

01-08-2023

Software Development

1st lecture

Computer

Algorithms

computer is an integrated set of Algorithem and structure that a capable of retaining vi manipulating storage. storing retting manipulating according to the

Algorithm :-

To solve Problems step by steps."

⇒ Math is a language

Forms of Algorithm

(1) Flow chart:-

The diagrammatic representation of Programming Language.

(2) Pseudo code:-

Linguistic representation of solution of Problems in natural language.

(3) Source code:-

The code written in Programming language Solution of the Problem.

Syntax:-

The grammatical arrangement of words is called syntax.

(Algorithm) is organized in English, مُعِرَّبٌ (Algorithm) is organized in Arabic, مُعَرَّبٌ

"Arabic"

2nd Lecture 02/08/2023

Software:-

intangible parts of the computer is called Software. The set of instructions are called software.

Hardware:-

Tangible parts of the computer is called Hardware.

Language:-

زبان

communication medium b/w two or more Partys. Partys.

میکرو ہر لئے زبان اس تھالی ہے۔

جواب :- میکرو میں باعثیتی زبان اس تھالی ہے۔

اصل فریضیاتی سرنٹ ہے۔

Flow of charges called current.

(Bit)

دیلنج :-

مرنٹی دھرنا، جو بندہ کی دوسری طبقہ میں ہے جو داری کی دلیل نہ لانی ہے اس کو بولائیں ڈفرینس بھی سمجھیں

(Bit)

0V - 5V

ایسا اس سکلر:-

Distance by nature \rightarrow Time Travel

Voice analog ha.

آدا اس اس سکلر:-

0V

ڈیجیٹل سکلر:-

پہل آن:- مالک طور پر بیت لامبی

$T = 3V$

فیل آف:-

$5V < 3V$

سکلر حاملہ ایشادر

what are integers?

W^e \mathbb{Z} \mathbb{N} integers $\mathbb{Z} = \{... -3, -2, -1, 0, 1, 2, 3, ...\}$ which are whole numbers are called integers.

Fond Exde-

1 , 2 , 3 , 4 , 5

Floating numbers.

Exp - 3.4., 44.5 - 64.8

اس فیبر جو مکمل و یا پرنسپرینگ فلائٹ میکر اس کے لئے اس

ڈیجیٹل سینٹر اور بیتھکز (Bit) کا کام نہ ہے۔

بِحَمْلِ سُكُنٍ رِيَائِسٍ كَبِلَةٍ هُنَّ

ADC are called modulation.

DAC one called demodulation

←

P. 5

مکھری کا ایڈرنس ایبل یونٹ بائیکٹ سے -

~~Bit is the Addressable unit of the memory.~~

Bit -

The smallest unit of the memory which is addressable is called Bit.

3rd Lecture

Seven bit system.

四

Computer Language

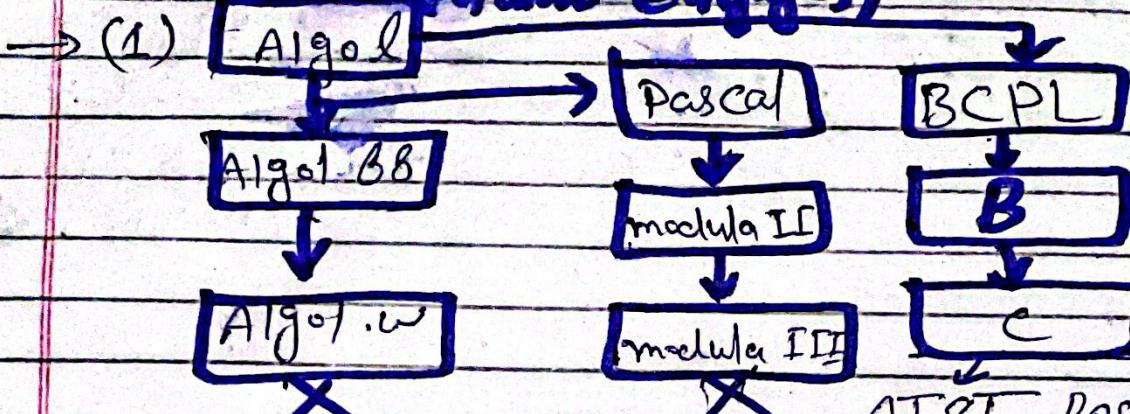
پہنچانے والے بینیاتی لغت

$A \rightarrow 0100001$

$B \rightarrow 0100000$

جسکو کچھ سامنے لے کر نیکوں ایجاد کر سکا جس کو اسی کام میں بھائی بنے۔
Assembler

Natural Languages



AT&T Research

1960

کامپیوٹر سیستم کا ایجاد کرنے والے
Operating system

ANSI (American National Standard Institute)

American national standard institute.

Operating system (MS-DOS, Linux, Unix, Mac OS, iOS, Android, etc.)

Windows, Mac OS, Linux, Android

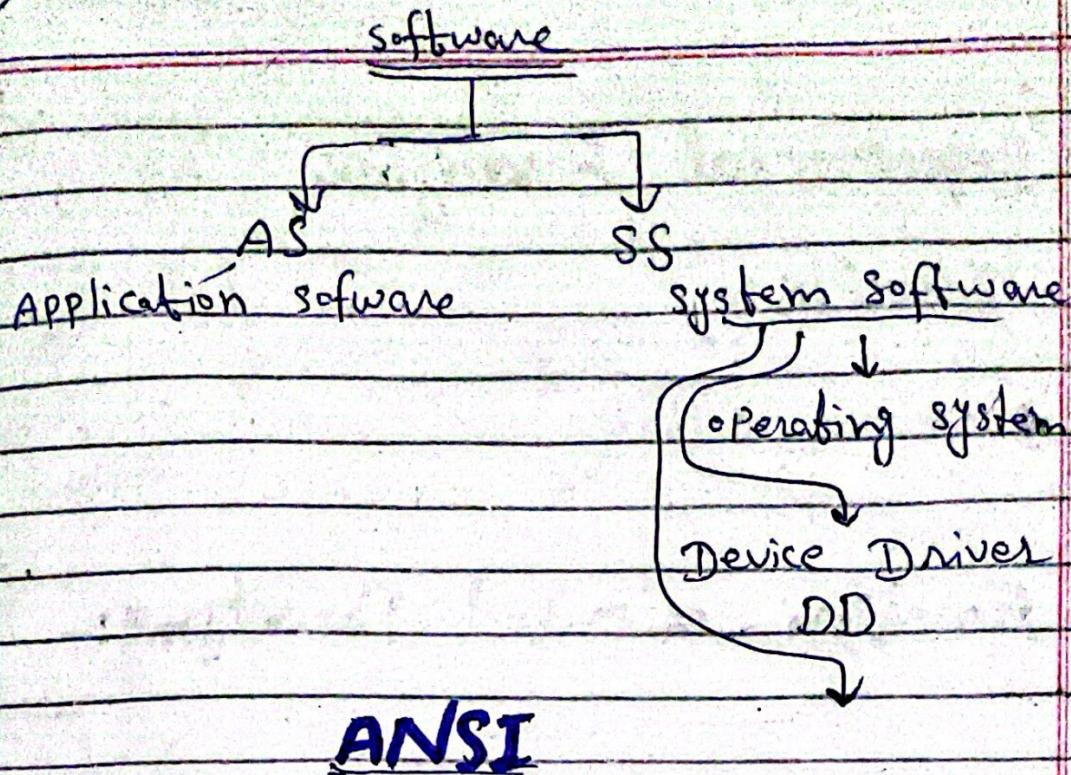
Operating System

پروگرام

مشترک

Windows

• Linux, Mac OS, etc.
interface



ANSI

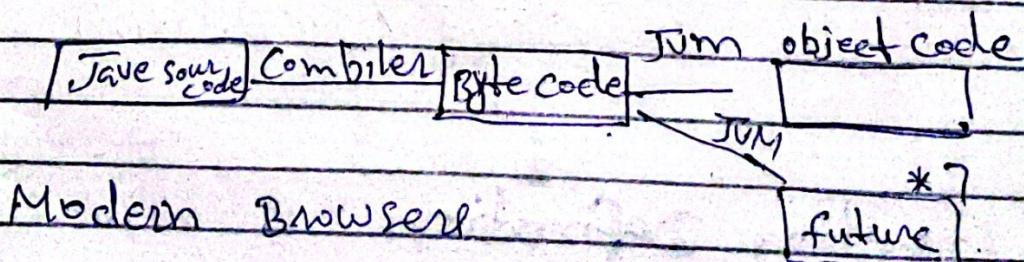
C++

objective c

PC :- - بس بن کی جسے بھی بنی۔ حرف
One Person user operating system
- کمپیوٹری Language اور آئی ڈی رکارڈنگ Language oak
- جو کام زبان وہاں تک ملے تو دیا جائے۔ Java

Web 1.0

Web 2.0



Modern Browsers

* future

~~Java Script~~ ~~Java Script~~ ~~Java Script~~ ~~Java Script~~ ~~Java Script~~

67-08-2023

Programming Paradigme sequential Paradigme

iii

Example:-



-) gt compiles the instruction in sequence one by one.

⇒ Function oriented Paradigme:-

- Translate ↔ ~~list~~
execution ↔ lists
-) In this compiler can jump from one code segment to another to execute function call.

⇒ Function

-) A subprogram in a program
-) A named group of instruction is known as function.

(1) Function Definition .

(2) Function Call

(3) Function Prototyping / Declaration

1

Function Definition

Types:

list definition

- (a) Function Header
- (b) Function Body

Parenthesis ()

Braces { }

Bracket []

(a) Function Header

Type:-

- RDT (return Data type)
- Function identifier → FI
- List of Parameters List of Parameter

RDT FI (List of Parameter if any)

Function Header ← { written value
 } int sum (int a, int b)

Void sss (int a, int b, int c, int d)
 { return a+b+c+d; }

"Reserve word:-"

The word which have
 Predefined meaning is
 called Reserve word.

int sum(int a, int b)

Function body ← {
 } return a+b;

Void display ()

return data type is

sum (10, 20)

RDT

int

FT

sum

List of Parameter

(int a, int b)

if any

int sum(int a, int b)
{ return a + b; }

function call

sum(10, 20);

Actual Parameters (اے جو دیے جائے جائیں) ←

Formal Parameters (اے جو دیے جائے جائیں) ←

Arguments (اے جو اپنے Prototyping کے لئے دیے جائے جائیں) ←

Main function

int main()

start ← { APP life
} → end

Function Call

int c = sum(20, 40);

cout << c;

return 0;

~~float sss (float a, float b)~~
~~{ return a + b; }~~

1) encapsulation

2) Polymorphism

3) inheritance

4) Abstraction

four pillars of OOP

10-08-2023
Thursday

Function Prototyping / declaration

graphical user interface

1) GUI

2) CLI

include <stdio.h>

// Function Prototyping / Declaration

int main()

{ // Function Call

int c = sum(23, 76);

Print ("%d", c);

return 0;

}

// Function Definition.

int sum(int x, int y)

{ return x + y;

}

oop Four Pillars of oop.

1) encapsulation

2) Polymorphism

3) Inheritance

4) Abstraction

10-08-2023
int sum

object

Class :: Classification

1st school of thought

④

classifies programming information
into organized systematic manner

~~برمجة~~ برمجة

(Syntax) هي لغة البرمجة

المطلوب ترتيب دين

- درج سكرينا

(3rd School of Thought)

2023
2023

- ② (a) class is a written description of an object,
(b) class is a Template of an object.

(1) Entity :-  **class**

ایسی بودیں جو ہنپاکی چیز جسے آپ خاصیت بتا سکتے تھے

-C3) An Entity

(Q) Attributes :- Data Members

کاروں کے - اسیں جو بھی خدمات اہم ہیں

ادبیوڑہ record مردانہ اجاتی سے کس سلسلے تک

(3) Behaviour:- → operations

Operations
اُس کا طور پر اُن کاروئیہ بیسا ہوگا۔ یعنی وہ سس طرح
میں اُس کو پڑائیں سے مرے کا Member Function

```
class student {  
    int roll No;  
    String name;  
    void input();  
    void output();  
}
```

The code illustrates a class definition for 'student'. It contains four members: two integer attributes ('roll No' and 'name') and two operation definitions ('input()' and 'output()'). A brace on the right side groups the attributes, and another brace on the left side groups the operations.

object :-

Physical existence of a class is called object.

IDE:-

stands for integrated Development Environment.

// Entity set is mapped into class

(3) 3rd School of thought :-

(3)

class is an **ADT** mean
(Abstract Data Type)

↓ ↓ ↓
1 [Declaration] 2 [Declaration of] 3 [Declaration of]
4 [of Data] encapsulation operation

اگر کوئی اپنے کام میں دینا نہیں چاہتا تو اس کو Custom ADT کہا جاتا ہے۔
کوئی اپنے اپنے دینا نہیں چاہتا تو اس کو Abstract DT کہا جاتا ہے۔
لورف ADT

For example :-

تم اپنے سکونتی کا دس بنائی پس۔ اس میں سکونتی کا نام، روپیہ، رہائش۔ اس کے والر کا نام یعنی اس کے ایم ایڈ وہ لکھتے ہیں۔

Java کے

پندی **fundamental** اس کا دینا کرئے ہے۔
جیسے Abstract DT کو اس کے دینا کہا جائی اور اس کے fundamental کو دینا کہا جائی۔

ویٹھائیں

fundamental

Void, String, float, int

double - دینے

جس طرح دینے کی میں اس کو تا توسرے if مناسبت نہیں دیں

- جس طرح دینے کی میں اس کو تا توسرے if مناسبت نہیں دیں

17-08-23

8 (Data Types)

Data:-

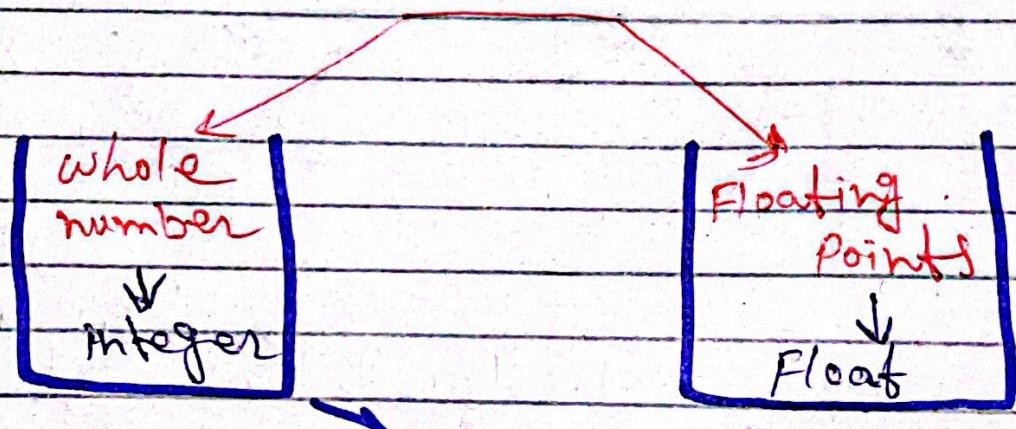
The Raw information is called Data.

information:-

After processing data or material is called information.

Data Types

1) Numeric Data Type:-



Sub types of integer

Byte	1 byte	-128 to 127
short	2 byte	-32768 to 32767
int	4 byte	
long	8 byte	

Sub types of Float

Float	4 byte
double	8 byte

2) Boolean Data type:-

- false

- True

$== / < / > / >= / <= /$ ← used in

3) Character Data type:-

- 2 byte

Example:-

• @, !, :, ;

(1)

- (1) 2 byte characters in keyboard

BCD

Binary coded
Decimal.

ASCII → 7 bit

stands for American Standard
for Coding Information Interchange.

ASCII → 8 bit

uni code → 16 bit

uni cod Text formating

UTF - 8 bit

UTF - 16 bit

UTF - 32 bit

important:-

variable, Constant

memory
constant

literal
constant

4) String Data Type:-

اے جس میں 16 بیٹس میں چار دوسرے ہیں۔ "Pak" string
constant 16 bytes - اس میں 16 بیٹس میں "Pak" ہے۔

Pakistan combination of characters is called string.

Primitive / String (جاوہ میں) = primitive / string (کمپلائیر میں)

5) void Data Type:-

ایسے دیٹا تائپ کے وقت اسکے لئے جب تم کو (Return) کے لئے جب تم کو (Instructions) کو execute کرو۔ اور اس کو execute کرو۔

Member access Modifier

-) Private •) Default
-) Protected •) Public

کنٹیننٹ =

J: یہ APP کے Company نام ہے

- Domain نام ہے

ایسا APP کے نام رکھتے ہیں

Exp:

CAS.org.PK.com

New PK = PK.org.ca

first demo

Accessible یعنی موصوف ہے

private یعنی موصوف ہے

Accessible یعنی child ہے

Accessible یعنی protected

Accessible یعنی public

1st Pillar of oop:-

~~Abstract Data Type~~ :-

1) Encapsulation:-

- Class Library Developer
- Application Developer.
- Set of Pre defined rules is called Protocol.

Example of encapsulation:-

دیکھو، ہر جو ایسے سیکھ دیکھو۔ یہ مخفی دیتے ہیں۔
Data abstraction and data hiding
is called Encapsulation.

Hiding of unnecessary data called
APK:-

stands for Android Pakage
Kit.

Pakage:-

The combination of
classes is known as
Pakage.

ایسے سچے کوڈ ہے جو کلasse بنائے جائیں اسے ایسے دیکھو
کہ کلasse کا کام کیا ہے اور کلasse کا کام کیا ہے۔

Setter:-

یہ وہ فنکشن ہوتا ہے جو اسے دیکھو جو دیکھو دیکھو
سکو رکھ لے دیکھو۔

Getter:-

یہ وہ فنکشن ہوتا ہے جو اسے دیکھو جو دیکھو دیکھو
کا اماؤنٹ باتھر نکلو دیکھو۔

(Basic oop)

Constructor:-

special member function of the class is known as constructor.

Ex:- `student student = new student();`

1) constructor Identifier must be a class Identifier.

⇒ implicit:-

2)

- خود کرپے compiler پر لئے

- کوئی explicit نہ دینا (Implicit)

- دوڑھ سے عالی (new)۔

- ترکیب کے object کے لئے

- (.) operator اور (.) Member Access operator

[Professors]
[مدرسین کی جانب]

⇒ Stack memory:

value type پر مبنایا جائے

- (S) signs limited by stack of APP

- (D) declare (S) signs ؟ i.e. (M) heap

⇒ Heap memory:-

(D) declare (S) signs ؟ i.e. (M) heap

Reference Type:-

`Student *s;`

Reference
Type.

`s = new Student();`

Boxing lib from stack to heap

unBoxing lib from heap to stack

Reference type programming

C++ for initialization (initialization) variables

topic in C++

جیو دیکلئر اس میں اسی دیکلئری کی
تو اس اور اس کے ساتھ میں اسی دیکلئری کی
اکسس اور اس کے ساتھ میں اسی دیکلئری کی
اینیٹیلائزیشن کی

Ex:- ① int a; دیکلئر اس اور اس کی اینیٹیلائزیشن

② int a = 20; اگر دیکلئر اس کی
اینیٹیلائزیشن کی تو اس کا اسکے مطابق

Ex:-

Private int b; دیکلئر اس کی
Public int c; دیکلئر اس کی

Private float a = 40.1; اینیٹیلائزیشن
 کی تو اس کا اسکے مطابق

ion :-

eration :-

new int;

new student();

APPLICATION APP

Dedicated
memory

stack

share
Memory

Heap

Second Pillar of oop :-

2) Polymorphism

Many

One

Types:-

-) overloading (Adhoc) Polymorphism
-) overriding

1) over Loading:-

Multiple copies
of the functions with same

name but different signature.

فناشر کی اس سے زیادہ میں اسے اردو =
نہیں لے سکتے جیسا میرا میرزا نہ سمجھو

overloading

- No of Parameters change
- Data type of Parameter change
- Sequence of Parameter change

Question:- اور لوڈنگ یوں اعمال رہے:-

اپنے کو یہ کام کر کے سروت پڑھ کر اسکا
کام کرے گے

Function overloading:-

ایسی اور بڑی تل کو فناشر میں ہوئی ہے اس فناشر
اور لوڈنگ کے لئے

constructor overloading:-

او، ایسی اور بڑی تل کو فناشر میں ہوئی ہے اس فناشر
اور لوڈنگ کے لئے

(Types of explicit const)

1) Default constructor:-

اے ایسی list of Parameters کو نہیں دیں
- (0) ہے نہیں دیں Default ہے

2) Parametrized constructor:-

اے ایسی Parameters کو دیں
- (n) ہے Parametrized constructor ہے

Student student = new student (roll: 16, class: 2H);

3) Copy constructor:-

اے object کو دیں جو دیں جو دیں
کوئی کوئی کوئی کوئی کوئی کوئی Reference

Static

ممبرز static ہوئے ہیں وہ
بینر object کا ایک کال ہے جس
کلاس کے ایام کے ساتھ۔

Non static

ممبرز nonstatic ہوئے ہیں وہ
بینر کال ہے جس کا object
بانی لازم ہے اور اس کے
لئے object کے لئے

سینکڑے دو امتیازی

constructor

construction

default, empty

Implicit constructor ہے جس کو
سادہ تر ہے implicit con

(Types of class Members)

1) instance Member:-

- Non static → غیر ساکن → جس کا تعلق object پر ہے
- Non shared → فرمنٹر کر مثال = طور پر اس کا ایسا عمل کا نام رہ لیجئے جس کو ہر object پر ملے گا

2) Non instance Member:-

- static ← کیا یہ keyword with this ہے؟
- Shared shared → یہ object پر ہے؟
- ڈیلٹر، ڈیلٹر، ڈیلٹر، ڈیلٹر، ڈیلٹر

Private static int count;

Public static void PrintRules() {}

Non-static Member Access

ReferenceVariableName + MemberAccess

operator + MemberName

ReferenceVariableName . MemberName

oldCar . SetModel("2022");

Static Member Access

ClassName MemberAccess operator MemberName

ClassName . MemberName

student . PrintRules();

System.out.println(ClassName . static MemberName);

Temp Shadowing:-

- Two or more identifiers same جو بے جو same
 - Two or more identifiers same جو بے جو different
- مثال میں shadowing ہے جو کوئی ترمیم فرما سکے
- d. جو shadowing ہے اس کا this.
- جو کوئی this نہیں فرما سکے اس کا different ہے
 - میں اس کا object فارینٹ ہے جو by this فرما سکتے ہیں
 - اس کا This فرما سکتے ہیں اس کا This
 - اس کا This فرما سکتے ہیں اس کا This

Exp Public void setTeacherName (String teacherName)
 { this.teacherName = teacherName; }
 حوالہ دیں

ایک کامیابی کو this پر static
 کا تعلق فرما سکتے ہیں

جو shadowing ہے = this

= فارینٹ object

this by default فارینٹ object

- اس کا جو

(سچھے سا) use کرو جو ~ static

- اس کا جو declare کرو

initialize initialize = static block

کیا static ہے Java

- اس کا جو static block

Private static int count;
 static{
count = 50;
 }

important

نکا - سوئی کرسن { مسل کو نہ تو ۰۰ علاس
جوا تمبر بے - گلول حکمی دین

اد اکر سی فنلندی پر سینز } میں کھوئے جائے
تو وہ فنلندی والوں پر سینز بروتاں ۔

—(Naming Convention) —

۴- سُرکرد فنداشت، یعنی کارهای میهمان نویسی و اداری را در یکجا میگیرند.

1) Pascal case :-

جتنی ورد آرکیوں میں case یا character کا لکھنا ہے۔

Example:- character (م) مقطوع

Sliver List Child Builder Delegate

2) Camel case:-

اگر اس کی وردی چھوٹا ہے تو دوسرا وردیز کا
برداشت کیا جائے رہے گا ۔

Example:-

Silver List 11/11/11

$\text{U}_{\text{go3}} \leftarrow$

فک سوپا نیک پاسکال کافه (Pascal cafe) 

٥) مهر زار، لونه و نمی ایشان camel case لایبل

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

3) Capital under SCONE Case:-

~~Ex: Public static final int BOLD = 1;~~

- the $\langle \psi |$ mode of ϕ is ψ in

Public static final String UNIVERSITY_NAME

This is under score

Main Function

جادہ ایکٹریم سے بچوں کا
Main Cr. Cm. - Cm. Cm. Cm.

Top Level Class

Top Level
• عکس TOP یعنی هر دو فایل هم در یک پوشه قرار گیرند.
• same که آنها را در یک فایل خانام می‌نامند.
• هر دو فایل باید در یک پوشه قرار گیرند.

~~exist entry GAPPSSS Main~~

code snippet

کنسول پر input & output لینے کا طریقہ

سچی بھی کلاس میں سب سے پہلے ڈیٹا میمبر بنانا سر کان کے لیے اور سیٹر بنانا چاہئے ہے۔ پھر اسے دیفائل کنٹرولر بنادیا جائے گا۔

Scanner System. a
M11NO =
Sout

Name

Common command-line interface (CLI)

کامپیوٹر لائبری انسٹریوٹر " کا مطلوب (IDE) میں جو
کنسول نظر آتا ہے۔ اسے input میں دینا
اور آورڈ بیٹ خاصل کرنا ہے

Based - CLI (جی پی اے)
Based - GUI (جی یو اے)

اور اسی طرح اس میں الین پیٹ رکھوادیں
رسن کے بعد نسیکے آئند اور OutPut کے نام کی شے منتکش بنائیں اس میں آدھ پیٹ لے دے۔

لائی انٹر ڈائیز " فا مطلب (IDE) میں Input میں نظر آتا ہے۔ اس میں ورث پیٹ حاصل ہوتا ہے

(Small overview Non-oop)

Scanner scanner = new Scanner(system.in);

Anonymous:-

int dataAny = new Scanner(system.in).nextInt();

* Anonymous:-

کتابخانے کرنے والے Scanner کو Scanner() input نہیں تھا۔
آخر سے دو مرکزی اور سطح پر ترتیب کئے تو ہم object کو ہم
منزہ ساز کو اپنے reference پر لے کر اسے اپنے object کے طور پر
انواع کا کوئی نام نہیں ملے۔

Anonymous

آخر سے دو مرکزی اور سطح پر ترتیب کئے تو ہم object کے طور پر

Example - Scanner name = new Scanner(system.in).nextLine();

name = new Scanner(system.in).nextLine();

(Operators)

operator Def:-

Definition:-

A symbol or Ad
those guides to
for a certain
Perform specific compi
(Op, ex)

ایک علامت یا دیاں identifier کی خاص operation کی طبقات کو کہا جاتا ہے۔
compiles to operation
operator کی طبقات کو کہا جاتا ہے۔

operation
جو خاص کو کاغذ پر کرنا ہے۔
Compiler

i) unary operator:-

Definition:-

A symbol or identifier which there guides to compiler for a certain operation.

Perform (with compiler) (operator) (Opnd)

i) unary operator :-

(Opnd) (operator) (Opnd)

Example = -10, -a, +a, +10

++a, a++, --a, a-- etc.

ii) Binary operator:

(Opnd) (operator) (Opnd) (operator) (Opnd)

Ex:- a+b, a-b, a*b and etc.

Types of Operator

- 1) Arithmetic operators
- 2) Relational operators
- 3) ~~Bitwise operators~~
- 4) Logical operators
- 5) Assignment operators

iii) Ternary
Op: 2, 3
Expressions

operators wl

? : (Sijri)

→ (W W)

" false

~~an expression~~

Expression:-

The combination of operators and operands which evaluate to a single value is called Expression.

Ex:- $a+b$

(1)

8(Arithmetic operators)

- Addition + ↳ (W, side effects W)
- Subtraction - ↳ Multiplication *
- Modulus % ↳ Division ÷ /

(2)

iii) Ternary operator

Ternary operator
C: 2, JS: 6 operators (?:) & operators ==
C: 1, JS: operator ?: (Sister)

Expression:-

- c_1, c_2 conditions $\Rightarrow (c_1^* c_2^*)$

-) true
•) false

Expression:-

The combination of operators and operands which evaluate to a single value is called Expression.

$$\text{Ex: } a+b$$

(1) Arithmetic operators

c. gen. side effects (w)

-) Addition + •) Multiplication *
 -) Nafî - •) Division ÷ /
 -) Modulus %

(Relational^(f) operator)

Boolean logic operators

-) equal == •) greater than >
 -) not equal != •) less than <
 -) less than or equal <=
 -) greater than or equal - >=

$\text{f} \circ \text{g}$ relation ($\text{f} \circ \text{g}$) opens \Rightarrow

اہم بات = Expression of condition

عکسیل = what is side effect of operator

Question ۱۵: من فرقة من Expression , و condition

Condition:-

- Boolean operation لیکری بولین
 - True or false Condition Expression
 - All conditions are Expression
 - But All Expressions are not condition.
- Def:- condition is an Expression that returns logical values are called condition.

-:Casting:-

وہ تابعی و نکودھی کسی Casting کی

کیا ہے Casting

implicitly بیانی جو compiler

کرتا ہے میں تابع کیسے

اے up-casting compiler

کر سکتے ہیں اسے down-casting

3) (Cast operator)

implicit casting:

double a = 34;

int b = 34;

a > b;



کوئی کسی کو بلایا جو compiler پر علیم

- implicitly ہے

Important Questions

- 1) Differentiate b/w condition and Expression?
 - 2) side effect of operator?
 - 3) What is short circuiting?

casting :-

f) d;
 $\sin(\ln(d*i))$

4) -Logical operator

- جملہ کی تکمیل اور جمع کیا جائے گا

What operates?

(i) ! its a unary operator.

$$(ii) \underline{BB} = - \quad (iii) \underline{II} : -c_1 \underline{c_2}, \text{ result}$$

C_1	C_2	R
F	F	$F = F$
F	T	$T = F$
T	F	$F = F$
T	T	$T = T$

Operations سے 11 = 88

Questim ha Ya

(short circuiting)

condition 1 } condition 2 } corol 3
 88 88

88

88

Corol 3

11

11

Ans True

جو، اگر کوئی read کرنا چاہے تو جس کوئی compiler جو 88c
کو check کرے تو اس کو اسی کوئی راستے پر اگلے کوئی ہے تو اس کو اسی کوئی

لیکن اگر $x = 1$ تو $x > 0$ باشد اما $x^2 \leq 0$ نباشد.

وہی کو read کرنے والے compiler ہیں 11

- Click True

Explicit casting:-

int a = (int) d;

System.out.println(d * i)

4) -&(Logical operator)

! its a unary operator.

BB :-

(iii) || :- C₁ C₂ result

C ₁	C ₂	R	
F	F	F = F	
F	T	T = F	
T	F	F = F	
T	T	T = T	

Question la yq

Q. = Q(side Effect of operator) :-

official [ɪ'fɪʃl] operator [ə'pəʊtə] نُورَسْ الْعَوَالِي الْعَوَالِي الْعَوَالِي الْعَوَالِي

٣) operon کو ایجاد کرنے سے
وہ مکروہ نامیں کہا جاتا ہے۔

Side effect Example:-

Let :-

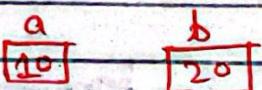


بسا لک برو بیری ای جبل (b) کی ویلے

21 جو گئی مطلوب ساختہ الفیلڈ ہے۔

No side effect Example:-

Let ..



$$a+b;$$

$$10 + 20 = 30$$

5) Assignment operator

ii) Arithmetic wise.

$$0) = \quad 0) + = \quad \underline{B} \quad , \quad \cancel{\frac{1}{x+1}} = 1$$

٥٣) حساباً غير المنهجية

$$9) \quad 1 = \quad 9) \quad \% =$$

•) •)

iii)  bit wise

$$\textcircled{1} \quad 8 = \quad \textcircled{2} \quad 1 =$$

$\sigma_A = \sigma_B =$

o) << = o >>=

These all are called Compound Assignment operators.

Example: Public void Data(string myname ,string name){
sister =>Data(s, name); equals(s, name);

Example - لے کر compare کرو جو دو چیزوں کو (objects) (.equals)

System.out.println(myname==name) - لترجع object.equals =

Precedence of operator.

→ It's important to know Precedence & Assignment Operator complexity (1) C, C++, C#, VB, Java etc. will follow
 (2) In Precedence (1+2)*3 solve 1st then
 $\rightarrow | - | + | \div | * | \% |$ → - +, ÷, * , %
 ↳ Right to Left ↳ Associativity (3) C, C++, C#

Associativity of operator:-

→ Same Precedence for expression in C?
 compiler's (3) Preference of left to right
 ↳ Left to Right (4)

(Prefix operator)

1. $a++$ پہلی تھس
 2. $++a$ دوسری تھس

(Postfix operator)

int a = 10;

int b = (a++) + j;

← →
 Convert Any number to binary.
 Example:-

int a = 90;

int b = 1090;

System.out.println("a:" + Integer.toBinaryString(a));
 System.out.println("b:" + Integer.toBinaryString(b));

Sept 2021

ASCII Number System

Binary = 0 - 1

Weight * Value

Answers

11

Octal = 0 - 7

Answers

111

Decimal = 0 - 9

Hexa-Decimal = 0 - Fifteen

0 - 15

0 - 15

کامپیوٹر کا ایک اس طرح کا دسی جزوی سیستم ہے جو اس کا مجموعہ ہے۔

Bitwise Operations

Classification:

1) Bitwise AND operation
2) Bitwise OR operation
3) Bitwise XOR operation
4) Bitwise NOT operation

Bit by Bit operation کہا جائے گا۔

② BCD stands for Binary coded Decimal
 $BCD \rightarrow 7 \text{ bit system}$

③ ASCII $\rightarrow 7 \text{ bit}$

0 - 127

ئے 128

④ ASCII $\rightarrow 8 \text{ bit}$

0 - 255

ئے 256

⑤ Unicode $\rightarrow 16 \text{ bit}$

0 - 65535

ئے 65536

(Bit wise operator)

1) 8 truth table Bit wise exclusive OR

0	0	0
0	1	0
1	0	0
1	1	1

2) Λ ⁻ exclusive or

0	0	0
1	0	1
0	1	1
F	1	0

3)	11	0	0	0
	on	1	0	1
		0	1	1
		1	1	1

(Data) Mentalization

Definition:-

بے ایک ایسے Term کے جس میں دُشنا رسانہر د کمردیا
جانا ہے اور اس کی افادت کو برخادرا باتا جائے۔
اگر اس کے Datei Maturization سے ہیں

Font style

~~Wie kann man stabilen Grundwasser~~

جواب عبارت expression (س. ۱۵) کی تینوں ترکیبیں

Example = 1, 2^o, 4^o etc.

↳ ~~Memory Constant~~

وسائل انتقال → Means

~~→ size is final & non-static~~ (

- final یا last-Stage
final یا last-Stage

Example:-

• Public static final int BOLD = 1;

وَرَأَيْتُمْ أَبْرَقَ code رُونَّا وَ(دوال) عَلَيْهِ تَعْمَلُ مُنْ

لارنس-زیر سلحہ بنی سے final ۱۳ افغانیت تھے

4) Left Shift operator:

int a = 10;

a \ll 1;

- جنہی کو left shift operator کہا جاتا ہے
 اور جس کو right shift operator کہا جاتا ہے
 - لیکن جو side effect ہے

int = 4 byte = 32 bits

float = 4 byte = 32 bits

Byte = 1 byte = 8 bits

Short = 2 byte = 16 bits

Long = 8 byte = 64 bits

double = 8 byte = 64 bits

5) Right shift operator

int a = 10;

a \gg 1;

- جنہی کو right shift operator کہا جاتا ہے
 اور جس کو left shift operator کہا جاتا ہے
 - 16 bits کو 8 bits کا معاون کرنا ہے
 - 3 bits کو 8 bits کا معاون کرنا ہے

6) Unsigned Right shift operator

Magnitude bit + anti-magnitude

a $\gg\gg$ 1; • Total wait is called magnitude.

i) Most significant bit

128 64 32 16 8 4 2 1



• bit number سے آخر والی bit کو most significant bit کہا جاتا ہے
 - sum bits for magnitude wait کا سامنہ ہے
 - یعنی sign bit کا سامنہ ہے

3) Compliment operator (\sim)

- \sim Bu, \sim (takig), operator \sim (c)
 - \sim (takig), \sim unang \sim (c)

Example =

```
int a = 10;
System.out.println( $\sim$ a);
```

Answer = -11



String Type Building Type ← Built in type

• Constant string literal pool.

String name = "Ali";

String name1 = "Ali";

- \sim (takig), \sim immutable object is string

• System.out.println(name == name1);

Compared, \sim objects \sim \sim HEEP \sim

- \sim (JVM), address? \sim \sim \sim

• System.out.println(name.equals(name1));
 \sim (Compared objects) \sim \sim HEEP \sim

• System.out.println(name.toLowerCase());

• // // // (name.toUpperCase());

• // // // (name.charAt(0));

• // // // (name.concat("zindabad"));

• // // // (name.contains("pak"));

• // // // (name.endsWith("fan"));

• // // // (name.indexOf("a", 2));

• // // // (name.substring(2, 7));

StringBuilder stringbuilder = new StringBuilder();

Flow Control Statements

Conditional Block.

if :- set of statements
or set of statements which execute
else :-
set of statements after else, if, else if

Example:-

```
int marks;  
System.out.println("Enter the marks = ");  
marks = new Scanner(System.in).nextInt();  
if(marks >= 60){  
    System.out.println("Pass");  
    System.out.println("Well Done");  
    System.out.println("Keep it up");  
}  
else{  
    System.out.println("Fail");  
    System.out.println("This is because your non serious  
behaviour");  
    System.out.println("Try once more and get success");  
}
```

True block

False block of else

```
StringBuffer wazai = new StringBuffer();  
wazai.append("Pakistan");  
wazai.append(" is ");  
wazai.append(" good ");  
wazai.append(" country ");
```

It is called compound statement
braces { } . operator .

- of nested if) >

Nested if → True Block

(Nested else if)

Nested `else` is true you set false if it's true

Nested else if → False block.

(Explanation Nested if)

Differentiate b/w the behavior of Nested if and Logical and &

اگر اس کا نتیجہ لیٹھ بچھ میں ہے تو اس کا نتیجہ اس کا نتیجہ

Nested else if

Writing if & If Possibility & else if

Example - If needed else if - If, Else
if (marks >= 80) {
 cout
} else if () {
} else if () {

Dangling if

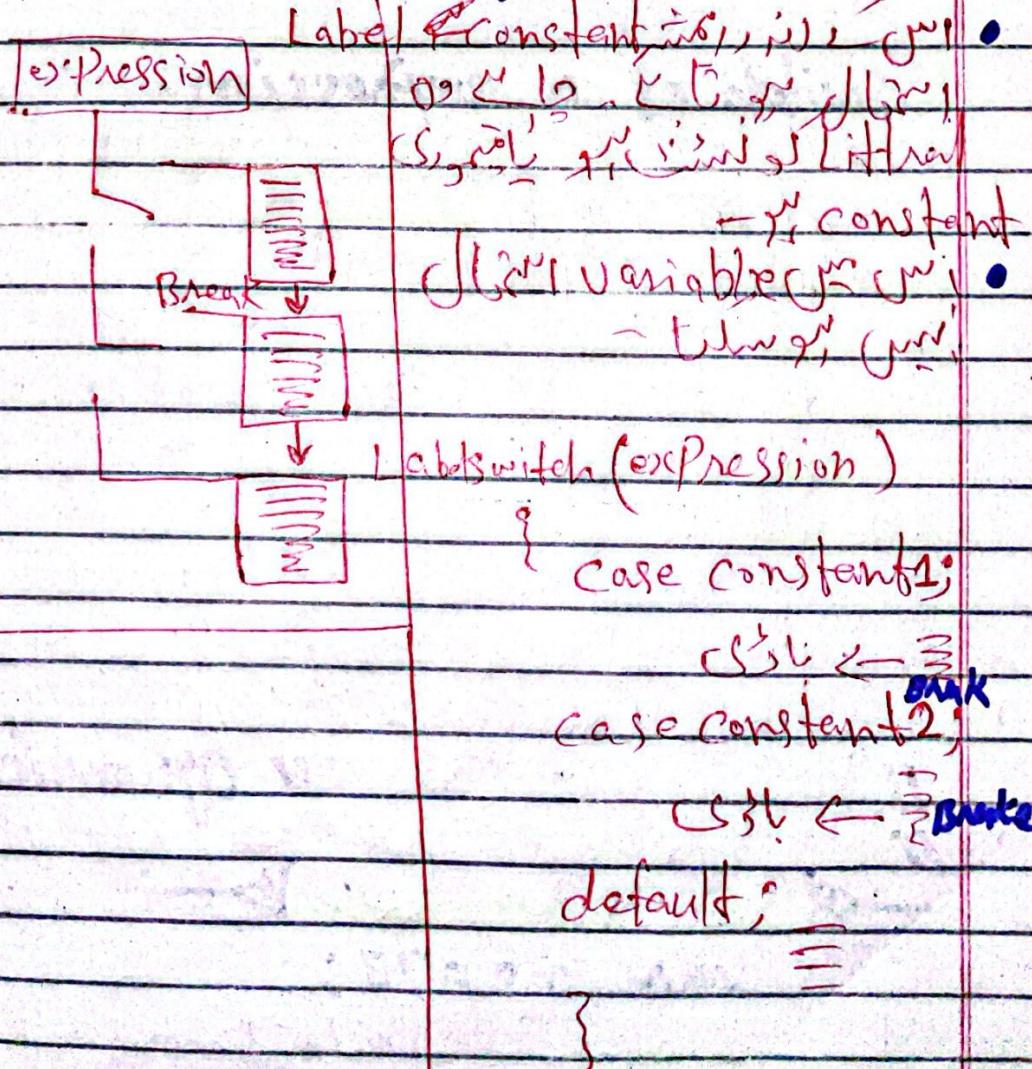
عندما يكتب if و لا يوجد else في نفس الكائن

Public

(switch)

عندما يكتب switch و لا يوجد nested else if
أو break أو case أو default

Flow chart of switch



Primary expression:

1. (primary expression) نہیں تو choice expression (choice expression)

Exp: switch (choice) {

Exp 2 - choice expression

choice 2 طرف سے ساختے تو اسے choice کہا جاتا ہے۔

. (لماں استعمال ہوتے) switch

switch کی صورت میں match, result, expression کو کرنا ہے۔

else if () ; (multiple times)

Switch as a expression.

Lambda expression = .

return switch (operator) {

case '+' → getfirstOperand() + getsecond();

case '-' → 11 - 11;

case '*' → 11 * 11;

case '/' → 11 / 11;

default → 0.0;

}

// Arrow Function

// Lambda Expression -

Type inference

var n = 78;

- t

شیوه اولیه برای اینیتیلایز کردن اسکرپت
که باید بیند شود

Loop

Type :-

- i) conditional loop
- ii) counter loop

شیوه اولیه برای اینیتیلایز کردن اسکرپت

Example - ۱

i) Condition Loop :-

Type

SubTypes

i) while

ii) do-while

• Pre-conditional Loop

• Post-conditional Loop

Loop شرطی است که در

نهایتی

while (condition)
statement;

ii) Counter Loop

Type

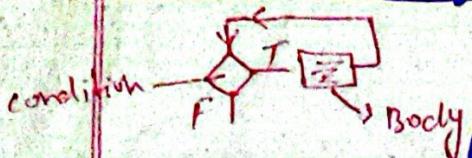
• For

• For each

while Loop

در این شیوه ابتدا شرط اینیتیلایز شده است
و سپس دستورات اجرا شوند

Flowchart diagram



(Pre condition/ while Loop)

scanner.hasNextInt()
int o = scanner.nextInt();
if(o < 0) {
 System.out.println("minimum value = " + o);
}

Example of conditioned Loop

— syntax —

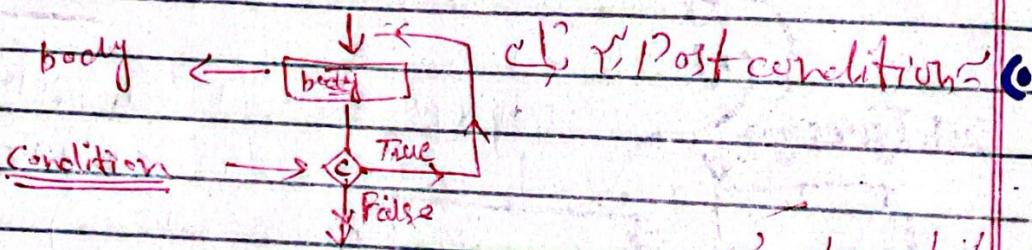
while (condition){

 statement;

}

(Post conditional)

— do(Do while Loop) —



— syntax —

do { body }

 while (condition);

↳ (one) minimum

↳ Compiler reads loop
↳ condition in while loop
↳ loop if true, read
↳ loop if false, break

second type of Loop.

ii) Counter Loop for loop.

سادہ سلسلہ کا نام loop ہے
 اس کو initialize کرنے والے دو ٹکنیک
 یعنی گلوبال compiler (C/C++) میں
 library کا نام true (1) ہے جو true
 کا لب (1) update کرے

Syntax

for (initialization ; condition ; update)
 statement;

}

(Nested Loop)

For loop کا یہ سکھیں
 اس کا list ہے
 - 1D Nested Loop (1D), one Dimension
 - three dimension also 3D in real world

Example:-

1	2	3	4	5	6
---	---	---	---	---	---

← X-axis →

1D loop 1D loop also oneD

Example of 2D

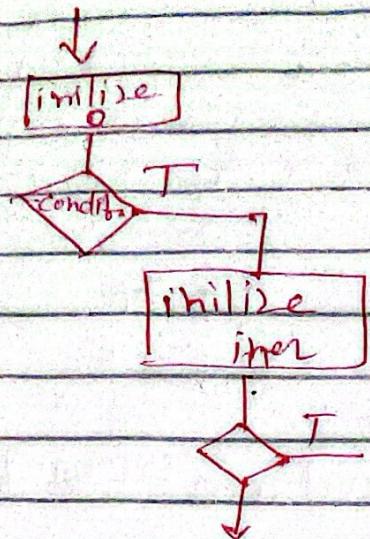
1	2	3
4	5	6
7	8	9

2D

Nested for loop

دو لوبی اسے 1D کا کہا جاتا ہے

(Behaviour of Nested Loop)



iteration (3 Outer)
Do for inner
2D grid

Example for Loop Nested Diamond

```
int length;  
int start, mid, end;
```

```
public void input() {  
    input li user say length me  
}  
public void output() {  
    start = mid = end = length / 2;  
    for (int row = 1; row <= length; row++) {  
        for (int col = 1; col <= length; col++) {  
            if (col >= start && col <= end) {  
                sout.print("*");  
            } else { sout.print("space"); }  
            sout.print("\n");  
            if (row < mid) { start++; end++; }  
            else { start++; end--; }  
        }  
    }  
}
```

Image Processing

• It's a process of completing an image or compressing it.

Types :-

- vector images
- hosted images → bitmap images
- vector images

Compression Algorithms of bitmap images

- TIF
- PNG
- TIF
- GIF

(Lossless bitmap images)

W compression ratio

— high

- vector images ka file format SVG HTML XML

— error

→ runtime exception

Exception handling in use of bitmap images

• Bitmap read from disk

• Buffered image

• File from ram → image object

• File read error → runtime exception

→ try catch block Try catch

Syntax:

BufferedImage sourceImage = null;

File file = null;

try

{

} catch (Exception ex) {

3

Parent class \rightarrow 3 Exception classes
child class \rightarrow (پدر کلاس \rightarrow ۳ اکسپشن کلاس) \rightarrow (بچه کلاس \rightarrow (پدر کلاس کی ۳ کیوں اکسپشن کلاسیں ہیں۔)

(•) inheritance generalized to specialized
hierarchy.

specialized \rightarrow (بچہ کلاس) catch block first
generalized \rightarrow (پدر کلاس) catch

finally block first \rightarrow finally block
finally block last \rightarrow finally block

Row = Height

col = width

File extension

RGB for Ping

RGB for JPG

Hexadecimal tea littoral constant = 0xFF

→ 80 for filter image processing 80

Quantity of pixels in a pixel \rightarrow (کلریز پیکسل کی تعداد) RGB
negative \rightarrow (نیگیٹیو) RGB

oop Based implement image Processing

With you to find is oop based implement
in image processing

Reference types

- class
- string
- Array

System{

read & image lib.

FileUtil class

write2, read2

FileUtil class

effect, effect

FileUtil class

image, image

FileUtil class

proper, proper

~~out.print~~

(Data structure) :- Collections

Array:-

The collection of homogenous data type is called Array.

Homogenous → same data type

Definition:-

The contiguous block of same data type is called Array.

old style of Array

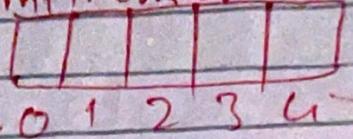
int List[] = new int [5@];

New style of Array

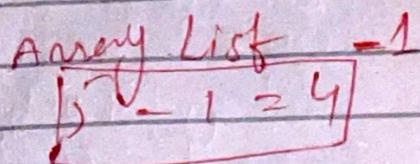
int [] List = new int [5@];

Last index

↳ implementation concrete (list)



lower bound



↳ List [0] → This is subscript operator

Lower Bound check

↳ $\text{int } [0:2] \{ \dots \}$ if $i = 2$ then $i < size$
 initialize $\{ \dots \}$ if $i = 1$ then $i < size$
 $i = 1$; if $i = 0$ then $i < size$

↳ contiguous → in a single line.

Upper Bound check

↳ $\text{int } [0:n]$; $[]$ → subscript operator
 $\text{int } [0] = 1;$

↳ $1 \rightarrow$ Assigned 0 value
 initialize $\{ \dots \}$ if $i = 2$ then $i < size$

$\text{int } [] \text{ arr} = \{ 1, 2, 3 \};$

↳ initializations

redirection $\{ \dots \}$ if $i = 2$ then $i < size$

$\text{int } [] \text{ list} = \text{new int } [10];$

Two Dimensional Arrays

int [] [] List = new int [3][3];
syntax

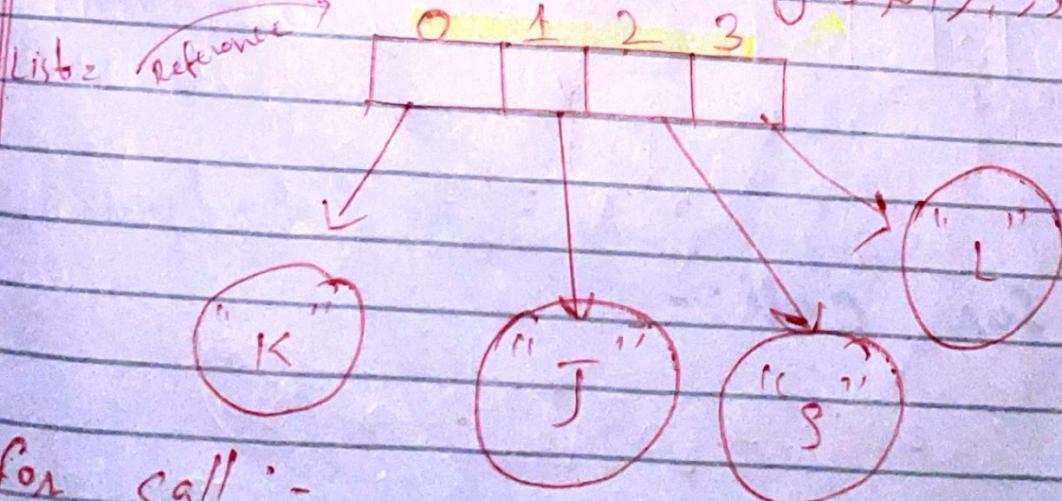
Multy Dimention on Two Dim

int [] [] List = new int [3][3];
Last row index 3
Last column index 3
int 0 < i < 3, 0 < j < 3

Single Dimention Array in String

string [] List = { "K", "J", "S", "L" };

چار ڈائیجیٹ لیٹریل اے جو اس کا ایک object ہے اور یہ object کا interior جو اس کا میراں ہے اس کا reference ہے



for call :-

sout (List[0]), sout (List[2]);
یہ اس کا call ہے جو 0 کو یہ کوئی کام کرنے والا ہے اور 2 کو یہ کوئی کام کرنے والا ہے

Single Dimension for int

Example:-

`int [] List = new int [5];`

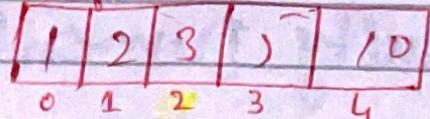
This is **declaration**.

`int [] List = {1, 2, 3, 4, 10};`

This is **initialization**.

for call:-

`cout < List[2];`



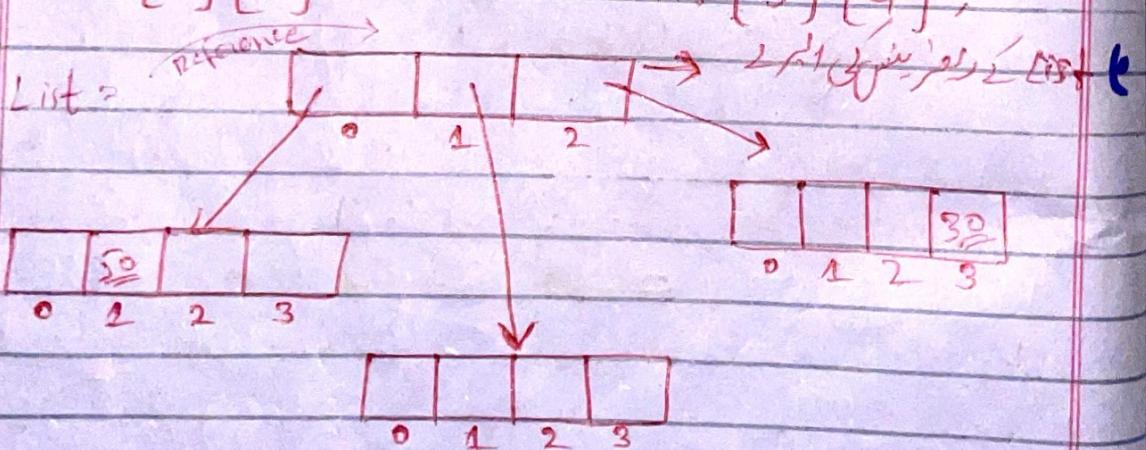
→ index

Answer: 3

(3 वाले संख्या का प्रिंटिंग करना है।)

for Multi Dimension int

`int [][] List = new int [3][4];`



for call:-

`cout < List[0][1];` → 50

`cout < List[2][3];` → 12

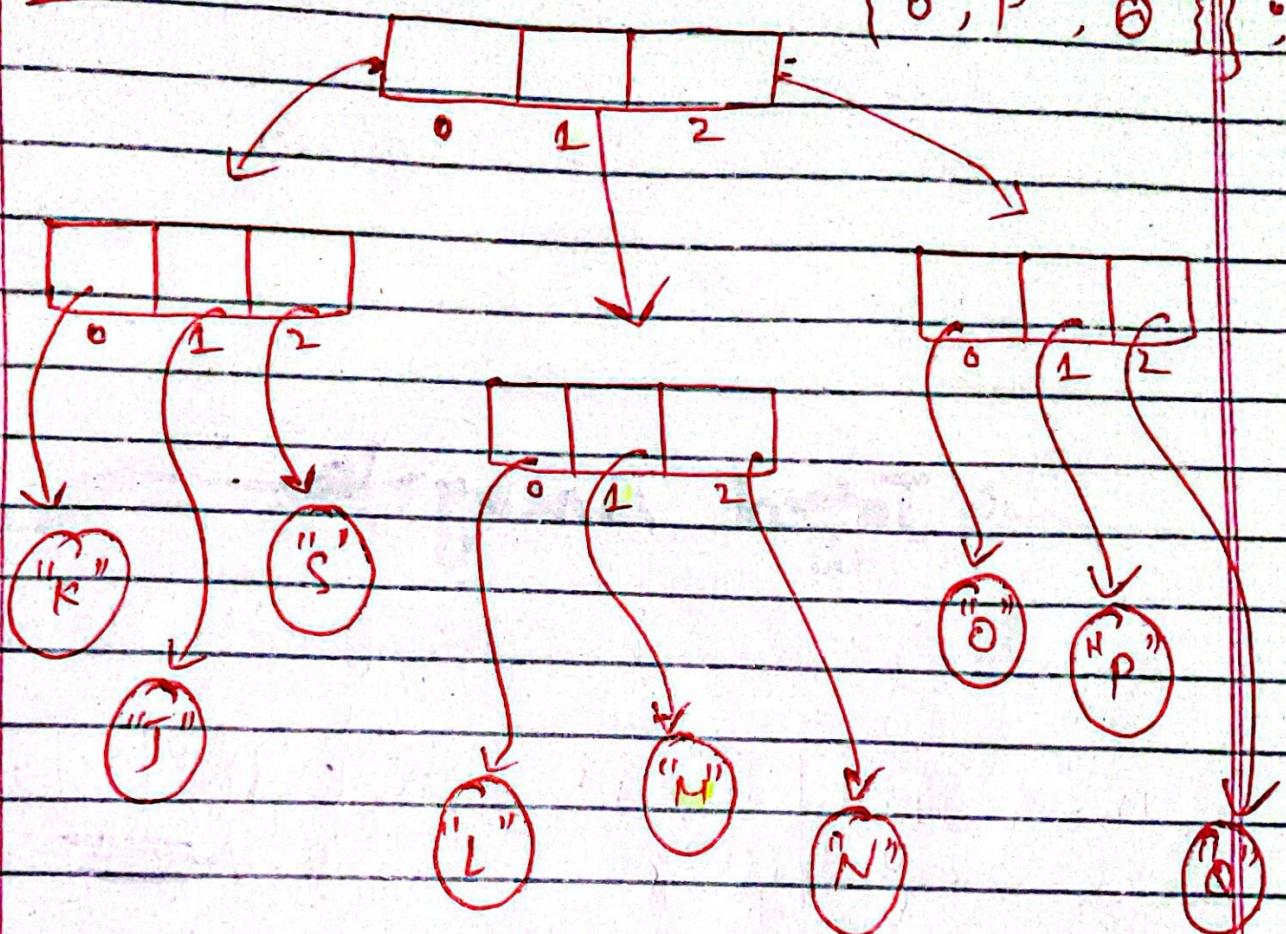
दो नियमों का उपयोग करके इनिशियलाइजिंग करें।
1. 3x4 का एक 3x3 का एक
2. 1x4 का एक 1x3 का एक

Multy Dimentions for String

Example:

String [] [] List = { { "K", "J", "S" }, { "L", "M", "N" }, { "O", "P", "Q" } };

List = Reference



for call:-

`sout(List[1][1]);` → Ans "M";



Practical implementation

FPS → means Frame Per Second

Array implementation Reference Type

، الگوریتم های راهنمایی پردازی

جواب

(Jagged Arrays)

- $\{ \{ \text{int} \} \} \cup \{ \text{int} \} \cup \{ \text{int} \}$

`int [][] List = new int [2][3];`

↓، array size() \rightarrow $\{ \text{int} \} \cup \{ \text{int} \}$, or ~~array size()~~

- $\{ \text{int} \} \cup \{ \text{int} \} \cup \{ \text{int} \}$

`int [][] List = new int [3] [];`

- $\{ \text{int} \} \cup \{ \text{int} \} \cup \{ \text{int} \}$ Jagged

که این دو نوع $\{ \text{int} \} \cup \{ \text{int} \}$ هستند و این دو نوع را با $\{ \text{int} \}$ می خواهیم
 $\{ \text{int} \}$ را manage کنیم (برای این دو نوع)
 این دو نوع Jagged هستند و $\{ \text{int} \}$ هستند

8 (ArrayList)

- `ArrayList<T> list = new ArrayList<T>();` ↳ `T` encapsulate
- `<T>` Template type generic ↳
- `ArrayList<Integer> list = new ArrayList<>();` ↳
- `<>` generic behaviour ↳
- `Template type` ↳

Properties for add Int

```
ArrayList<Integer> list = new ArrayList<>();
list.add(34);
list.add(12); ↳ Integer class
and etc. ↳ List interface
sout(list); ↳ Approach
```

Properties add string

```
ArrayList<String> names = new ArrayList<>();
names.add("Muhammad Waqar");
names.add("Ali Raza");
names.add("Noman Ameen");
sout(names);
```

remove a specific name or etc.

```
names.remove("Ali Raza");
names.remove("Muhammad Waqar");
names.remove(0); ↳ remove 0 index
names.remove(1); ↳ remove 1 index
removed ↳ remove 0 index
removed ↳ remove 1 index
```

names.remove(2);

names.remove(0);

etc.

sout(names);

Properties for copy constructor

Example:-

```
ArrayList<string> names = new ArrayList<string>();
names.add("Ali");
names.add("Omer");
names.add("Usman");
copy construct given below...
```

```
ArrayList<string> list1 = new ArrayList<string>(names);
list1.print(List1);
```

Properties for add all

```
list1.addAll(names);
```

Properties for clear

```
names.clear();
```

```
names.clear(data);
```

Java provides two types of clear methods:
 1) Clear(): It removes all elements from the list.
 2) Clear(data): It removes all elements which are equal to data.

LinkedList

singly

Many ArrayLists in a LinkList.

```
LinkedList<string> list = new LinkedList<string>();
```

Template ^{TYPE}
 ArrayList<TYPE> & list;
 TYPE represents
 list & list & Node

Example

data	node	data	data
------	------	------	------

list manage object size linked list
 list dynamic list

Properties

- `LinkedList<String> list = new LinkedList<>();`
- `List.add()`
- `List.addlast()`
- `List.addfirst()`
- `List.clear()`
- `List.contains()`
- `List.addAll()`

Real Stack Properties

- `push`
- `pop`
- `peek`

(`Stack`) stack

(`Stack`) stack

- `Operations`

`Stack` (`Java`) `Stack`

(`interface`) (`Stack`) (`method`)

- `Operations`

`Stack<String> stack = new Stack<>();`

`stack.push("Ali");`

`stack.push("Waqar");`

`stack.push("Anjum");`

`sout(stack.pop());`

`sout(stack.peek());`

HashMap

`HashMap<Integer, String> hashMap = new HashMap();`

- Inception Asymmetrical hash map.

- `Entry` Hash entry (`map`)

- Map Key Value Pair hash.

- called Map or dictionary.

- `Map` unique key (`Map`)

Q(Hashing)

By fast retrieving list
Hashing

list, unique in set

Set, unique in list

- List with unique values is called **set**.

```
Set<String> st = new HashSet<>();
```

Q(Properties)

```
HashMap<Integer, String> list = new HashMap<>();
list.put(0, "Ali");
list.put(1, "Umar");
list.put(2, "Usman");
list.remove();
and etc...
```

cout(list.get(1));

Q(For each) go loop

```
List<String> names = new ArrayList<>();
names.add("Ali");
names.add("Umar");
names.add("Usaif");
names.add("Waqar");
for (String name : names) {
    cout(name);
}
```

for (String name : names) {
 cout(name);
}

90-11/2023
Monday

(Advance oop)

3rd pillar of oop

(inheritance)

عواید اور میراث کی تعریف
Inherit: جو ایک جنگل کا کوئی کامپنی
کے لئے Reusability - ہے۔

Parent class → child class

super class → sub class

base class → drive class

• Reusability • Generalization

• اریٹیشن ہے جو is a - logically
• اس کا protocal ہے جو
• Specified of generalized ہے
• True جو اس کا ہے

Example:-

extend MCS → Student
extend Mgt → MCS
Specified → Mgt
extended → MCS
child → Student
Student → Mgt
Mgt → MCS
Student → MCS

• Multi Level Inheritance

Syntax -

public class student extends MCS
{
 //...
 //...
 //...
}
class Parent {
 //...
 //...
 //...
}

3. child (black) - (black) - (black)

~~(explicit inheritance)~~

- جیسے اس کے لئے explicit inheritance کا استعمال کیا جائے تو اس کو explicit inheritance کا نام دیا جائے گا۔

Public class MCS student extends student {

} - ~~(implicit inheritance)~~

- If -> object v (i.e.) , implicitly
object object = new object ();
object . Properties

object class (\vdash) \rightarrow \vdash \vdash \vdash \vdash object class
- \vdash \vdash \vdash \vdash inherit

Ellipses → Parent shape with child circle, rectangle → child ellipse - Ellipse with line with

Rectangle rectangle = new Rectangle(10, 20, 100, 200);
rectangle. Properties;

- Although $\langle \text{put } j \rangle \in \mathcal{S} \cdot \mathcal{I}$ Parent
 - This $\langle \text{put } j \rangle$ objects \rightarrow child

Generalization: handle objects $\rightarrow \mathcal{S} \cdot \mathcal{I}$ (C b w)

Anaylist <shape> shape = new Anaylist<>();

Object \rightarrow Properties \rightarrow Parents \rightarrow Properties \rightarrow Child
Object \rightarrow Properties \rightarrow Child; Access \rightarrow Properties \rightarrow Parents
Object \rightarrow Properties \rightarrow Access \rightarrow Properties \rightarrow Parents

↳ What is meant by CO?

- Base address of object
- Base address of Member
- Base address of Meta-data.

↳ This is the base address of the object which is the base address of the member or the base address of the meta-data.

↳ Cost-effectiveness generalization

Function overriding

↳ If a function is defined in the parent class and it is redefined in the child class, then it is called overriding.
↳ If a function is defined in the parent class and it is redefined in the child class, then it is called overriding.
↳ If a function is defined in the parent class and it is redefined in the child class, then it is called overriding.

↳ It is done within a - overriding
↳ It is done within a - overriding

Recursion:-

• self calling of a function is called Recursion.

↳ It is called self calling of a function.
↳ It is called self calling of a function.

• otherwise it is overriding of inheritance
↳ It is called self calling of a function.

~~↳ In C++ class, if Parent C is overriding
Functionally or Child C is extending C, then hide
Functionally or C is overriding function of C~~

3 Types of overriding

• virtual overriding → Dynamic binding

• nonvirtual → static Binding

• synonyms

virtual

Nonvirtual

• Dynamic Binding

• Static Binding

• Late Binding

• Early Binding

• Runtime Binding

• Compile time Binding

Non virtual overriding

Ex) If child inherit Parent, then
non virtual function is used.

↳ child reference variable can access
parent function. This is called
generalization in inheritance.

RTTI

Run Time Type Identification.