B

00000CS403121902

Pages: 4

Reg No.:_____ Name:_____

APJ ABDUL KALAM TECHNOLOGICAL UNIVERSITY

Seventh Semester B.Tech Degree (S, FE) Examination January 2023 (2015 Scheme)

Course Code: CS403

Course Name: PROGRAMMING PARADIGMS

Max. Marks: 100 Duration: 3 Hours

PART A

Answer all questions, each carries 4 marks. Marks

1 Why does the use of dynamic scoping imply the need for run-time type checking? (4)

2 Consider the following (erroneous) program in C: (4)

```
void foo( ) {
    int i;
    printf("%d", i++);
}
int main( ) {
    int j;
    for (j = 1; j <= 10); j++)    foo( );
}
```

Local variable i in subroutine foo is never initialized. What will be the output of this program. How the output differs in the following cases, justify your answers:

a) int i=0;

b) static int i=0;

3 Differentiate between a value model of variables and a reference model of variables. Why is the distinction important? (4)

4 What is a subroutine closure? What is it used for? How is it implemented? (4)

5 With appropriate example demonstrate lazy Evaluation (4)

6 What are facts, rules and queries? (4)

7 Analyse greedy and minimal matches with an example. (4)

8 a) What does the following code snippet in Perl 5 prints? (4)

```
{   package Integer;
    sub new {
```

```
        my $class = shift;
        my $self = {};
        bless($self, $class);
        $self->{val} = (shift || 0);
        return self;
    }
    sub set {
        my $self = shift;
        $self->{val} = shift;
    }
    sub get {
        my $self = shift;
        return $self->{val};
    }
}
$c1 = Integer->new(2)            # Integer::new("Integer", 2)
$c2 = new Integer(3);            # alternative syntax
$c3 = new Integer;               # no initial value specified
print $c1->get," ", $c2->get, " ", $c3->get, " ", "\n";
$c1->set(4);  $c2->set(5);  $c3->set(6);
print $c1->get," ", $c2->get, " ", $c3->get, " ", "\n";
```

b) What does the following code snippet in Tcl prints?

```
proc bar { } {
    upvar i  j          ;# j is local name for caller's i
    puts "$j"
    uplevel 2 {puts [expr $a + $b] }
        # execute 'puts' two scopes up the dynamic chain
}
proc foo { i } {
    bar
}
set a 1;    set b 2;    foo 5
```

9 What is symbolic debugging? (4)

10 With an example, demonstrate reflection in Java. (4)

PART B

Answer any two full questions, each carries 9 marks.

11 a) Explain the different storage management in C. (6)

b) How a recursive program is running, show the process of recursive call with an example. (3)

12 a) How static and dynamic type checking differs, explain with examples. (7)

b) Explain the meaning of the following C declarations: (2)

(i) double (*c[n])( );

(ii) double (*b)[n];

(iii) double *a[n];

13 a) Explain the different notions of binding times. (6)

b) Differentiate between records and variant records. Give an example illustrating the difference. (3)

PART C

Answer any two full questions, each carries 9 marks.

14 Explain the typical calling sequence of caller, prologue and epilogue. (9)

15 a) With the help of an example, differentiate between let, let* and letrec functions. (6)

b) What is Lambda Calculus? How it is related to programming languages? (3)

16 a) Write a program in Java to demonstrate exception handling, and explain the key concepts. (3)

b) Consider the following program in Scheme: (6)

```
(define A
  (lambda ( )
    (let*  ( ( x  2)
          (C  (lambda  (P)
                (let  ( (x  4) )
                  (P) ) ) )
          (D  (lambda  ( )
                x) )
          (B  (lambda  ( )
                (let  ( (x  3) )
                  (C  D) ) ) ) )
      (B) ) ) ) )
```

(i) What does this program print?

(ii) What would it print if Scheme used dynamic scope with shallow binding?

(iii) What would it print if Scheme used dynamic scope with deep binding?

PART D

Answer any two full questions, each carries 12 marks.

17 a) Summarize on the characteristics of scripting languages. (6)

b) Illustrate public, private and protected access members with an example (6)

18 a) Summarize the architecture of Java virtual machine. (5)

b) With a neat diagram, explain the architecture of threads. (7)

19 a) Illustrate the significance of variable interpolation and capture with an example. (3)

b) With sufficient examples, explain the purpose and working of semaphores. (6)

c) Explain constructor overloading with an example. (3)

****