

Comprehensive Examination Question Paper
 BITS Pilani, Dubai Campus. International Academic City
 III Year SECOND Semester 2010-2011

Degree: B.E.(Hons.). Branch: C.S.

Course No : CS C362

Course Title: Programming Languages and Compiler Construction

Date: 29/05/11 Sunday Time: 12.30 p.m.- 3.30 pm Total marks: 80 Weight-age: 40%

Data provided are complete. *Closed Book.*

This question paper has 4 pages.

PART A: (write in PART A answer booklet)

Answer **all** Questions

A.1) Write a PICO LISP program to implement the following RECURRENCE RELATION: (Assume that n is ≥ 0). **[5M]**

T(n) =

- $n^2 + 3$, for $n=0,1$.
- $3T(n-2) - 2T(n-1)$ for $n > 1$.

A.2) SWI PROLOG program & Queries for a simple database application **[7 M]**

a) Store the following information in a SWI PROLOG Database.

DL_NO : Driving License No, VP_NO : Vehicle Plate No.

| PERSON | |
|--------|-------|
| DL_NO | PNAME |
| 11 | aaa |
| 12 | bbb |
| 13 | ccc |
| 14 | ddd |
| 15 | eee |

| TRANS | |
|-------|----------------|
| VP_NO | PLACE_OF_ISSUE |
| v1 | fff |
| v2 | ggg |
| v3 | hhh |
| v4 | iii |
| v5 | jjj |

| TRANS PERSON | |
|--------------|-------|
| VP_NO | DL_NO |
| v1 | 14 |
| v2 | 11 |
| v3 | 12 |
| v4 | 13 |
| v5 | 14 |

b) Answer these queries in SWI PROLOG:

- List all the vehicles owned by a given person. i.e. you must display the person name (PNAME) and VP_NO for a given person (PNAME).
- Given a Driving License No (DL_NO), list all the vehicle nos (VP_NO) and the place of issue (PLACE_OF_ISSUE) i.e. you must display the VP_NO and PLACE_OF_ISSUE.

example: For DL_NO = 14, output is

v1 fff
v5 jjj

P.T.O.

A.3) Write a JAVA program involving Class Inheritance for the credit cards. [10M]

- Create a base class **Card** that can store the following attributes [i.e. data items] **cardNo** (of type string), **custName** (of type string) and **validity** (of type integer). This class can contain a *constructor* to set the values for the attributes **cardNo**, **custName** and **validity**. This class contains another *method* to display the **cardNo**, **custName** and **validity**.

If validity =1, it displays the message, "Valid"

If validity =0, it displays the message, "Invalid"

- Create a derived class **CreditCard** that extends the base class **Card**. This class can store an attribute **categorycode** of type integer. (Assume that the *categorycode* can have a value 1 or 2 or 3 when an object is created). This class can contain a constructor to set the values for **cardNo**, **custName**, **validity** and **categorycode**. This class contains another *method* to display the **cardNo**, **custName**, **validity**, **categorycode**. It also prints **one** of the following *messages* according to the value of **categorycode**:

If *categorycode*=1, it displays the message, "Platinum".

If *categorycode*=2, it displays the message, "Gold".

If *categorycode*=3, it displays the message, "Silver".

-Create a Class **Inheritance_Testing**. This class can store the **main** function. Now you create the following three objects for the class **CreditCard**:

"12345" "AAA" 0 1

"23456" "BBB" 1 2

"34567" "CCC" 1 3

Now print all these objects along with their appropriate messages, reflecting custNo, custName, category name and validity.

A.4) Break the following program into BASIC BLOCKS and Write them. [6 M]

1. $p \leftarrow 0$
2. $w \leftarrow 0$
3. if $w \geq n$ goto 15
4. $u \leftarrow v$
5. $v \leftarrow 0$
6. if $u < n$ goto 9
7. $w \leftarrow v + 4$
8. goto 3
9. $y \leftarrow A[r]$
10. $v \leftarrow v + y$
11. if $v \leq p$ goto 13
12. $p \leftarrow v$
13. $u \leftarrow u + 8$
14. goto 6
15. finalValue = p

P.T.O

A.5) Consider a *simple assignment statement*:

$$a := (b + c) * (5 + b) - (c + d)$$

You are required to *generate CODE* for the above statement using a *simple code generation algorithm* and *tabulate* your steps in appropriate rows and columns as shown below:

| Statement(s) | Code Generated | Register Descriptor | Address Descriptor |
|--------------|----------------|---------------------|--------------------|
| | | | |
| | | | |

[5M]

A.6) What is STATIC TYPE CHECKING? Give 2 examples of STATIC TYPE CHECKING. [3M]

A.7) What is the function of *Interference Graph* in Register Allocation in code generation phase of the compiler? [2M]

A.8) Write a brief technical note on *Non-disruptive, Incremental Tracing Collectors* (type of garbage collection (GC) activity) in compiler design. [2M]

P.T.O

PART B (write in *PART B answer booklet*)
 Answer **all** Questions

- B.1) Distinguish between EARLY BINDING and LATE BINDING w.r.t. programming languages. [2 M]
 B.2) What are the notational conventions for Terminals in CFG? [2 M]
 B.3) Distinguish between **def** and **use** in LIVENESS ANALYSIS. [2 M]
 B.4) What is Reduction in Strength in CODE Optimization? Give an example. [2M]
 B.5) What is LAZY EVALUATION in Functional Programming? [2 M]

B.6) Draw the layout of a typical STACK FRAME (ACTIVATION RECORD) for a function (diagram only needed here). [5 M]

B.7) Write an ALGORITHM (input, output, logical sequence of steps) to compute the FIRST SET: FIRST(α) in top down parsing. [5 M]

(note: here α is any string of grammar symbols)

B.8) Write a technical note on the actions of the SEMANTIC ANALYSIS phase in compiler design. [5 M]

B.9) Write the meaning of the following operators for regular expressions in LEX:

- | | |
|------|-------|
| i. | [xy] |
| ii. | [x-z] |
| iii. | [^x] |
| iv. | . |
| v. | x* |

[5 M]

B.10) Using **LEX** and **YACC**, implement the following context free grammar [i.e. you have to write **LEX source** program and **YACC source** program]. [5+5 M]

- | | | |
|---------------------|--------------------------|-------------------|
| $S \rightarrow Ax$ | $S \rightarrow By$ | $S \rightarrow z$ |
| $A \rightarrow 1CB$ | $A \rightarrow 2B$ | |
| $B \rightarrow 3B$ | $B \rightarrow C$ | |
| $C \rightarrow 4$ | $C \rightarrow \epsilon$ | |

TEST II Question Paper

BITS Pilani, Dubai Campus, Dubai International Academic City

III Year SECOND SEMESTER 2010-2011

Degree: B.E. (Hons.) Branch: C.S.

Course No : CS C362 Course Title: Programming Languages and
Compiler Construction

Date: 15th May, 2011 Sunday Time: 50 minutes Total marks: 40 (20% wt.)

Data provided are complete. **OPEN Book**. This qn paper has **2 pages**.

Textbook, Reference Book and student's own handwritten class notes permitted.

Answer all Questions.

1. It is required to store all PRIME NUMBERS between 50 and 100 in a Symbol Table. Assume a HASH TABLE implementation for the Symbol Table and the hash function is defined as follows:

“**Hash Value** = (INPUT PRIME NUMBER) % 6”. Here, % means modulus (remainder after division). This hash value will be the position at which the input prime number will be inserted. (Assume that collisions can be permitted)

Draw the layout of the Symbol Table showing its contents. [6 M]

2. Construct the DAG (directed acyclic graph for higher level representation of intermediate code) for the following arithmetic expression:

$$a * (a - b) + c * (a - b) + d * (a - b) + j * (g - h) + i * (g - h) \quad [5 M]$$

3. **Translate** the following expression into *intermediate code* using STACK MANIPULATION (**lvalue**, **rvalue**, **push**, **pop**, **:=** etc) operations in an ABSTRACT STACK MACHINE :

$$a := 2 + b * ((c * d) + e) \quad [4 M]$$

4. Distinguish between Stack Pointer and Frame Pointer in activation records [2 M]

5. **Translate** the following program segment into Three Address Code (Quadruples):

```

begin
  while ( x > 5)
    begin
      if ( y > x ) then a = a + b
      y = y + 1
      x = x - 1
    end
  end

```

Assume that a, b, x , & y are all integers and $a=y=0, b=2, x=20$. [4 M]

6. Consider the following syntax directed definition for a desk calculator program:

| PRODUCTION | SEMANTIC RULES |
|--------------------------------|---------------------------------|
| $L \rightarrow E \mathbf{n}$ | print($E.val$) |
| $E \rightarrow E_1 + T$ | $E.val = E_1.val + T.val$ |
| $E \rightarrow T$ | $E.val = T.val$ |
| $T \rightarrow T_1 * F$ | $T.val = T_1.val * F.val$ |
| $T \rightarrow F$ | $T.val = F.val$ |
| $F \rightarrow (E)$ | $F.val = E.val$ |
| $F \rightarrow \mathbf{digit}$ | $F.val = \mathbf{digit}.lexval$ |

Here \mathbf{n} denotes newline.

Construct an *annotated* PARSE TREE for the following input expression:

$(2 * (3 + 4) * (5 + 6)) \mathbf{n}$ [6 M]

7. Find the NULLABLE, FIRST and FOLLOW sets for the following CFG and then Construct the PREDICTIVE PARSING TABLE: [2+4+4+6 M]

- $S \rightarrow AS$
- $A \rightarrow BC$
- $A \rightarrow D$
- $B \rightarrow \epsilon$
- $C \rightarrow \epsilon$
- $C \rightarrow EF$
- $E \rightarrow \epsilon$
- $E \rightarrow 2$
- $B \rightarrow 4$
- $D \rightarrow 3$
- $F \rightarrow 5$

Note: Here, ϵ indicates null and $\$$ indicates END OF INPUT.

TEST I Question Paper

BITS Pilani, Dubai Campus, Dubai International Academic City

III Year SECOND SEMESTER 2010-2011

Degree: B.E. (Hons.) Branch: C.S.

Course No : CS C362 Course Title: Programming Languages and
Compiler Construction

Date: 27 Mar., 2011 Sunday Time: 50 minutes Total marks: 50 Weightage: 25%
Data provided are complete. **Closed Book**. This qn paper has **2 pages**.

Answer all Questions.

1. Write a SWI-PROLOG program to implement the following RECURRENCE RELATION; Assume that n is ≥ 0 .

$T(n) =$

- $2n^2 + 3n - 10$, for $n=0,1$ or 2
- $T(n-1) - T(n-2) + 2T(n-3)$ for $n > 2$.

Write the output for the following queries/goals:

i) $t(3, X)$.

ii) $t(4, Y)$.

[7+3 M]

2. Give an example context free grammar for each of the following categories,
a) Elimination of *left recursion* b) Perform *Left factoring*

[5M]

3. Consider the context free grammar given below

$E \rightarrow E + E \mid E * E \mid (E) \mid id$

Show the shift-reduce parsing steps in the form of a table

for input sentence: $id_1 * id_2 + id_3 * id_4$

[10M]

| Stack | Input | Action |
|-------|-------|--------|
| | | |

4. Write LEX Source [PROGRAM] to recognize any INTEGER CONSTANT as per C Language convention and display appropriate message on the screen [standard output].

[5M]

INPUT(s)

-10

+10

100

ABCD

OUTPUT

VALID INTEGER CONSTANT

VALID INTEGER CONSTANT

VALID INTEGER CONSTANT

INVALID DATA in INPUT

P.T.O

5. Write YACC specification [source program as per YACC format] for the following Context Free Grammar: [10M]

$$S \rightarrow AAS \mid AA \mid 0A1 \mid 01 \mid 0B1$$

$$A \rightarrow 0A1 \mid 01 \mid 0B1$$

$$B \rightarrow B1 \mid 1$$

6. Write the OUTPUT of the following C Program: [10 M]
/* Parameter Passing */

```
#include <stdio.h>
main ()
{
    void e (int *we, int *gg);
    int x[10], i;
    int n = 4;
    for (i = 9; i >= 0; i -= 1)
    {
        x[i] = n + 1;
        e (&x[i], &n);
        n = n + 10;
    }
}

void
e (int *we, int *gg)
{
    int m, z;
    m = *we + 2;
    z = *gg + 2 + (2 * m);
    m += z;
    printf (" z = %d m= %d \n", z, m);
}
```

=====

QUIZ II *bet A*

Course No : CS C362 Course Title: Prog. Lang. & Comp. Cons.

Date: 20/4/11 Wednesday Time: 20 min. Total marks: 15

Weightage: 03% Venue : As per seating arrangement **Closed Book.**

This *question paper* has 3 *pages* Data provided are complete
Use **Back Page** for **rough work only** (*this back page will not be evaluated*)

IDNO:

Name:

Write answers in the space provided in question paper. Answer all questions.

1. Distinguish between **Class** and **Interface** in Java? [2 M]

2. What is meant by **Abstract Class** in Java? [1M]

3. Write the output of the following Java program [5 M]

```
public class Test {  
    public static void main (String args[]) {  
        int a = 20;  
  
        for (int i = 1; i <= 5; i++ ) {  
            a = (21*a + 3) % 100;  
            System.out.println(a);  
        }  
    }  
}
```

P.T.O

QUIZ II *set A*

Course No : CS C362 Course Title: Prog. Lang. & Comp. Cons.

Date: 20/4/11 Wednesday Time: 20 min. Total marks: 15

Weightage: 03% Venue : As per seating arrangement **Closed Book**.

This *question paper* has 3 *pages* Data provided are complete

Use Back Page for rough work only (this back page will not be evaluated)

IDNO:

Name:

4. What is meant by **Overriding** in Java?

[2 M]

5. A common use of polymorphism in OOP occurs when a _____ reference is used to refer to a _____ object. [2M]

6. Briefly explain the working of the Java code shown below?

[3M]

```
class Info {  
    String weblink, filename;  
  
    public Info( String w, String f){  
        weblink = w;filename = f;  
    }  
  
    public void display() {  
        System.out.print(weblink + " " + filename );  
    }  
}
```

QUIZ I

Course No : CS C362 Course Title: Prog. Lang. & Comp. Cons.

Date: 02/03/11 Wednesday Time: 20 min. Total marks: 16

Weightage: 08% Venue : As per seating arrangement **Closed Book**.

This *question paper* has 2 pages Data provided are complete

Use **Back Page** for *rough work only* (this back page will not be evaluated)

IDNO:

Name:

Write answers in the space provided in question paper. Answer all questions.

1. Consider the following PICO LISP Program:

```
/* recul.1 */  
(de recul (n)  
  (  
    cond ( ( or (= n 0) (= n 1) ) (+ (* 2 n) 1) )  
          ( t (* 2 (recul (- n 1) ) ) ) )  
  )  
)  
/* ----- */
```

Write the **result (output)** of each of the following LISP Expressions for the above code:

| INPUT | OUTPUT (write answer in this column only) |
|-----------|--|
| (recul 2) | |
| (recul 4) | |
| (recul 7) | |

[3 M]

2. Distinguish between IMPERATIVE and DECLARATIVE Programming paradigm?

[2M]

3. Describe how PROFILING works?

[2 M]

4. Distinguish between the working of INTERPRETER and COMPILER.

[2 M]

BITS Pilani, Dubai Campus, Academic City, Dubai.

III Year Second Semester 2010-2011

Degree: B.E. Hons. Branch: C.S.

QUIZ I

Course No : CS C362 Course Title: Prog. Lang. & Comp. Cons.

Date: 02/03/11 Wednesday Time: 20 min. Total marks: 16

Weightage: 08% Venue : As per seating arrangement **Closed Book**.

This *question paper* has 2 pages Data provided are complete

Use **Back Page** for **rough work** only (*this back page will not be evaluated*)

IDNO:

Name:

5. Mention any two properties for BINDING during RUN TIME.

[2 M]

6. Distinguish between INTERNAL symbols and EXTERNAL symbols in PICO LISP?

[2 M]

7. Write a PICO LISP program to calculate the nPr (Permutation of r elements taken from a given set of n elements) by the following formula. $n!$ stands for factorial of n . (Assume that n is > 0 and $r \leq n$).

$$nPr = \frac{n!}{(n-r)!}$$

[3M]