



## Cs508 finalterm solved mcqs

Modern Programming Languages (Virtual University of Pakistan)

# cs508 modern programming languages

## Final Papers & Quizzes

One difference LISP and PROLOG is

1. AI
2. **Puzzle**
3. Game
4. All f the given

Variable of ..... Is not an object of java

1. Primitive
2. Reference
3. **Integer type**
4. Both reference and Primitive

In ..... Every class in Java is part is overloaded

1. **C++..?**
2. Java
3. C#
4. C# and java

C# and C++ have ..... size

1. Same
2. Different
3. **Distinct**
4. None of the given

\_\_\_\_\_ compile into machine independent language, independent code which run in a managed execution environment.

1. **C#**
2. Java
3. C++
4. Ada

\_\_\_\_\_ compile initially to an intermediate. Which can be run by interpretation or just in time compilation or an appropriate virtual machine?

1. C++ and C#
2. **C# and Java**
3. Java and C++
4. Ada , C++

In \_\_\_\_\_ the relationship between a get and set method is inherited, while in \_\_\_\_\_ it has to be maintained.

1. Java , C++
2. C++ , C#
3. Ada , Java
4. **C# , Java or C++**

How many modes for the source code are in C#?

1. One
2. **Two**
3. Three
4. Four

Which chaining type is used by Prolog?

1. **Backward**
2. Forward
3. Up
4. Horizontal
5. Maps in SNOBOL are also available in \_\_\_\_\_(C, C++, MATLAB, Prolog)

In C# struct are used as .....**parameters**.....

Javascript ...**related**.. from java language

In ... **JavaScript** ..... we have to address the client side compatibility issue.

\_\_\_\_\_exception inherits from exception class and \_\_\_\_\_ exception is anywhere in the program.

1. Java , C#
2. C++ , C# c
3. C# , Java
4. **Java ,C++ ..?**

In C# Managed code is executed under the control of

1. CLR
2. **CRL**

In \_\_\_\_\_ Boolean expression is convertible into integer type.

1. C#
2. C++
3. **JAVA..?**
4. Ada

PHP syntax looks like \_\_\_\_\_

1. ASP syntax
2. **C/C++ syntax**

For narrowing conversion which type conversion is appropriate?

1. Implicit Conversion
2. **Explicit Conversion**

In Java we can make pointer of \_\_\_\_\_.

1. Any type
2. **Reference type only**

The concept of package in Java is similar to \_\_\_\_\_

1. **Assembly in C#**
2. Global class in C#

The concept of C# jagged array is similar in function to \_\_\_\_\_

1. C++ rectangular Array
2. **Java Array of Array**
3. Java pointer Array
4. C# Array

Java has a String class which is not exactly an array of

1. **Char**
2. Elements
3. Indices
4. Long

**How many string operators are in PHP?**

1. 2
2. 3
3. 4
4. 10

**C# code when compiled is converted into \_\_\_\_\_ code.**

1. MSIL
2. MISL

**In C# \_\_\_\_\_ can be inherited from other class but no inheritance from it. On the other hand \_\_\_\_\_ cannot be inherited not inheritance is possible from it is possible.**

1. Private class , public class
2. Sealed class ,struct in C#
3. struct in C# , Sealed class
4. public class , inherited class

**The concept of sealed class in C# is similar to \_\_\_\_\_**

1. Struct in C#,
2. Struct in C++
3. Abstract class in C#
4. None of the given

**Enumeration type in C# may take any type of \_\_\_\_\_ in contrast to C++ where it take only \_\_\_\_\_**

- a. Numeric value, integer value
- b. Value type value, numeric value
- c. Primitive type value, reference type value
- d. Value type value, reference type value

**The IS operator in C# is used for \_\_\_\_\_**

1. Run time type conversion
2. Run time type checking
3. Run time type casting
4. None of the given..( The 'is' operator supports run time type information)

**PHP is a \_\_\_\_\_ typed language.**

1. Strongly
2. Dynamic

3. Static
4. **None of Given**

\_\_\_\_\_ is more strongly typed language than \_\_\_\_\_

1. C++, C#
2. C#, C++
3. C++, Java
4. **None of the given**

Anything starting with Capital or underscore is a variable in \_\_\_\_\_

1. **Prolog,**
2. Lisp,
3. Cobol

In C# the if statement condition is an/a \_\_\_\_\_

1. Arithmetic expression
2. **Boolean expression**
3. Numeric expression
4. Both Numeric expression and Boolean expression

In C# the value type and reference type variable are interconvert able through \_\_\_\_\_ concept.

1. Tagged type
2. **Boxing**
3. Interfaces
4. None of the given

C# support only \_\_\_\_\_ inheritance and it achieve \_\_\_\_\_ inheritance through the concept of interfaces.

1. Multiple, multiple
2. **Single, multiple**

Managed or safe code in \_\_\_\_\_ is executed under the control of common language runtime (CLR) with automatic garbage collection, no explicit memory allocation and de allocation and no explicit destructor.

1. C++
2. Java

3. Ada and C++

4. C#

PHP supported all major Databases including\_\_\_\_\_.

1. ODBC

2. Oracle

3. SQL Server

4. None Of given...?

Dynamic binding occurs at:

1. Compile Time

2. Design Time

3. Link Time

4. Run Time

Prolog language falls under the domain of:

1. Scientific Applications

2. Business Applications

3. Special Purpose Languages

4. None of these

4. In Prolog we specify solution and not problem

Object in java script can be accessed through .....

1. Reference pointer method.....not in handouts....using dot notation is in handouts

2. None of the above

3. **Capital letter or underscore in PROLOG** variable

4. **Variable in PROLOG .... Are any value is associated that can not be changed.**

For Recursion it is necessary that a language

1. Dynamic

2. Static

3. Both dynamic and static

4. Stack

First language that provided the concept of Pointer data type was \_\_\_\_\_.

1. COBOL
2. LISP
3. **PL/1**
4. JAVA

A ... **implicit..** is a default mechanism for specify types of variable?

A **reserved word** is a special word that cannot be used as a user-defined name

Variable use in VB without declaring decrease reliability and increase .....

1. Readably
2. **Writeable**
3. Cost
4. compile time

Dot operator in a .... SNOBOL?

1. **Reference pointer**
2. Unary pointer
3. Class pointer
4. Binary pointer

The GOTO statement in SNOBOL is .....

1. **Explicit**
2. Punter method
3. Implementation
4. An Indirect Reference

Ada has ... do while loop just like C++

1. Also
2. **No**
3. Defective
4. None of the above

We use **tagged type** in Ada.....

The last value execution in the ....LISP is the return value

1. Atom
2. **Object**

Dotimes loop of LIPS similar working to ada

1. **For**
2. Switch loop
3. Do while
4. While

Control structure is a .....**GOTO**/jump..:)..... statement

..... **anonymous** ..... is place holder value that is not required

Indexes of arrays in Ada DO NOT starts from

- **0**
- 1
- 2
- 3

The keyword unsafe is used while dealing with \_\_\_\_\_

1. Loop
2. Arrays
3. **Pointers**
4. Classes

Tasks in \_\_\_\_\_ are processes and cannot share data but thread in \_\_\_\_\_ do it.

1. C# , C++
2. C++ , Ada
3. Java , C#
4. **Ada , Java..?**

Java code when compiled is converted into \_\_\_\_\_ code.

1. Bit code
2. **Byte code**
3. Kbytes code
4. Giga byte Code

If you want an argument to a function to always be passed by reference, you can prepend

1. Percentage sign (%) to the argument name in the function definition
2. Dollar sign (\$) to the argument name in the function definition
3. **An ampersand (&) to the argument name in the function definition**
4. Tilled sign (~) to the argument name in the function definition

**Ada pointers are called Access types.**

- a) **TRUE**
- b) FALSE

**The dangling pointer problem is partially alleviated by Ada design.**

1. TRUE
2. **FALSE**

**Which statement is wrong about Static variables?**

1. They are bound to memory cells before the execution of program
2. They use to direct access the memory.
3. **There is run time overhead of allocation and de-allocation of memory.**
4. Storage can't be shared among variables

**Following statement returns the union of the two list in LISP.**

1. >(L1 UNION L2)
2. **>(union L1 L2)**
3. >Union L1
4. >(L1 union L2)

**Following is the correct structure of a "block" in Ada.**

1. declare           -- declare section optional  
   statements  
   begin  
   declarations  
  
   exception       -- exception section optional

handlers  
end;

2. **declare** **-- declare section optional**  
**declarations**  
**begin**  
**statements**

**exception** **-- exception section optional**  
**handlers**  
**end;**

3. declare -- declare section optional  
statements  
begin  
declarations  
handlers  
exception -- exception section optional  
end;
4. declare -- declare section optional  
statements  
declarations  
begin  
statements  
exception -- exception section optional  
handlers  
end;

In LISP, \_\_\_\_\_ is the main tool used for iteration.

1. **Recursion**
2. For Loop
3. While Loop
4. Do-While Loop

\_\_\_\_\_ represents the class of languages from functional paradigm.

1. **LISP**
2. PROLOG
3. Ada
4. C++

In LISP, following statement returns the difference of the two lists.

1. (Difference L1 L2)
2. **(set-difference L1 L2)**
3. (L1 Difference L2)
4. (L1 difference L2)

Lists can be constructed with the help of three basic functions which are \_\_\_\_\_.

1. scheme, common lisp and cons
2. cons, append and scheme
3. **cons, list and append**
4. list, atoms and append

Block statement in Ada is very different to a block in C.

1. True
2. **False**

Two important versions of LISP are \_\_\_\_\_.

1. Scheme and Atoms
2. **Scheme and Common Lisp**
3. Common Lisp and defacto
4. List and Common Lisp

Following is the correct syntax of 'for' statement in Ada.

1. **for variable in low\_value .. high\_value loop**  
    **-- Loop body goes here**  
    **end loop;**
2. for variable in low\_value .. high\_value loop  
    -- Loop body goes here  
end;
3. begin;  
for variable in low\_value .. high\_value loop  
    -- Loop body goes here  
end loop;
4. begin loop;  
for variable in low\_value .. high\_value loop  
    -- Loop body goes here  
end loop;

In \_\_\_\_\_, Enumeration type can also be used as indexes in arrays.

1. **Ada**
2. C
3. C++
4. LISP

SNOBOL was designed for \_\_\_\_\_ purpose.

1. **String manipulation**
2. AI
3. Business
4. Scientific

Computer architecture has a major influence on the design of programming language.

1. **True**

2. False

Too much simplicity in language design can also cause problems.

1. **True**
2. False

COBOL is mainly designed for \_\_\_\_\_.

1. Scientific experiments
2. **Business application**
3. AI applications
4. Publishing and writing algorithm

In SONOBAL binary operators have atleast \_\_\_\_\_ spaces

- 1
- **2**
- 3
- 4

In SONOBOL can the size of the array be determined at run time.

1. **Yes**
2. No

\_\_\_\_\_ is an important feature of Ada language used in embedded systems and operating systems. It is used in managing parallel threads of controls.

1. Controls
2. **Tasking**
3. classes

First electronic computer was \_\_\_\_\_

1. **ENIAC**
2. Babbage analytical engine
3. Intel 386
4. IBM x86

It is always better to find out the mistakes in the code at \_\_\_\_\_.

1. Runtime
2. **Compile time**
3. Link time
4. None of given option

Too much Orthogonality reduces Readability.

1. **True**
2. False

COBOL was the first language that brings the concept of \_\_\_\_\_.

1. **Records**
2. Structure
3. Object
4. Variable

Learning the implementation details of a language helps in the efficient use of that language.

1. **True**
2. False

We can have nested procedures in \_\_\_\_\_ language.

1. C++
2. **Ada**
3. C
4. PROLOG

One of the major design goals of Ada was

1. **to help developers to code the software in lesser time**
2. to make it difficult for developers to make mistakes
3. to help develop an IDE for Ada
4. to make testers to test Ada application easily

SONOBOL is case \_\_\_\_\_

1. Sensitive
2. **In-sensitive**
3. Super sensitive
4. Not sensitive

The main design goals of Ada were EXCEPT

1. Program reliability and maintenance
4. Design language for DoD
5. Programming is a human activity therefore avoid unintentional mistakes
6. **Designing a portable language**

The space operator has \_\_\_\_\_ precedence than arithmetic operators.

1. Higher
2. **Lower**

Pattern . Variable Upon successful completion of pattern matching, the substring matched by the pattern is assigned to the variable as \_\_\_\_\_

1. **Value**
2. String
3. Integer
4. Real numbers

A language evaluation criteria includes following factors EXCEPT

1. Readability
2. Writability
3. Portability
4. **Modularity**

The variable name has profound effect on \_\_\_\_\_.

1. Readability
2. Writability
3. Orthogonality
4. **Portability**

SIZE function in SONOBOL language is used to return the size of \_\_\_\_\_

1. **String**
2. Operator
3. Variable

4. Keyword

Indirect referencing in SONOBOL is same as of \_\_\_\_\_ in C.

1. Arrays
2. Pointers
3. Aliasing
4. Stack

Following factors influences a portable language design EXCEPT

1. Computer architecture
2. Readability
3. Programmer's time
4. Windows XP

The more is the Simplicity of a language, the more it will always be readable.

1. True
2. False

+ Sign is used for \_\_\_\_\_ in SONOBOL.

1. Line Continuation
2. Line Break

Which of the following is an incorrect option from the following statements regarding 'objectives of learning new languages'?

1. Help to compare different languages.
2. Help in transition from one language to other language.
3. Help in understanding the language piracy policy
4. Help to choose a language for development of a certain application.

\_\_\_\_\_ was the first object oriented language.

1. COBOL
2. LISP
3. JAVA

#### 4. SIMULA

In SONOBOL 2 spaces can be used, the purpose of 1st space is for \_\_\_\_\_ and 2nd for \_\_\_\_\_

1. Pattern matching, String Concatenation
2. String concatenation, Pattern matching

LISP used in \_\_\_\_\_ and \_\_\_\_\_ (functional paradigm, AI)

First arguments in LISP is \_\_\_\_\_ symbol..? (Atom, argument, integer, LIST)

Maps in SNOBOL are also available in \_\_\_\_\_ (C, C++, MATLAB, Prolog)

We use \_\_\_\_\_ indirect referencing operator in SNOBOL (Binary “.”, Binary \$, Unary “.”, Unary \$)

Elementary types are also called \_\_\_\_\_ in ADA (Static, user defines, builtin)

ADA is a \_\_\_\_\_ typed language (partially strongly)

\_\_\_\_\_ has elaborated exception handling (ADA, C++, JAVA, COBOL)

\_\_\_\_\_ has a powerful set of operators but poor type checking (C, C++, ADA, LISP)

\_\_\_\_\_ and \_\_\_\_\_ have declaration paradigm (Prolog, SQL)

1<sup>st</sup> machine independent language is \_\_\_\_\_ ALGOL

To make necessary arguments at run time error is called \_\_\_\_\_ (exception handling)

Language with wide domain of application has higher \_\_\_\_\_ (generality)

\_\_\_\_\_ has distributed computing architecture (CORBA)

Readability has no influence on the cost of \_\_\_\_\_ (deployment)

The depth at which we can think is influenced by \_\_\_\_\_ (implementation details)

There are no struct, union, enum, unsigned, typedef, or pointers types in Java

=====

**write three concepts which are introduced in LISP first time?**

- Conditionals: such as if-then-else construct.
- Function types: where functions are just like integers and strings
- Recursion: first language to support it.
- Dynamic typing: all variable are pointers.
- Garbage-Collection.
- Programs composed of expressions.
- A symbol type.
- The whole program is a mathematical function

### Ada is strongly typed language describe how?

ADA is strongly typed. A programming language is strongly typed if type errors are always detected and in ADA Errors can be signaled as exceptions and handled explicitly. Many serious errors such as computational overflow and invalid array indexes are automatically caught and handled through this exception mechanism, improving program reliability....?

### Write a function in LISP to concatenate two lists List a=(3 4 5) List b=(c d 6)

```
>(append '(3 4 5) '(c d 6))
```

(3 4 5 c d 6)..concatenated List.

### In C++ ^ and ~ are used for bitwise exclusive OR and bitwise NOT write there equivalent in ada

Ans:

^ bitwise exclusive OR of c++ = Xor in ADA..

~ = Not in ADA

### What code in LISP language used to compute power of x to y. Marks3:

```
(defun power (x y)
```

```
(if (= y 0) 1 (* x (power x (1- y))))
```

This function computes the power of x to y. That is, it computes  $x^y$

by recursively multiplying x with itself y number of times.

Example;

```
>(power 3 4)
```

A code was given about ADA case concept. Please refer topic "Case statement"

case expression is

when choice list =>

sequence-of-statements

when choice list =>

sequence-of-statements

when others =>

sequence-of-statements

end case;

**How can we determine the size of Array in SNOBOL...2**

In SNOBOL, The size of the array can be determined at run time by using input from the system of using a variable.

Example:

A = ARRAY(INPUT)

May 11, 2012

**Write IO header file for Ada..**

with Ada.Text\_IO; -- intent to use

use Ada.Text\_IO; -- direct visibility

**name functions that are used in List construction**

Lists can be created and extended with the help of three basic functions.

1. cons,
2. list
3. append.

**What are operators for Logical AND && and Logical OR || used in ADA.**

Logical AND &&= And in ADA

Logical OR || = Or in ADA

### Features added in COBOL not included in SNOBOL.

COBOL has introduced hierarchical data structures (records) and nested selection statements.

The COBOL language supported long variable names (up to 30 characters) with hyphens, data division, and fixed-point arithmetic.

These all feature in COBOL language are those which were added or enhanced and were not included in SNOBOL..

### 2013 spring mid paper:

#### what is the functionality of KEYWORD INT|CAHR IN snobol4 2

In SNOBOL 4 Vertical bar is used to specify pattern alternation.

KEYWORD INT|CAHR

This statement assigns the pattern 'INT' | 'CHAR' to the variable KEYWORD.

KEYWORD will now get a new value which is 'INT' | 'CHAR' So we can create new pattern by using the assignment statement.

#### What is error in it >(sqrt x) in LISP 3

Ans:

This code generates an error as the symbol x is not associated with any value and thus cannot be evaluated.it will give error like this.

Error: The variable X is unbound.

#### How many element in it (a (d e)) related to lisp... 2

There are two elements.

Atom = a

List = (d e)

### What is predicate in lisp language conform ?...3

A predicate is a special function which **returns NIL** if the predicate is false, T or anything other than NIL, otherwise. Predicates are used to build Boolean expressions in the logical statements.

Comparative operators in LISP used as functions for numerical values and return a T or NIL. =, >, <, >=, <=;

For example 1:

```
(= (+ 2 4) (* 2 3))
```

T

example 2:

```
(> (- 5 2) (+ 3 1))
```

NIL

**write two comparative operator used as function for numerical values in lisp programming language ? 2 marks**

Comparative operators which are used as functions for numerical values are

=, >, <, >=, <=;

**which data structure map pairs of associated data object in snobol? 3**

A table data structure maps pairs of associated data objects. Tables have varying lengths and are dynamically extended if necessary.

T = TABLE().....If want to creates a table of unspecified length.

T = TABLE(N).....if want to creates a table that initially has room for N elements

Once created, table can be used in a manner similar to arrays. The only difference is that they are indexed by the key value and not the index number. For example:

T<'A'> = 1

Associates key 'A' with value 1.

which language first provided exception handling among c++ , ada? .... 3 marks

Ans;

Although PL/1 has introduced the concept of exception handling. But ADA has elaborated this concept.

- Ada was the first language to provide a structured exception handling mechanism.
- Ada includes an exception handling mechanism so that these problems can be dealt with at run-time.
- Errors can be signaled as exceptions and handled explicitly.
- Many serious errors such as computational overflow and invalid array indexes are automatically caught and handled through this exception mechanism, improving program reliability.

write a complete program in ada to print this output by only using put\_line command. use the loop isn't allowed.

print this code

```
*  
***  
****  
*****
```

Ans .

procedure Starprint is

begin

```
Put ("*");  
New_Line ;
```

```
Put ("***");  
New_Line ;
```

```
Put ("*****");  
New_Line ;
```

```
Put ("*****");  
New_Line ;  
  
end Starprint;
```

write program in ada using loop to print this code

code :

```
*  
**  
***  
****  
***** (these star have to print using loop)
```

Ans:

procedure Loopprint is

```
    Number_Of_Time: Integer:=5;
```

```
begin
```

```
    for I in 1..Number_Of_Time loop
```

```
        for J in 1..I loop
```

```
Put ('*');
```

```
        end loop;
```

```
        New_Line;
```

```
    End loop;
```

```
end Nested_Loops;
```

```
end Loopprint;
```

**does ada believe in human programming if yes then y?**

yes, ada believe in human programming .Programming as a human activity is the main goal of ADA. In ADA the whole philosophy is that a programmer is a human and all humans make

mistakes. So to make it safer, Ada uses discriminants in a record that tell what type of data is currently stored there.

### **c or ada, which support subprograms?**

ADA supports subprograms. There are two types of subprograms:

1. procedures and
2. Functions.

we can also have nesting of subprograms. A subprogram defined inside another subprogram can be used by its parent, children, or sibling subprograms only.

### **concepts of car, cdr, first and rest in Lisp?**

In order to select elements from a list, selectors functions are used. There are two basic selector functions known as

1. first (or car) and
2. rest (or cdr).

The rest can be build with the help of these functions.

1. **First (or car)** takes a list as an argument and returns the first element of that list.

**example:**

```
>(first '(a s d f))
```

a

2. **rest (or cdr)** takes a list as its argument and returns a new list after removing the first element from the list.

examples:

```
>(rest '(a s d f))
```

```
(s d f).
```

### **Note language generality? 2marks**

#### **Generality**

Generality is also an important factor and deals with the applicability of the language to a

range of different domains. For example, C is more general purpose than LISP or FORTRAN and hence can be used in more domains than these two languages.

### CS 508 mid term spring 2010 paper

#### **Compare ADA dotimes loop with dolist and dotime in LISP (5)**

Apart from recursion, in LISP we can write code involving loops using iterative non recursive mechanism. There are two basic statements for that purpose: **dotimes** and **dolist**.

#### **DOTIMES**

**dotimes** is like a counter-control for loop. Its syntax is given as below:

**(dotimes (count n result) body)**

It executes the **body** of the loop n times where count starts with 0, ends with n-1.

The **result** is optional and is to be used to hold the computing result. If **result** is given, the function will return the value of **result**. Otherwise it returns NIL. The value of the **count** can be used in the loop body.

#### **DOLIST**

The second looping structure is **dolist**. It is used to iterate over the list elements, one at a time. Its syntax is given below:

**(dolist (x L result) body)**

It executes the **body** for each top level element x in L. x is not equal to an element of L in each iteration, but rather x takes an element of L as its value. The value of x can be used in the loop body. As we have seen in the case of **dotimes**, the **result** is optional and is to be used to hold the computing result. If **result** is given, the function will return the value of **result**. Otherwise it returns NIL.

#### **Differentiate between ADA access types and C/C++ pointer type (5)**

An *access* type in ADA roughly corresponds to a C++ pointer.

**type** Address\_Ref is access Address;

A\_Ref := new Address;

```
A_Ref.Postal_Code := "94960-1234";
```

Unlike C, there is no notational difference for accessing a record field directly or through an access value.

To refer to the entire record accessed by an access value use the following notation:

```
Print(A_Ref.all);
```

### **Why predicate is a special function in LISP (3)**

A predicate is a special function which returns NIL if the predicate is false, T or anything other than NIL, otherwise. Predicates are used to build Boolean expressions in the logical statements.

The following comparative operators are used as functions for numerical values and return a T or NIL. =, >, <, >=, <=;

### **What additional features are added in COBOL (2)**

COBOL uses level numbers to show nested records; others use recursive definitions

Business applications

reports, decimal arithmetic, character operations - COBOL

It was designed to look like simple English to broaden the base of computer users.

### **Is there anything like templates as in C++ (2)**

Generics are like templates in C++ and allow parameterization of subprograms and packages with parameters which can be types and subprograms as well as values and objects.

### **Difference of C and C++ 2 marks**

C++ Differs from C in two ways:

1. The control expression can also be Boolean
2. The initial expression can include variable definitions (scope is from the definition to the end of the function in which it is defined).

**Write types of loop in Ada.**

There are three types of loops in ADA.

1. For loop
2. While loop
3. Simple ..unconditional infinite loop

### **Brief note on Orthogonality.**

Orthogonality is a very important concept. It addresses how relatively small number of components that can be combined in a relatively small number of ways to get the desired results. It is closely associated with simplicity: the more orthogonal the designs the fewer exceptions and it make it easier to learn, read, and write programs in a programming language. The meaning of an orthogonal feature is independent of the context. The key parameters are symmetry and consistency. For example pointers are an orthogonal concept.

```
1)S=0
   S=0
-- N=LT(N,13)N+1:F(DONE)
SUM=SUM+_N:(ADD)
DONE OUTPUT=SUM
```

### **SYNTACALLY SHOW HOW CAN WE DEFINE THE LISP?**

LISP is a representative language of this style of programming.

Lisp stands for "LIST Process". It was invented by John McCarthy in 1958 at MIT.

1. It has simple data structure (atoms and lists) and makes heavy use of recursion.
2. It is an interpretive language and has many variations.
3. The two most popular versions are
  - Scheme and
  - Common Lisp
4. It is the most widely used AI programming language.
5. Atom and LIST are two main elements of LISP.

???????????

3)DEFERENTIATE BETWEEN TEH ada record type and c/c++??

4)define struct with the proper example

6) is polymorphism allowed in Ada???

There are two means of using polymorphism in Ada. *Static polymorphism* is provided through the generic parameter mechanism whereby a generic unit may at compile time be instantiated with any type from a class of types. *Dynamic polymorphism* is provided through the use of so-called class-wide types and the distinction is then made at runtime on the basis of the value of a tag.

8) Algol main feature?

**Salient features of the language are:**

1. Concept of type was formalized
2. Names could have any length
3. Arrays could have any number of subscripts. Lower bound of an array could be defined
4. Parameters were separated by mode (in & out)
5. Subscripts were placed in brackets
6. Compound statements were introduced (begin ... end)
7. Semicolon as a statement separator was used
8. Assignment operator was :=
9. if had an else-if clause

**Why parallel programming is used in ADA? (3) marks**

**Babbage's Analytical Engine (2)**

Charles Babbage is considered to be the inventor of the first computer. This machine, known as the Analytical Engine or Difference Engine, was invented in the 1820's. In the beginning, it could only be made to execute tasks by changing the gears which executed the calculations. Thus, the earliest form of a computer language was physical motion.

The key features of this machine included

- 1) Memory (called store),
- 2) jump,
- 3) loop, and the
- 4) concept of subroutines.

It was motivated by the success of power looms. It had limited capability and because of the technological limitations this design could not be fully implemented.

December 1, 2010 at 8:23pm

### What is the difference between $>$ (intersection L1,L2) and (set-difference L1,L2) Lisp functions. 5

These are set operations in LISP.

- $>$ (union L1 L2) ; returns the union of the two lists
- $>$ (intersection L1 L2) ; returns the intersection of the two lists
- $>$ (set-difference L1 L2) ; returns the difference of the two lists

### Give silent features of packages in Ada 5

#### Packages

These are primary structure used for encapsulation in.

- 1) Packages are used to group data and subprograms.
- 2) Packages can be hierarchical, allowing a structured and extensible relationship for data and code.
- 3) All the standard Ada libraries are defined within packages.
- 4) Package definitions usually consist of two parts:
  1. package specification and
  2. package body.
- 5) Package bodies can also declare and define data types, variables, constants, functions, and procedures not declared in the package specification.

### Creating list functions in Lisp may be written as (append '(ab)'(cd)) so why we use the single quote (') in front of atom or list. 3

it a shorthand notation for the function, quote. Quote is a function that takes in its argument and sends back as its answer, the same value that went into it.

### What is batting-average in Ada explain with suitable example. 3

The type Batting\_Averages is a fixed point type whose values are evenly spaced real numbers in the range from 0 through 1. The distance between the values is no more than 1/1000. The actual distance between values may be 1/1024, which is 2-10.

For example:

type Batting\_Averages is delta 0.001 range 0.0..1.0;

The distance between values is implemented as a power of 2.

### Write pattern matching statement in SNOBOL. 2

In SNOBOL Space is used as the pattern matching operator and the statement has the following form:

subject pattern

There is no assignment operator in this case. Both the subject and the pattern are strings and this statement tries to match the pattern in the subject. It will be successful if the match is found and will result in failure otherwise.

Example:

```
TRADE = 'PROGRAMMER'
```

```
PART = 'GRAM'
```

```
TRADE PART
```

### a) What are interpreted and compiled languages? Give examples (5)

Interpreted computing languages are languages whose source code is processed by a software program called an interpreter that reads in the text and immediately acts upon the instructions defined by the text. Compiled computing languages are languages whose source code is processed by a software program called a compiler that converts the source code into a file which can then be run directly or indirectly by a computer operating system.

1. Prolog It is an interactive (hybrid compiled/interpreted) language.
2. PHP is interpreted
3. FORTAN is compiled
4. SNOBAL is compiled

### Comparison between functional and imperative languages?

A most important class of programming languages, known as the imperative languages, is based upon the von Neumann Architecture. This includes languages like FORTAN, COBOL, Pascal, Ada, C, and many more.

While functional language is where there is no need for assignment and in it control flow is achieved via recursion and conditional expressions.

### Convert the following Ada code into equivalent C code.

```
case ch is
```

```
when 'A' | 'E' | 'I' | 'O' | 'U' =>
```

```
putline("this is an uppercase vowel");
```

```
when 'J' .. 'N' =>
```

```
putline("between uppercase J and N");
```

```
when others =>
```

```
putline("something else");
```

end case;

What is valid or invalid statement in ADA language 5

a=(1....5) integer...(invalid)

b=(1....5) integer...(invalid)

c,d=(1....5) float...(invalid)

All Declarations are invalid.

There should be proper assignment operator (:=) . In these colon before equal is missing ...

Improper array declaration here too .. only 2 dots should be there(..)

condition

a=b

a=c

b=a

Write Output of the following code. ....5

Procedure Color is

Color= BasicColor(Red, green, blue, yellow)

begin BasicColor

Color:=BasicColor'PRED(RED)

end color

Asn:

RED.

## How to compare and examine language?

1. Readability
2. Writ ability
3. Reliability

## Pattern matching Output of code?

Sentence='this is ur pen'

Sentence 'ur'= 'my'

Sentence 'my'='

Z= '10'

## Prolog learning curve is very high meanings of this statement 3marks

Prolog is a declarative programming language where we only specify what we need rather than how to do it. It has a simple concise syntax with built-in tree formation and backtracking which generates readable code which is easy to maintain. It is relatively case and type insensitive and is suitable for problem solving / searching, expert systems / knowledge representation, language processing / parsing and NLP, and game playing.

## Discuss the issue related with declaring a method/class as final java? 3 marks

### Final Fields and Methods

A final field is like a constant: once it has been given a value, it cannot be assigned to again. Fields and methods can also be declared final. A final method cannot be overridden in a subclass.

## With suitable examples the concept of boxing in C#. How C# is different there concept from other languages? 3 marks

### **Boxing**

Boxing is converting any value type to corresponding object type and convert the resultant 'boxed' type back again.

```
int i = 123;
```

```
object box = i; // value of i is copied to the object box
if (box is int) // runtime type of box is returned as boxed value type
{
Console.WriteLine("Box contains an int"); // this line is printed
}
```

**Notation of actual /formal Parameters? 3 marks**

**Actual/Formal Parameter Correspondence:**

1. Positional

2. Keyword

e.g. SORT(LIST => A, LENGTH => N);

**Default Parameter Values**

```
procedure SORT(LIST : LIST_TYPE;
               LENGTH : INTEGER := 100);
```

...

```
SORT(LIST => A);
```

**Dangling pointer how to state in java? 5marks**

Dangling pointers are those pointers that points to a heap-dynamic variable that has been Deallocated. Automatic garbage collection prevents memory leaks and dangling pointers.

Example missing...☹️

**Aliasing Problem in java Script? with suitable examples. 5 marks**

**Aliasing Problems in Java**

The fact that arrays and classes are really pointers in Java can lead to some problems. Here is a simple assignment that causes aliasing:

```
int [] A = new int [4];
```

```
int [] B = new int [2];
```

This obviously creates problems. programmers have to be very careful when writing programs in Java.

In Java, all parameters are passed by value, but for arrays and classes the actual parameter is really a pointer, so changing an array element, or a class field inside the function does change the actual parameter's element or field.

### Example:

```
void f( int [ ] A ) {  
    A[0] = 10;    // change an element of parameter A  
    A = null;    // change A itself  
}  
  
void g() {  
    int [ ] B = new int [3];  
    B[0] = 5;  
    f(B);  
}
```

B is not null here, because B itself was passed by value however, B[0] is now 10, because function f changed the first element of the array.

### **What is monitor in java thread?**

To prevent problems that could occur by having two methods modifying the same object, Java uses monitors and the synchronized keyword to control access to an object by a thread. Any object that implements the "synchronized" keyword is considered to be a monitor. A monitor is an object that can move a thread process between a blocked and running state. Monitors are required when a process needs to wait for an external event to occur before thread processing can continue. A typical example is when a thread can't continue until an object reaches a certain state.

### **What queried in prolog?**

Queries are used to retrieve information from the database. A query is a pattern that PROLOG is asked to *match* against the database and has the syntax of a compound query. It

may contain variables. A query will cause PROLOG to look at the database, try to find a match for the query pattern, execute the body of the matching head, and return an answer.

### What is quoted atom in prolog?

Quoted atoms - sequence of characters surrounded by single quotes. Examples: 'Apple' 'hello world'

### What is managed and unmanaged code?

Ans;

#### Managed code

Managed code is executed under the control of Common Language Runtime (CLR).

It has automatic garbage collection. That is, the dynamically allocated memory area which is no longer in use is not destroyed by the programmer explicitly. It is rather automatically returned back to heap by the built-in garbage collector. There is no explicit memory's allocation and deallocation and there is no explicit call to the garbage collector.

#### Unmanaged code

The unmanaged code provides access to memory through pointers just like C++. It is useful in many scenarios. For example:

- Pointers may be used to enhance performance in real time applications.
- In non-.net DLLs some external functions requires a pointer as a parameter, such as Windows APIs that were written in C.
- Sometimes we need to inspect the memory contents for debugging purposes, or you might need to write an application that analyzes another application process and memory.

### How many ways the static binding can be define?

#### Static Binding

A binding is static if it occurs before run time and remains unchanged throughout program execution

If static, type may be specified by either an explicit or an implicit declaration .

**An explicit declaration** is a program statement used for declaring the types of variables .

**An implicit declaration** is a default mechanism for specifying types of variables i.e the first appearance of the variable in the program.

## Write three different methods how to write strings in PHP? 3

### Strings

PHP strings are created using single quote or double quote.

They can also be created by <<< which is called heredoc. One should provide an identifier after <<<, then the string, and then the same identifier to close the quotation. The closing identifier must begin in the first column of the line.

## How to convert PHP string in numeric value? 2

### String conversion to numbers:

When a string is evaluated as a numeric value, the resulting value and type are determined as follows.

- The string will evaluate as a **float** if it contains any of the characters '.', 'e', or 'E'. Otherwise, it will evaluate as an integer.
- The value is given by the initial portion of the string. If the string starts with valid numeric data, this will be the value used. Otherwise, the value will be 0 (zero).

### Example:

```
<?php
    $foo = 2 + "10.5";           // $foo is float (12.5)
    $foo = 1 + "-1.1e3";        // $foo is float (-1099)
    $foo = 1 + "Ali-1.3e3";     // $foo is integer (1)
    $foo = 1 + "10 Small Cats"; // $foo is integer (11)
    $foo = "10.0 cats " + 1;    // $foo is float (11)
?>
```

Using strings in numeric expressions provides flexibility but it should be obvious that it also is a source to numerous programming errors and mistakes. This kind of error is also very hard to debug and detect.

## Discuss readability problem using control statements? 2

Control statements play an important role in readability.

1. We are all aware of the hazards of goto statement. If a language relies on goto statements for control flow, the logic becomes difficult to follow and hence understand and maintain.
2. Restricted use of goto in extreme was needed and useful as well but with the emergence of new language constructs to handle those situations, it probably not an issue any more.

**What are the enumeration and subrange types? Differentiate these two in points.**

- a) An enumeration is a special kind of value type limited to a restricted and unchangeable set of numerical values. By default, these numerical values are integers, but they can also be longs, bytes, etc. (any numerical value except char). while subrange type is an ordered contiguous subsequence of an ordinal type.
- b) Subrange types behave as their parent types; can be used as for variables and array indices
  - E.g. type pos = 0 .. MAXINT;
- c) Enumeration types are implemented as integers while Subrange types are the parent types with code inserted (by the compiler) to restrict assignments to subrange variables.

**Short circuiting in java and c and c++ 3marks**

**Short Circuit Evaluation;**

- A and B

- A or B

- Example

index := 1;

while (index <= length) and

(LIST[index] <> value) do

index := index + 1

**C, C++, and Java:** use short-circuit evaluation for the usual Boolean operators (&& and ||), but also provide bitwise Boolean operators that are not short circuit (& and |) Structs c++ and c# structs are public by default. Structs are basically "objects" that contain variables inside of them. The **struct** type: In C#, a **struct** is a value type.

**what seal struct and abstract means in C# ?**

An *abstract class* is any class that includes an abstract method. It is similar to Pure virtual in C++. If a class includes an abstract method, the class *must* be declared abstract, too.

**Write two differences between C++ and Java**

Java is more robust than C++. Some of the reasons are:

- Object handles are initialized to null (a keyword)
- Handles are always checked and exceptions are thrown for failures
- All array accesses are checked for bounds violations
- Automatic garbage collection prevents memory leaks and dangling pointers
- Type conversion is safer
- Clean, relatively fool-proof exception handling
- Simple language support for multithreading
- Bytecode verification of network applets

**Final Term fall 2010**

**Difference between proper example Union type in c/c++ and ada discriminated type?**

**2marks**

Discriminated records are like union types in C. There are however major differences between C union types and Ada discriminated records. The union type in C is fundamentally unsafe, and therefore unacceptable.

**Division by zero is an expression? what type of error is this and either handled by compiler or it through exception? 3marks**

Division by zero is due to Inherent limitations of arithmetic. it is run-time errors sometimes called exceptions.

**How Short circuit evolution is performed in Ada and fortran? 3 marks**

In Ada, programmer can specify either short-circuit is specified with

1. 'and then'
2. 'or else'

While in FORTRAN 77, short circuiting is there, but any side affected place must be set to undefined. The Problem with Short Circuiting is

`(a > b) || (b++ / 3)`

Short-circuit evaluation exposes the potential problem of side effects in expressions

**What term Class attribute in C#? marks**

In C#, Attributes are used to give information to .Net compiler

**Example:**

it is possible to tell the compiler that a class is compliant with .Net Common Language Specification.

```
[CLSCompliant (true)]  
public class MyClass  
{  
    //class code  
}
```

**Define and explain “throws Clause” and the “finally Clause” with respect to exception handling in Java. (5)**

**Throws Clause**

Throws clause is overloaded in C++ and conveys two different meanings: one as specification and the other as command. Java is similar in syntax but different in semantics. The appearance of an exception class name in the **throws** clause of Java method specifies that the exception class or any of its descendents can be thrown by the method.

A C++ program unit that does not include a throw clause can throw any exceptions. A Java method that does not include a throws cannot throw any checked exception it does not handle. A method cannot declare more exceptions in its throws clause than the methods it overrides, though it may declare fewer. A method that does not throw a particular exception, but calls another method that could throw the exception, must list the exception in its throws clause.

## The finally clause

A Java exception handler may have a finally clause. A finally clause *always* executes when its try block executes (whether or not there is an exception). The finally clause is written as shown below:

```
try {  
...  
}  
catch (...) {  
...  
}  
...  
finally {  
...  
}
```

A finally clause is usually included to make sure that some clean-up (e.g., closing opened files) is done. If the finally clause includes a transfer of control statement (return, break, continue, throw) then that statement overrides any transfer of control initiated in the try or in a catch clause.

**Explain the Prolog features regarding its applications in real world. (5)**

We can find PROLOG in areas like

1. Expert systems
2. Artificial intelligence,
3. Natural language understanding,
4. logical puzzles and games.
5. theorem proves construction
6. The variant of PROLOG called DATALOG is used in database management.
7. It is relatively case and type insensitive and is suitable for problem solving / searching, expert systems / knowledge representation, language processing / parsing and NLP, and game playing.

Prolog is a declarative programming language where we only specify what we need rather than how to do it. It has a simple concise syntax with built-in tree formation and backtracking which generates readable code which is easy to maintain.

### **Describe the programming steps in Prolog? (5)**

PROLOG programming follows the following steps:

- Declaring some facts about objects and their relationships
- Defining some rules about objects and their relationships
- Asking questions about objects and their relationships

### **How to define PHP classes?**

Classes and objects in PHP are similar to Java. A variable of the desired type is created with the *new* operator. It supports Single inheritance only and uses the keyword *extends* to inherit from a super class. The inherited methods and members can be overridden, unless the parent class has defined a method as *final*.

### **Union in c++ and java difference?**

C and C++ have free unions (no tags)

- Not part of their records
- No type checking of references

Java has neither records nor unions

---

### **Two difference b/w c++ and java**

1. Control expression must be Boolean
2. Scope of variables defined in the initial expression is only the loop body

### **Two difference b/w c and c++**

The control expression can also be Boolean

The initial expression can include variable definitions, scope is from the definition to the end of the function in which it is defined.

### **What is notify and wait in java**

Threads are based upon the concept of a Monitor. The wait and notify methods are used just like wait and signal in a Monitor. They allow two threads to cooperate and based on a single shared lock object.

### **what is notify() and notifyall() in java?**

There is a slight difference between notify and notifyAll. As the name suggest, notify() wakes up a single thread which is waiting on the object's lock. If there is more than one thread waiting, the choice is arbitrary i.e. there is no way to specify which waiting thread should be re-awakened. On the other hand, notifyAll() wakes up ALL waiting threads; the scheduler decides which one will run.

Explain Stack Dynamic variable with example? 5

- Fixed stack dynamic - range of subscripts is statically bound, but storage is bound at elaboration time e.g. C local arrays are not static
  - Advantage: space efficiency
- Stack-dynamic - range and storage are dynamic, but fixed from then on for the variable's lifetime e.g. Ada declare blocks declare

STUFF : array (1..N) of FLOAT;

begin

...

end;

Advantage: flexibility - size need not be known until the array is about to be used

### **Differentiate between the implicit and explicit type conversion with respect to Java. (5)**

Java is much stronger than C++ in the type conversions that are allowed.

Booleans cannot be converted to other types. For the other primitive types (char, byte, short, int, long, float, and double).

#### **Implicit conversions:**

An implicit conversion means that a value of one type is changed to a value of another type without any special directive from the programmer. A char can be implicitly converted to an int, a long, a float, or a double.

For example:

```
char c = 'a';  
int k = c;  
long x = c;  
float y = c;  
double d = c;
```

### Explicit conversions:

Explicit conversions are done via *casting*: the name of the type to which you want a value converted is given, in parentheses, in front of the value.

### For example,

The following code uses casts to convert a value of type double to a value of type int, and to convert a value of type double to a value of type short:

```
double d = 5.6;  
int k = (int)d;  
short s = (short)(d * 2.0);
```

Casting can be used to convert among any of the primitive types except boolean.

### Two example of atoms in PROLOG syntax?

Atoms:

AlphaNumeric start with lower letter

Quoted Atom sequence of character quoted by single quotes

Symbolic symbol & +-\* special atom !;[]{}

### Two example of predefined reference type in C#? 2 marks

1. object
2. string

**Differentiate between client side scripting and server side scripting languages? (5)**

**JavaScript – client side scripting**

Primary objective of JavaScript is to create dynamic HTML documents and check validity of input forms. It is usually embedded in an HTML document. It is not really related to Java

**PHP (Personal Home Page) – server-side scripting**

It is interpreted on the Web Server when the HTML document in which embedded is requested by the browser. It often produces HTML code as an output and is very similar to JavaScript. It allows simple access to HTML form data and makes form processing easy. It also provides support for many different database management systems and hence provides Web access to databases.

**Differentiate between client side scripting and server side scripting languages? (5)**

**JavaScript – client side scripting**

Primary objective of JavaScript is to create dynamic HTML documents and check validity of input forms. It is usually embedded in an HTML document. It is not really related to Java

**PHP (Personal Home Page) – server-side scripting**

It is interpreted on the Web Server when the HTML document in which embedded is requested by the browser. It often produces HTML code as an output and is very similar to JavaScript. It allows simple access to HTML form data and makes form processing easy. It also provides support for many different database management systems and hence provides Web access to databases.

**Differentiate between the Dynamic Type binding and Static type binding?**

**Static and Dynamic Binding**

A binding is static if it occurs before run time and remains unchanged throughout program execution.

A binding is dynamic if it occurs during execution or can change during execution of the program.

**What is the run time environment in case of C# similar to JVM in case of Java programming language?**

It is Similar to Java and C++ in many respects. The .NET platform is centered on a Common Language Runtime (CLR - which is similar to a JVM) and a set of libraries which can be exploited by a wide variety of languages which are able to work together by all compiling to an intermediate language (IL).

**Explain the Concept of Explicit and Implicit heap, dynamic variables and stack dynamic storage binding? Clarify the concept with suitable examples**

### Explicit Heap Dynamic Variables

- Allocated and de-allocated by explicit directives, specified by the programmer, which take effect during execution
- Referenced only through pointers or references

Examples: Dynamic objects in C++ (via new and delete) All objects in Java

**Advantage:** Provide dynamic storage management

**Disadvantage:** Inefficient and unreliable

### Implicit Heap Dynamic Variables

- Allocation and de-allocation is caused by assignment statements

Example: All variables in SNOBOL

**Advantage:** Flexibility

**Disadvantages:** Inefficient, because all attributes are dynamic Loss of error detection

**Write Ada language procedure (Code) to display your name on screen**

```
withAda.Text_io;  
  
useAda.Text_io;  
  
procedureHello is  
  
begin  
  
Put_Line("Shazia Bano!");
```

```
endHello;
```

### **In C# how many Allocation is handled through manage Code**

There is no explicit memory's allocation and deallocation and there is no explicit calls to the garbage collector i.e. call to destructor.

### **Do we have pointers for reference types in case of C#?**

No! in c#, Pointer can be declared to hold the memory addresses of value types. That is, we cannot have pointers for reference type.

### **How many character types are supported in Ada Language?**

There are two built in character types in Ada

- Simple 8-bit ASCII Characters
- Wide\_Character that support 16-bit Unicode/ISO standard 10646.

These are good enough for most purposes, but you can define your own types just like the integer types:

### **What are subprogram issues in different languages?**

Subprograms are complex structures in programming languages, and it follows from this a lengthy list of issues in their design. Following are some issues with sub programming,

1. What parameter-passing method or methods are used?
2. Are the types of the actual parameters checked against the types of the formal parameters?
3. Are local variables statically or dynamically allocated?
4. If subprograms can be passed as parameters, what is the referencing environment of such a subprogram?
5. If subprograms can be passed as parameters, are types of parameters checked in calls to the passed subprograms?
6. Can subprograms be overloaded?
7. Can subprograms be generic?
8. Is neithersparate or independent compilation possible?

### **What is seal struct and abstract in C#**

**Abstract:** A class declared as 'abstract' cannot itself be instanced - it is designed only to be a base class for inheritance.

**Sealed:** A class declared as 'sealed' cannot be inherited from. It may be noted that structs can also not be inherited from. But it can inherit from other class.

**Structs:** Structs in C# are designed to encapsulate lightweight objects. They are value types (not reference types), so they're passed by value. They are sealed, which means they cannot be derived from or have any base class other than System.ValueType, which is derived from Object. Structs cannot declare a default (parameterless) constructor. Structs are more efficient than classes, that's why they are perfect for the creation of lightweight objects.

**SNOBOL stands for what?**

SNOBOL stands for StriNg Oriented SYMBOLic Language.

**Write a program in Ada that prints the text Area of your course for example "BS-Computer Sciences."**

```
with Ada.Text_IO;  
use Ada.Text_IO;  
procedure BSCS is  
begin  
  Put_Line("BS-Computer Sciences!");  
End BSCS;
```

**Implementation models?**

**Implementation Models:**

Pass-by-value (in mode): Either by physical move or access path

**What is the function of cut(!) predicate in PROLOG? 2 marks**

cut <or> ! is a special predicate. And it is used in the body of a given clause so as to avoid using clauses appearing after the given clause in the program. It is like a goal which is always succeeded but never backtracked past.

For example, to prevent extra solutions being found by Prolog and avoid additional computations that are not desired or required in a program.

**How do Ada and COBOL differ by syntax and semantics in referencing a record element?  
Give examples of each (5)**

In ADA, Record types are like structures in C.

type Address is record

Street : Unbounded\_String;

City : Unbounded\_String;

Postal\_Code : String(1..10);

end record;

- COBOL uses level numbers to show nested records; others use recursive definitions

- Examples:

COBOL

field\_name OF record\_name\_1 OF ... OF  
record\_name\_name

According to readability point of view COBOL record definition syntax is easier to read and understand but according to writability point of view other languages record definition syntax is easier to write and less time consuming.

We studied several reasons to study programming languages. With that context in mind discuss:

(a) Increased capacity to express programming concepts

(b) Improved background for choosing appropriate languages and show the output.

Discuss the difference between enumeration types of C and Ada. Which is better and why?

Although Just like C, an enumeration type in Ada is a sequence of ordered enumeration literal, It is however different from C in many respects:

There is no arithmetic defined for these types.

**Example**

S1, S2 : State;

S1 := S1 + S2; -- Illegal

Unlike C, the same symbolic literal can be used in two enumeration types.

**Example:**

type RainbowColors is (Red, Orange, Yellow, Green, Blue, Indigo, Violet);

type BasicColors is (Red, Green, Blue);

The Ada compiler will use the type of the variable in question and there will be no ambiguity. Enumeration types are used as attributes or properties of different objects.

## Why it is necessary to use (:) in control flow statement?

PHP offers an alternative syntax for some of its control structures; namely, if, while, for, foreach, and switch. In each case, the basic form of the alternate syntax is to change the opening brace to a colon (:) and the closing brace to endif;, endwhile;, endfor;, endforeach;, or endswitch;, respectively.

## Give two examples of errors when pointer are wrongly used ?

**Dangling pointers:** A pointer points to a heap-dynamic variable that has been deallocated

**Lost Heap-Dynamic Variables :** A heap-dynamic variable that is no longer referenced by any program pointer. The process of losing heap-dynamic variables is called memory leakage

## Difference between Actual Parameter and Predicate parameter? 3 marks

A parameter association defines the association between an actual parameter and a formal parameter.

A predicate is a special function which returns NIL if the predicate is false, T or anything other than NIL, otherwise. Predicates are used to build Boolean expressions in the logical statements.

## Can we define members and length function of our own in LISP?if yes then explain your answer with suit able example.

yes we can, The LISP built-in function list-length counts the number of elements in a list. For example,

```
USER(33): (list-length '(2 3 5 7 11 13 17 19))
```

8

a function can be implemented recursively. A given list  $L$  is created by either one of the two constructors, namely nil or a cons:

- *Case 1: L is nil.*  
The length of an empty list is zero.
- *Case 2: L is constructed by cons.*  
Then *L* is composed of two parts, namely, (first *L*) and (rest *L*). In such case, the length of *L* can be obtained inductively by adding 1 to the length of (rest *L*).

????

Write a code in SNOBOL by using indirect referencing which decreases the value by 1.

**Whose better among C/C++ and Ada according to overloading concept..**

As Use of an operator for more than one purpose is known as operator overloading. Ada allows a limited overloading of operators. C++ and Ada allow user-defined overloaded operators...???

What is inference mechanism in Prolog? Give example.

How to type comment in PROLOG language.

What will happen if user does not provide the string array as the argument to the method in Java, will there be any kind of error or not ?

2. case A is

when 1 =>fly;

when 11| 14 => null;

when 22| 20.....30 =>swim;

when others => complain ;

end case;

what will be executed if A condition is 12. .... 2 marks

type big is mod 8.

big :Big :=3; this code will generate an error in ada ?

And ek qs tha keeping in mind that expressions in LISP are in the List show the top variables of the following list.

Ese kr k lists thin...

(add 4, mult 3)

(oranges, grapes, apples)

PC130200\*\*\*\*\*