Decision control statements → for, while, do-while

Conditional statements → if, else, else if, switch

Data types and variables → int, long, string, float ...

C++, Java, python

Arrays

50 students

↓

store name

address

String s1 = "Amit"

String s2 = "Karan"

String s3 = "Rahul"

→ Homogenous Information

s50 = "____"

10 students

10000 students

Arrays

collection of similar type of elements which has contiguous memory location.

20 students

0 1 2 3 4 . . . . . .

"Ram" "Shri" "Jay" a 7 ... ...

students

students[0] → "Ram"

c a t

a20 a21 a22 . . . . . 1 byte

→ fixed size

Advantages

optimised → cost

random access → arr[19]

Java

char → 2 bytes

short → 2 bytes

int → 4 bytes

boolean → 1 bit

0 1 2 3 4 5 6 7 8 9 ← indices

Array length is 10

Disadvantage

→ fixed size ;

[I@6996d68 → reference to the first place.

Array

Integer array

int string char byte long bool

1-D Array

1-Dimensional

x-axis

y-axis

col 1 col 2 col 3 col 4

row 0

row 1

row 2

row 3

rows

matrix

X-axis

rows cols

int[][] arr = new int [3][4];

* rows (compulsory)

* cols → ✓ X

address of another array of size 4

row 0 → 200

row 1 → 650

row 2 → 720

200

650

720

cols = 4

Jagged arrays (array with diff no. of arrays of columns)

row 0

row 1

row 2

row 3

2. Given an array of 0's and 1's, sort the array.

Input → 0 1 0 1 1 0 1

output → 0 0 0 1 1 1 1

1. Approach

a. Count the no. of 1's and 0's

b. Put 0's in front and then 1's.

2. Approach (Two pointer approach)

right left

0 1 0 1 1 1 1

* Sort an array of 0's 1's and 2's.

0 0 0 1 1 1 1 1

right left

3. Given an array of integers, return true iff it is a valid mountain array.

an array is a valid MA :-

1. arr.length >= 3
2. exists i

arr[0] < arr[1] < . . . . < arr[i-1] < arr[i]

arr[i] > arr[i+1] > . . . . > arr[n-1]

Eg:

0 2 3 4 5 2 1 0

true

Eg:

0 2 3 3 5 2 1 0

false