Q66. If the format specifier doesn't match the type?

→ It gives garbage or unexpected output, since it reads the wrong number of bytes.

Q67. Difference between scanf and printf here?

→ scanf reads a value into n, printf displays the value stored in n.

Q68. Purpose of a switch statement?

Q69. When is switch preferred over if-else-if?

Q70. Syntax of a switch statement?

Q71. Purpose of the case statement?

Q72. Why is break used inside switch?

9. switch vs. if...else-if...else

Q81. Major difference between switch and if-else-if?

→ switch checks exact matching values, if-else-if can check ranges and complex conditions.

Q82. Can switch evaluate age >= 18 directly?

→ No, switch cannot directly evaluate relational conditions.

Q83. Why is if-else better for range-based conditions?

→ Because it can handle relational and logical expressions, which switch can't.

Q84. Why is switch suitable for fixed choices?

→ Because it's cleaner and easier to read for a known set of fixed values.

Q85. Can a triangle classification be done using switch?

→ Not directly, since it needs range comparisons — if-else is more suitable there.

Q86. Positive/negative/zero — switch or if-else?

→ if-else, because it involves relational comparisons, not fixed values.

Q87. Calculator menu — switch or if-else?

→ switch, since the operators are fixed, distinct values.

Q88. Can the same problem use both switch and if-else?

→ Yes, in many cases both can implement the same logic.

Q89. Advantages of using switch?

→ Cleaner code, easier to read, and faster for many fixed cases.

Q90. Limitations of switch compared to if-else?

→ It can't handle ranges, floats, or complex conditions the way if-else can.

10. Higher-Order / Tracing Viva Questions

Q91. Predict output of multiple assignments/arithmetic?

→ Trace step by step — apply precedence rules first, then follow each assignment in order.

Q92. Identify which operation runs first, and why?

→ Follow precedence — multiplication, division and modulus before addition and subtraction, and parentheses first if present.

Q93. Determine which if/else-if block executes?

→ Check the conditions from top to bottom — the first true one runs, the rest are skipped.

Q94. Find the logical error in an odd/even check?

→ It's usually a wrong modulus condition, or using = instead of ==.

Q95. Error where = is used instead of ==?

→ It assigns the value instead of comparing, so the condition ends up always true if the value isn't zero.

Q96. Predict output of switch with missing break?

→ Execution falls through and keeps running the next cases until it hits a break or the end.

Q97. What happens on an invalid menu choice?

→ The default case runs and shows an invalid choice message.

Q98. Classify a triangle as equilateral, isosceles, scalene?

→ All three sides equal means equilateral, two sides equal means isosceles, all different means scalene.

Q99. Trace a program with nested if statements?

→ Check the outer condition first, then move into the matching inner condition, and follow that block.

Q100. Identify data types, operators, conditions used?

→ Look at the variable declarations for data types, the symbols used for operators, and if/switch/loop keywords for control structures.