

FIFTH SEMESTER B.Tech. (CSE) DEGREE END SEMESTER EXAMINATION

NOV./DEC. 2013

DESIGN AND IMPLEMENTATION OF PROGRAMMING LANGUAGES (CSE 301)

DATE: 03-12-2013

TIME: 3 HOURS

MAX.MARKS: 50

Instructions to Candidates

- Answer **any five** full questions.

1A. Classify the following into different abstractions.

i. goto ii. file iii. struct iv. class v. int vi. procedure

1B. Write any two phases of translation.

1C. Write the equivalent BNF notation for the following EBNF:

$E \rightarrow [T] \{R\} [F]$

$P \rightarrow w\{x\}y$

1D. Given the following grammar, write the left most derivation, draw parse tree and syntax tree for the arithmetic expression, $5*(4+3)*(2+1)$

$\text{expr} \rightarrow \text{expr} + \text{term} \mid \text{term}$

$\text{term} \rightarrow \text{term} * \text{factor} \mid \text{factor}$

$\text{factor} \rightarrow (\text{expr}) \mid \text{number}$

$\text{number} \rightarrow \text{number digit} \mid \text{digit}$

$\text{digit} \rightarrow 0 \mid 1 \mid 2 \mid 3 \mid 4 \mid 5 \mid 6 \mid 7 \mid 8 \mid 9$

(3+1+2+4)

2A. Briefly explain the following:

i. Extensibility ii. Restrictability iii. Simplicity

2B. Write first and follow sets for the following grammar:

$\text{expr} \rightarrow (\text{list}) \mid a$

$\text{list} \rightarrow \text{expr} [\text{list}]$

2C. With neat sketch explain structure of a typical environment with a stack and a heap.

2D. Draw box and circle diagram at point-1 indicated by the comment. What is the output of the following C program?

<pre>#include<stdio.h> #include<conio.h> main() { int **x, *y, z=8; *y = 7; y = &z; **x = *y;</pre>	<pre>printf("%d \n", *y); **x = z; printf("%d \n", *y); **x = 6; /* point-1 */ printf("%d \n", *y); return(0); }</pre>
---	--

(3+2+2+3)

3A. Show the symbol table for the following C program at POINT-1 indicated by the comment. **(a)** using lexical scope and **(b)** using dynamic scope. What does the program print using each kind of scope rule?

#include<stdio.h> #include<conio.h> int x,y; void g(void) { x = x + 1; y = x + 1; }	void f(void) { int x; y = y + 1; x = y + 1; g(); }	main() { x = 1; y = 2; f(); /* POINT - 1 */ g(); printf("x=%d,y=%d\n",x,y); }
---	---	--

3B. Is the following procedure legal? If so what it does? If not, why not?

```
void fun(int* x){
    *x++;
}
```

3C. What is a type constructor? Briefly explain any three type constructors.

3D. Evaluate the statements `sum(cube(2), cube(3), cube(4))` and `cube(sum(2, 3, 4))` with respect to applicative and normal order of evaluation using the following functions:

int sum(int a, int b, int c){ return a+b+c; }	int cube(int s){ return s*s*s; }
---	--

(5+1+2+2)

4A. Show that the following grammar is ambiguous. How C programming language removes this ambiguity? How to remove the ambiguity without using C convention?

`stmt → if(expr) stmt [else stmt]`

4B. Write a java program to remove duplicate elements in a 1D array without using 2nd array.

4C. Explain any four different ways that a software component can be modified for reuse.

4D. Consider the following scheme definitions: `(define a 3)` `(define b `(4 7 8))` `define(c (cons a (cdr b)))`. What scheme interpreter will print when the following two lines are typed? i. `c` ii. `(cadr c)` (3+3+2+2)

5A. List and explain different formal semantic methods.

5B. Briefly explain different processor architectures.

5C. Write a scheme program to generate Fibonacci numbers up to a given limit.

5D. Given only the following axiom: `human(bob)`. What is the output of the following queries and why? i. `human(X)`. ii. `not (human (X))` (3+2+3+2)

6A. Write a Java program to multiply two matrices. Create one thread for computation of each element in the resulting matrix.

6B. Write Horn clauses to find the factorial of a number.

6C. Illustrate what happens in each step of the UNIX pipeline.

`$ls -l | grep "Aug" | sort`

(5+2+3)
