Semester - VII - ISE :→ (C++)

1st Sem → C
2nd Sem → Python
3rd Sem → Java DSA in C
4th Sem → DBMS + Advanced Java → Collection Framework
5th Sem →
6th Sem → MEKN → (M, E, R, N) +
7th

1942 :→ Denis Ritchie
Ada Lovelace → (UNIX) OS

A + modif^n → B (ANSI)
1972 : → C [Not accepted]
(BIET | BMSCE) {B.Tech, NAAC, NBA} (AICTE)
(1989) 1st std ANSI - C

C / C++ (Compiled) → Pointers
.c / .cpp → .exe
Java / Python (How?) → Both interpreted & compiled

C + SIMULA = C++ (MCQ)
Bjarne Stroustrup → Denmark

1991 → Guido van Rossum
Monty Python Flying Circus
1995 → James Gosling C + C++ = Java
Sun MicroSystems
Oracle

WORA
* JDK → System o/s
+ env. vars path
Java - Home C/ /bin JDK 17.0
(javac)
JRE → OS
(JDK + JRE) = JVM → .java → .class → bytecode → java code o/p
JIT Just In Time Compiler Symbolic Language

Data Types :→
Integer: INT-MAX $2^{31}-1$ ✓
INT-MIN -2^{31} ✓
(W3schools)

Primary (C)
Integer → short (2) int (4) long (8)
Float → float double
Character → char
Void → void

Enumerated → enum
Named Constants
M-S J-D — SDLC
Derived → Arrays Pointers Struct Unions
[String (class)] C++
(-2^{31}) $(2^{31}-1)$

Java :→
Primitive (Simple Values)
8 categories
byte
short
int
char
long
double → default
float → f or F
boolean

Non - Primitive (Reference)
(Objects) (Complex Objects)
String
Date
HashMap <K, V>
<100, Samar>

[450 dsa.com]

Operators :→
* Arithmetic Operators: +, -, /, *, % Python: //, **
* Assignment Operators : =, +=, -=, /= - - - - -
↳ Augmented*
↳ Short hand*
a = 5
a = a+10 ✓
a += 10 ✓
* Logical Operators : &&, ||, ! { and or not }
* Relational | Comparison : >, <, >=, <=, !=, ==
(Boolean)
* Unary Operators: → ++ , --
Prefix | Postfix
* Ternary Operator:
Short-hand if-else operator:
returntype var = (Condition) ? truevalue : falsevalue;
String returntype String return type String

Binary
* Bitwise :→
(i) AND → & → Ampersand
(ii) OR → | → Pipe Symbol
(iii) XOR → ^ → Caret
(iv) Right Shift → >> } Angular Brackets
(v) Left Shift → << }
(vi) NOT → ~ → Tilde or negation

int int ↓ Bit Operation → int
1001 0001

5&7 = (5)
5 - 0101
7 - 0111
5&7 → 0101 → Bin = (5)
0111 → 7
Bit Masking & Bit Manipulation

8 → 1000
9 → 1001
0001
XOR → S → 0
D → 1

Bitwise Shift Operations:→
* Value
a = 10
Step, unit
step = 2
a >> 2
8 bit
0 0 0 0 0 1 0 1 0
→ 0 0 (added to balance) 0 0 0 0 0 0 1 0 ~~1 0~~ discarded
10>>2 = 2
↓
Right shift the values decrease

128 64 32 16 8 4 2 1
~~0 0~~ 0 0 1 0 1 0 0 0 (added to balance)
discarded
10<<2
40 Leftshift the values increase

X
T	F
F	T

0	1
1	0

Toggle

(-50) = -(-50) - 1 = 49
5 → 0101 ✗
~5 → 1010 = 10
~5 = -6 How
~5 = 10
10 kan
5
abs (-6) = 6 → 0110
negation of 2 { 2's com { 1's com 1001, +1 0001
1010 Theoretical Value
~5 (i) 10 (ii) 5 (iii) 4 (iv) -6 ✓
General rule : $\sim n = -n - 1$ $-5-1 = -6$

(-6) → 10 ~5 → 0101 1010
↓
Compiler → (-)ve
absolute abs(-6) = 6 → 0110
2's com { 1's 1001, +1 0001
1010 (TV)

$\sim n = (-n-1)$
MCQs

F2F Viva

* Conditional Statements:
(i) Simple if statement → one 1 condition
(ii) if-else statement → Exactly 2 conditions
(iii) if-else-if-else statement → More than 2 conditions
(iv) nested if-else statement → Branching or Nesting of conditions
(v) switch case → Better version of (iii)
(vi) ternary Operator → Short hand if-else