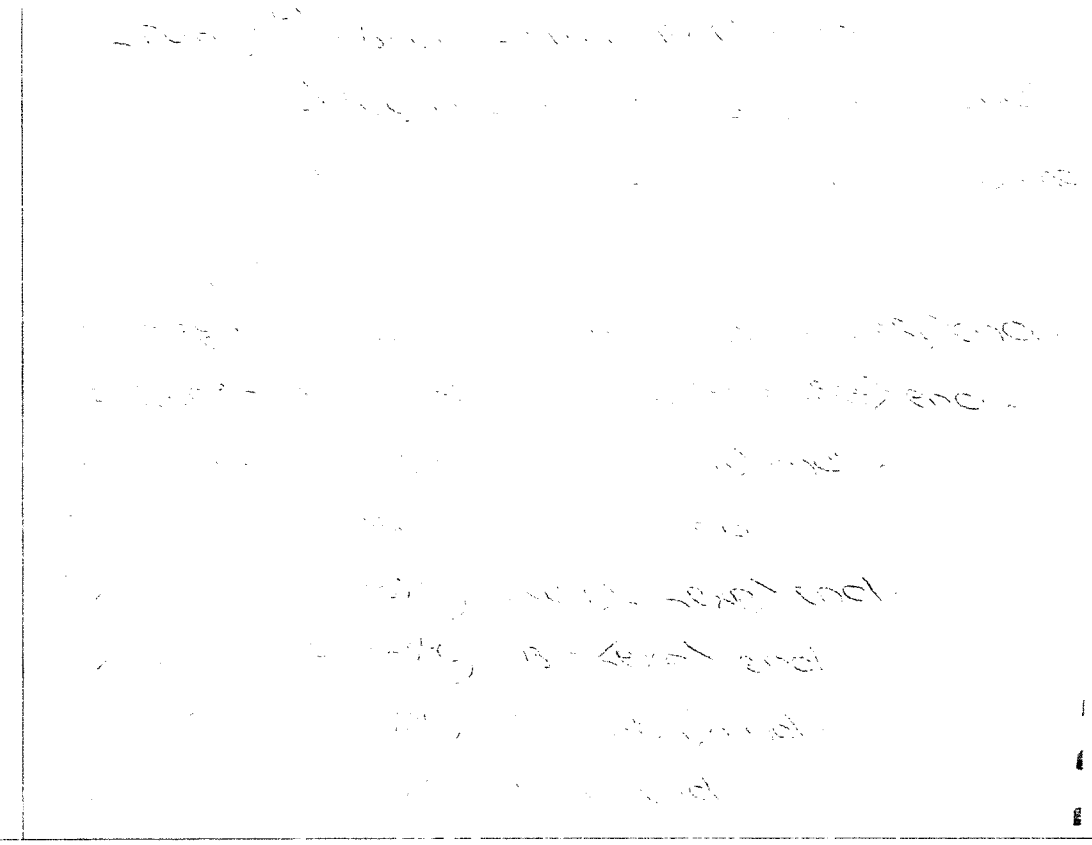
	Artificial intelligence Systems programming Web services		

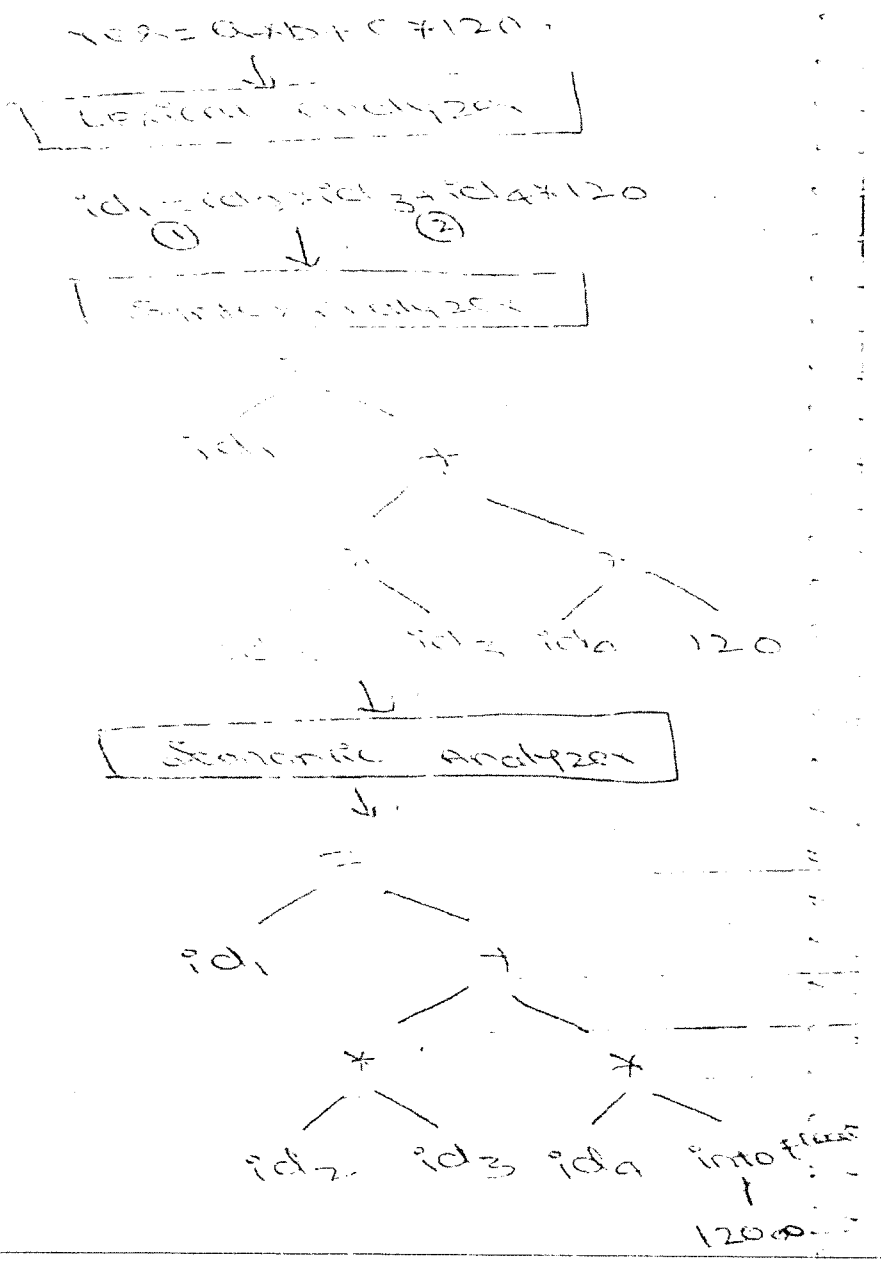
Part B (2Q x 5M = 10Marks)

Q No	Solution	Scheme of Marking	Max. Time requir for eac Questi
3	 E → E − T | E + T T → T / F | T * F F → (E) E → id	2.5x2=5	8 Minut
4	address = Base address + (row no * no of cols * width) + (j * width). a[4][5] = 1000 + (4 * 20 * 4) + (5 * 4) = 1000 + 320 + 20 a[4][5] = 1340	2M 3M	4 Minut

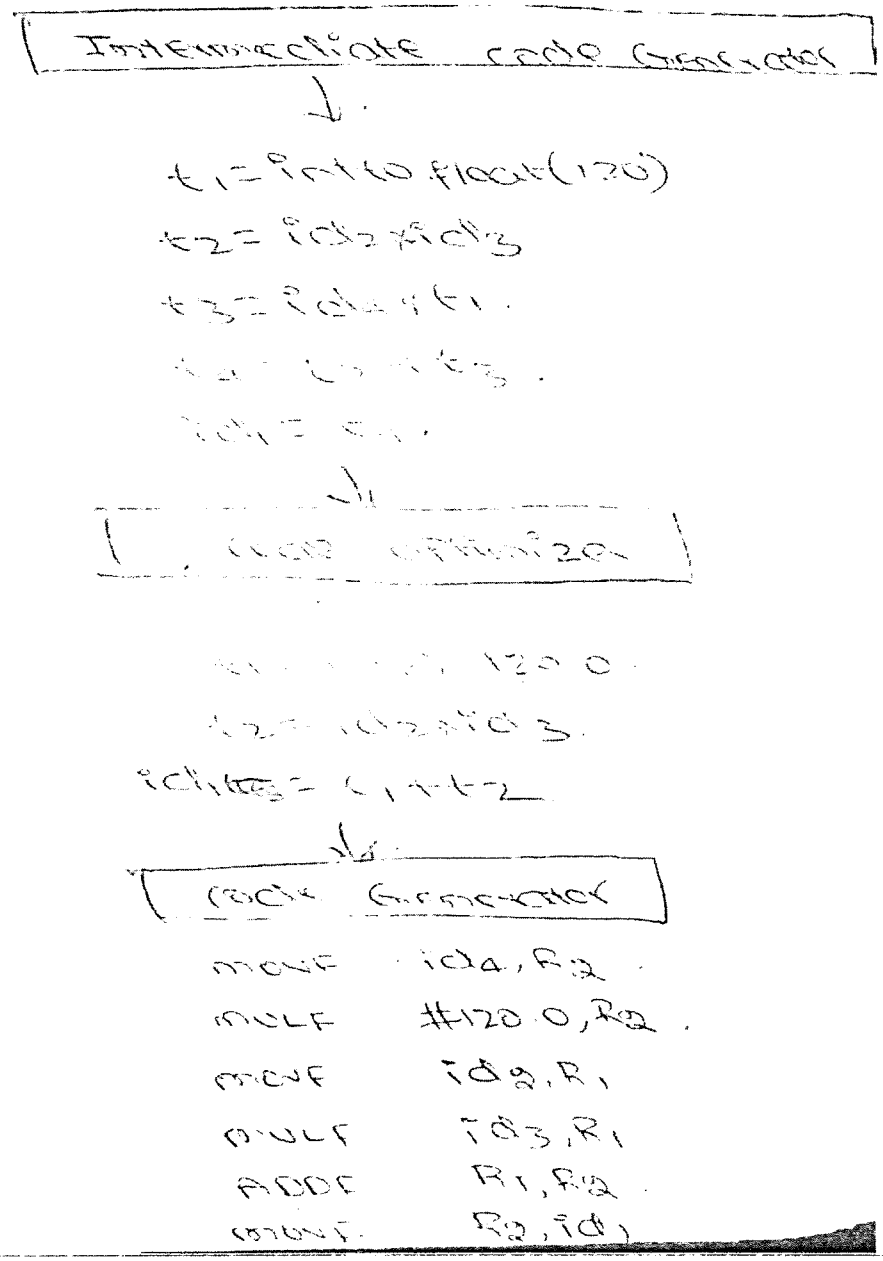
Part C (2Q x 10M = 20Marks)

Q No	Solution	Scheme of Marking	Max Tim requi for ea Quest

5		Each step carries 1M	15 minute

6		1st 3 phases 5M	15 minute

Last 3 phases 5M



Roll No.													

PRESIDENCY UNIVERSITY
BENGALURU

SCHOOL OF ENGINEERING

TEST – 2

Sem & AY: Odd Sem 2019-20 | **Date**: 19.11.2019

Course Code: CSE 214 | **Time**: 2.30 PM TO 3.30 PM

Course Name: PRINCIPLES OF PROGRAMMING LANGUAGES | **Max Marks**: 40

Program & Sem: B.Tech (CSE) & V | **Weightage**: 20%

Instructions:

(i) Answer All the Questions

Part A [Memory Recall Questions]

Answer both the Questions. Each Question carries five marks. (2Qx5M=10M)

1. Define Array. Let A denote an integer array of size 20, and let c and i be integers. Assuming that the width of an integer is 4. Generate three address code for expression c + A[i]? [5M](C.O-2) [BL -Knowledge]

2. Define record. Create an Employee record with fields EmpId, EmpName (FirstName, MiddleName, LastName), Hourly-Rate in COBOL. Explain how to access a field in COBOL? [5M](C.O-2) [BL -Knowledge]

Part B [Thought Provoking Questions]

Answer both the Questions. Each Question carries ten marks. (2Qx10M=20M)

3. Explain the classification of varibles according to their Lifetime. Consider the following program and classify the variables in the Program according to their Lifetime?

```
static int a;
int b;
int fun(int c)
{
    static int x = 0;
    int y[100] ;
    int *ptr = new (int);
    static char *p= {"G","A","T","E"};
} .
```

[10M] (C.O-3) [BL-C Comprehension]

4. Define static scope, and dynamic scope. What will be the output of the following Program with static and dynamic scoping?

```
float x;
void show( )
{
printf("%f",x);
}
void print( )
{
float x;
x=1.25;
show( );
}
int main( )
{
x=2.5;
show( );
print( );
}
```

[10M] (C.O-3) [BL-Comprehension]

Part C [Problem Solving Questions]

Answer the Question. The Question carry ten marks. **(1Qx10M=10M)**

5. Define call by value, call by value-result, and call by reference. What will be the output of the following program with call by value, call by value-result, and call by reference.

```
begin
integer x;
    procedure p(y: integer);
            begin
            x := x+1;
            y := y+4;
            print(x);
            end;
x := 0;
p(x);
print(x);
end;
```

[10M] (C.O-3) [BL-Application]

SCHOOL OF ENGINEERING

GAIN MORE KNOWLEDGE
REACH GREATER HEIGHTS

Semester: 5

Course Code: CSE 214

Course Name: Principles of Programming Languages

Date: 19-11-2019

Time: 2.30 TO 3.30 PM

Max Marks: 40

Weightage: 20%

Extract of question distribution [outcome wise & level wise]

Q.NO	C.O.NO	Unit/Module Number/Unit /Module Title	Memory recall type [Marks allotted] Bloom's Levels			Thought provoking type [Marks allotted] Bloom's Levels			Problem Solving type [Marks allotted]			Total Marks
			K			C			A			
1	2	Module 2		5								
2	2	Module 2		5								
3	3	Module 3					10					
4	3	Module 3					10					
5	3	Module 3								10		
	Total Marks			10			20			10		

K =Knowledge Level C = Comprehension Level, A = Application Level

Note: While setting all types of questions the general guideline is that about 60%

Of the questions must be such that even a below average students must be able to attempt, About 20% of the questions must be such that only above average students must be able to attempt and finally 20% of the questions must be such that only the bright students must be able to attempt.

Annexure- II: Format of Answer Scheme

SCHOOL OF ENGINEERING

GAIN MORE KNOWLEDGE
REACH GREATER HEIGHTS

SOLUTION

Semester: 5

Course Code: CSE 214

Course Name: Principles of Programming Languages

Date: 19-11-2019

Time: 2.30 TO 3.30 PM

Max Marks: 40

Weightage: 20%

Part A (2Q x 5M = 10Marks)

Q No	Solution	Scheme of Marking	Max. Time required for each Question
1	**Definition** **Three address code for expression c+a[i] is** **t 1 = i * 4** **t2 = a [t 1] or a+t1** **t 3 = C + t2**	**Definition 2M** **Three address code 3M**	**10 Min**
2	**Definition of** record 01 EMP-REC. 02 EMP-NAME. 03 FIRST PIC X(20). 03 MID PIC X(10). 03 LAST PIC X(20). 02 HOURLY-RATE PIC 99V99 Accessing field **field_name OF record_name_1 OF ... OF record_name_n** MID OF EMP_NAME OF EMP_REC	**Definition of** record 1M Syntax 2M Accessing field 2M	**10 Min**

Part B (2Q x 10M = 20 Marks)

Q No	Solution	Scheme of Marking	Max. Time required for each Question
3	Static Stack-dynamic Explicit heap-dynamic Implicit heap-dynamic	Classification of variables 5M Classification of variables in given program 5M	12 **Min**
4	Definition of static scope, and dynamic scope	Definitions 4M OUTPUT: **Static scoping: 3M**	12 **Min**

	Static scoping: 2.5 2.5 **Dynamic scoping: 2.5 1.25**	**Dynamic scoping: 3M**	

Part C (1Q x 10M = 10Marks)

Q No	Solution	Scheme of Marking	Max. Time required for each Question
5	Definitions of call by value, call by value-result, and call by reference Output call by value: 1 1 call by value-result: 1 4 call by reference: 5 5	Definitions of call by value 1M call by value-result 1M call by reference 1M Output of call by value 2M call by value-result 2M call by reference 3M	16 **Min**

Roll No												

GAIN MORE KNOWLEDGE
REACH GREATER HEIGHTS

PRESIDENCY UNIVERSITY
BENGALURU

SCHOOL OF ENGINEERING

END TERM FINAL EXAMINATION

Semester: Odd Semester: 2019 - 20 | **Date**: 28 December 2019

Course Code: CSE 214 | **Time**: 9.30 AM to 12.30 PM

Course Name: PRINCIPLES OF PROGRAMMING LANGUAGES | **Max Marks**: 80

Program & Sem: B.Tech (CSE) & V | **Weightage**: 40%

Instructions: *Read the all questions carefully and answer accordingly.*

Part A [Memory Recall Questions]

Answer all the Questions. Each Question carries 5 marks. **(4Qx5M=20M)**

[Knowledge]

1. List the various language evaluation criteria's and the characteristics that affect them. (C.O.No.1)
2. List the various categories of arrays with an example. (C.O.No.2)
3. Consider the following assignment statement
   a=a+5; Explain different possible binding times for the above statement. (C.O.No.3)
4. What is lazy Evaluation? Explain its advantages and disadvantages (C.O.No.4)

Part B [Thought Provoking Questions]

Answer all the Questions. Each Question carries 10 marks. **(3Qx10M=30M)**

[Comprehension]

5. Explain the structure of compiler. Convert the following statement to assembly code.
   position=initial+rate*60 (C.O.No.1)
6. Suppose you want to design arrays for your own imperative programming language. Illustrate how you will address the different design issues related to arrays. How it will be implemented? (C.O.No.2)
7. Explain the difference between static and dynamic scoping with an example. What will be the output of the following pseudo code when parameters are passed by reference and dynamic scoping used. (C.O.No.3)

```
a=3;
void n(x)
        {x=x*a;
         print(x);}
void m(y){
        a=1;
        a=y-a;
        n(a);
        print (a);}
void main()
        {m(a);}
```

Part C [Problem Solving Questions]

Answer all the Questions. Each Question carries 10 marks. **(3Qx10M=30M)**

[Application]

8. Create Activation Records in Runtime memory for the following program? (C.O.No.3)

```
void main()
{
int a,b,c;
c=sum(a,b);
print(c);
}
int sum(int m, int n)
{
int res;
res = m+f(n);
return(res);
}
int f(int p)
{
int i;
for(i=0;i<5;i++)
{
 p= p+i;
}
return(p):}
```

9. Explain Deep and Shallow access. Draw the memory layout for the following program with shallow access? (C.O.No.3)

Fun2(int z)	Fun1(int x,int y)	main()
{	{	{
int m=10;	if(y<=0)	int a=5,b=2;
z=z-1	return();	Fun1(a,b)
Fun1(m,z);	else	Fun2(0);
}	Fun2(y);	}
	}	

10. Explain Lambda Calculus? Describe the Syntax of Lambda Calculus and evaluate the following expression using alpha and beta reduction rules? (C.O.No.4)

(λx . (λx . + (− x 1)) x 3) 9

GAIN MORE KNOWLEDGE
REACH GREATER HEIGHTS

SCHOOL OF ENGINEERING

END TERM FINAL EXAMINATION

Extract of question distribution [outcome wise & level wise]

Q.NO.	C.O.NO (% age of CO)	Unit/Module Number/Unit /Module Title	Memory recall type [Marks allotted] Bloom's Levels	Thought provoking type [Marks allotted] Bloom's Levels	Problem Solving type [Marks allotted]	Total Marks
			K	C	A	
1	1	1	5			5
2	2	2	5			5
3	3	3	5			5
4	4	4	5			5
5	1	1		10		10
6	2	2		10		10
7	3	3		10		10
8	3	3			10	10
9	3	3			10	10
10	4	4			10	10
	Total Marks		20	20	40	80

K =Knowledge Level C = Comprehension Level, A = Application Level

Note: While setting all types of questions the general guideline is that about 60%

Of the questions must be such that even a below average students must be able to attempt, About 20% of the questions must be such that only above average students must be able to attempt and finally 20% of the questions must be such that only the bright students must be able to attempt.

I hereby certify that all the questions are set as per the above guidelines.

Faculty Signature: Dr. J. Andrews 11/12/19

Reviewer Comments:

Format of Answer Scheme

GAIN MORE KNOWLEDGE
REACH GREATER HEIGHTS

SCHOOL OF ENGINEERING

SOLUTION

Semester: Odd Sem. 2019-20
Course Code: CSE214
Course Name: Principles of Programming Languages
Program & Sem: B.Tech &5th sem

Date: 28.12.2019
Time: 3 HRS
Max Marks: 80
Weightage: 40%

Part A (4Q x 5M =20Marks)

Q No	Solution	Scheme of Marking	Max. Time required for each Question
1	Simplicity Orthogonality Syntax design data types Abstraction Type checking Exception handling Restricted aliasing	**Listing 5M**	**10 min**
2	Static Fixed stack dynamic Stack dynamic Fixed heap dynamic Heap dynamic	Definition 3M Example 2M	**10 min**
3	a) Language design time Language implementation time Compile time Load time Run time	1 1 1 1 1	**10 min**
4	Lazy Evaluation Advantages Disadvantages	**2M** **2M** **1M**	**10 min**

Part B (3Q x 10M =30 Marks)

Q No	Solution	Scheme of Marking	Max. Time required for each Question
5	Phases of compiler	Explanation 4M Translation of statement 6M	**20 min**

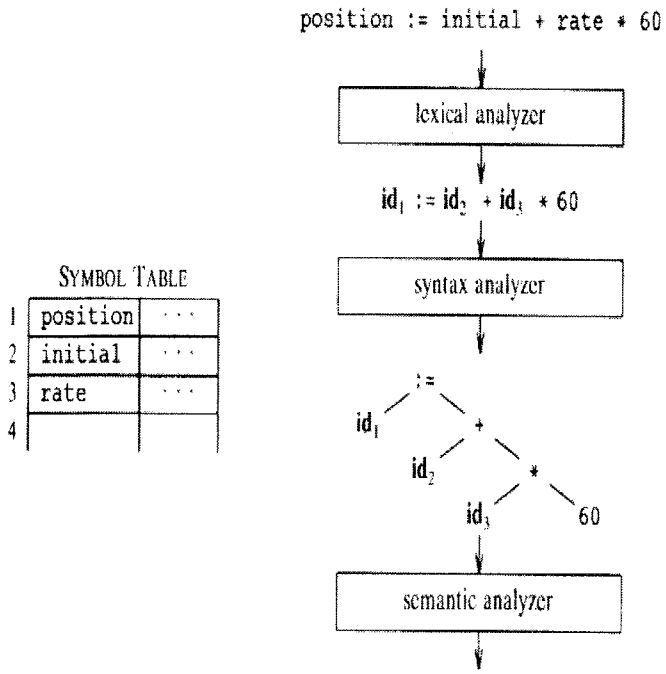


Fig. 1.10. Translation of a statement.

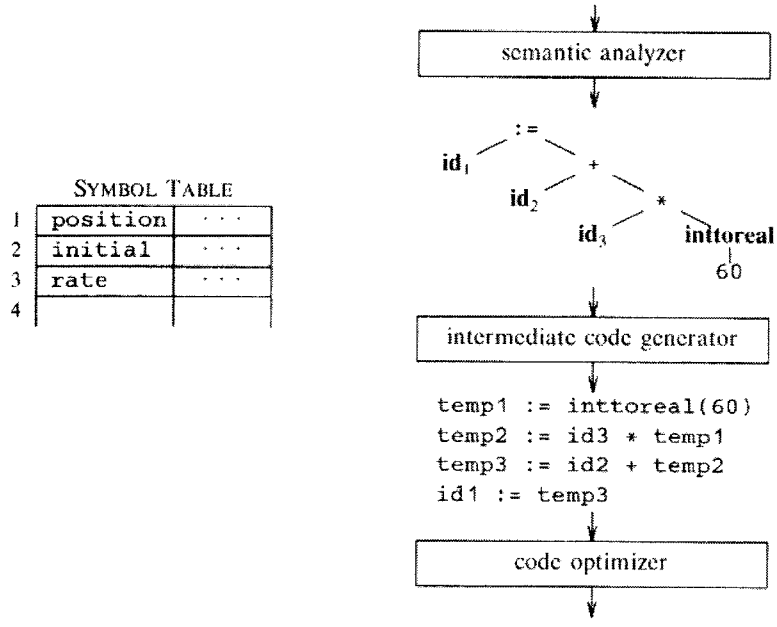


Fig. 1.10. Translation of a statement.

- **Code Optimizer**

  temp1 :=id3*60.0
  id1:=id2+temp1
- Code Generater

	MOVF id3, R2
	MULF #60.0, R2
	MOVF id2, R1
	ADDF R2, R1
	MOVF R1, id1		

6	Design issues to be addressed Categories of arrays to be designed Implementation issues		20 min

(3Q x 10M =30Marks)

Q No	Solution	Scheme of Marking	Max. Time required for each Question
7	Static scoping Check the value in the current block or function if not then search it in the outer block. Dynamic scoping Check the value in the current block or function if not then search it in the calling function and so on. Output: 4,4	Definition 4M Output 6M	**20 min**
8		Activation Record for each procedure 3M (3X3=9) Entire structure 1M	20 min

temp	**1023**
return value	**1022**
i	**1021**
p	**1020**
NIL	**1019**
1009	**1018**
return sum	**1017**
NIL	**1016**
return value	**1015**
res	**1014**
n	**1013**
m	**1012**
NIL	**1011**

1000	**1010**
return main	**1009**
NIL	**1016**
return value	**1015**
res	**1014**
n	**1013**
m	**1012**
NIL	**1011**
1000	**1010**
return main	**1009**

9	Deep Access –Activation records Shallow access				Deep and Shallow access Explanation 4M Deep and Shallow access Memory layout 3M+3M	**20 min**
10	Lambda Calculus **Function creation** – Church introduced the notation **λx.E** to denote a function in which 'x' is a formal argument and 'E' is the functional body. These functions can be of without names and single arguments. **Function application** – Church used the notation $E_1.E_2$ to denote the application of function E_1 to actual argument E_2. And all the functions are on single argument **Syntax of Lambda Calculus** Lamdba calculus includes three different types of expressions, i.e., E :: = x (variables) | E_1 E_2(function application) | λx.E (function creation)				Explanation 3M Syntax 3M Evaluation 4M	**20 min**

		Fun1(10)	Fun1(0)		
		Fun1(10)	Fun1(1)	Fun2(1)	Fun2(10)
main	main	Fun1(5)	Fun1 (2)	Fun2(2)	Fun2(10)
a	b	x	y	z	m

	Where **λx.E** is called Lambda abstraction and E is known as λ-expressions. **Alpha Reduction** Alpha reduction is very simple and it can be done without changing the meaning of a lambda expression. **λx . (λx . x) (+ 1 x) ↔ α λx . (λy . y) (+ 1 x)** For example − (λx . (λx . + (− x 1)) x 3) 9 (λx . (λy . + (− y 1)) x 3) 9 (λy . + (− y 1)) 9 3 + (− 9 1) 3 + 8 3 11 β-reduction When there are multiple terms, we can handle them as follows − (λx . (λx . + (− x 1)) x 3) 9 The inner **x** belongs to the inner **λ** and the outer x belongs to the outer one. (λx . + (− x 1)) 9 3 + (− 9 1) 3 + 8 3 = 11		