

**Question bank Pattern**

**[subject code] [ EMBEDDED C PROGRAMMING]**

**Unit 2**

**[OPERATORS ,CONDITIONAL & ITERATIVE STATEMENTS]**

**PART A (10x2)**

**Certainly! Here are 10 questions related to Unit II: Operators, Conditional & Iterative Statements in C programming, each with their key answers, and each question carrying 2 marks:**

Question 1: What is the classification of operators in C based on the number of operands they work on?

Key Answer 1: Operators in C are classified into three categories based on the number of operands they work on:

1. Unary operators (work on one operand).
2. Binary operators (work on two operands).
3. Ternary operators (work on three operands).

Question 2: Provide examples of each of the following types of operators in C: Arithmetic, Relational, Logical, Bitwise, and Assignment.

Key Answer 2:

* Arithmetic: +, -, \*, /, %
* Relational: ==, !=, <, >, <=, >=
* Logical: &&, ||, !
* Bitwise: &, |, ^, <<, >>
* Assignment: =, +=, -=, \*=, /=

Question 3: Explain the purpose of the sizeof() operator in C. Provide an example.

Key Answer 3: The sizeof() operator in C is used to determine the size in bytes of a data type or an expression. Example:

int size = sizeof(int);

In this example, size will contain the size of the int data type in bytes.

Question 4: Describe the ?: operator in C and provide an example of its usage.

Key Answer 4: The ?: operator is the conditional (ternary) operator in C. It is used for conditional expressions. Example:

int x = 5, y = 10;

int result = (x > y) ? x : y;

In this example, result will be assigned the value of x if x is greater than y, and y otherwise.

Question 5: What are conditional statements in C? Provide examples of if and if...else statements.

Key Answer 5: Conditional statements in C are used to execute different code blocks based on certain conditions. Example of if statement:

int num = 10;

if (num > 5) {

printf("The number is greater than 5.\n");

}

Example of if...else statement:

int num = 3;

if (num > 5) {

printf("The number is greater than 5.\n");

} else {

printf("The number is not greater than 5.\n");

}

Question 6: What is a nested if...else statement in C? Provide an example.

Key Answer 6: A nested if...else statement in C is an if...else statement that is placed inside another if or else block. Example:

int x = 10, y = 20;

if (x > 5) {

if (y > 15) {

printf("Both x and y are greater than 5 and 15, respectively.\n");

} else {

printf("x is greater than 5, but y is not greater than 15.\n");

}

} else {

printf("x is not greater than 5.\n");

}

Question 7: Explain the purpose of the switch statement in C, and provide an example.

Key Answer 7: The switch statement in C is used for multi-way branching based on the value of an expression. Example:

int day = 3;

switch (day) {

case 1:

printf("Monday\n");

break;

case 2:

printf("Tuesday\n");

break;

case 3:

printf("Wednesday\n");

break;

default:

printf("Other day\n");

}

Question 8: Describe the while loop in C, and provide an example.

Key Answer 8: The while loop in C is used for repetitive execution of a block of code as long as a specified condition is true. Example:

int i = 1;

while (i <= 5) {

printf("%d\n", i);

i++;

}

Question 9: Explain the purpose of the do...while loop in C, and provide an example.

Key Answer 9: The do...while loop in C is used for repetitive execution of a block of code at least once, and then as long as a specified condition is true. Example:

int i = 1;

do {

printf("%d\n", i);

i++;

} while (i <= 5);

Question 10: Describe the for loop in C, and provide an example.

Key Answer 10: The for loop in C is used for repetitive execution of a block of code with initialization, condition, and update expressions. Example:

for (int i = 1; i <= 5; i++) {

printf("%d\n", i);

}

**Part B (5\*13)**

Program 1 :Write a C program that calculates the factorial of a given non-negative integer using a for loop. Ensure that the program handles input validation.

Key Answer 1 :

#include <stdio.h>

int main() {

int n;

long long factorial = 1;

printf("Enter a non-negative integer: ");

scanf("%d", &n);

if (n < 0) {

printf("Factorial is not defined for negative numbers.\n");

} else {

for (int i = 1; i <= n; i++) {

factorial \*= i;

}

printf("Factorial of %d is %lld\n", n, factorial);

}

return 0;

}

Program 2: Write a C program to find and print the Fibonacci series up to a given number n using a while loop.

Key Answer 2 :

#include <stdio.h>

int main() {

int n, a = 0, b = 1, c = 0;

printf("Enter a positive integer 'n': ");

scanf("%d", &n);

printf("Fibonacci Series up to %d: ", n);

while (c <= n) {

printf("%d, ", c);

a = b;

b = c;

c = a + b;

}

printf("\n");

return 0;

}

Program 3 : Write a C program to find the largest among three numbers using nested if...else statements.

Key Answer 3 :

#include <stdio.h>

int main() {

double num1, num2, num3;

printf("Enter three numbers: ");

scanf("%lf %lf %lf", &num1, &num2, &num3);

if (num1 >= num2) {

if (num1 >= num3) {

printf("%.2lf is the largest.\n", num1);

} else {

printf("%.2lf is the largest.\n", num3);

}

} else {

if (num2 >= num3) {

printf("%.2lf is the largest.\n", num2);

} else {

printf("%.2lf is the largest.\n", num3);

}

}

return 0;

}

Program 4 : Write a C program that calculates the sum of natural numbers up to a given positive integer n using a do...while loop.

Key Answer 4 (13 marks):

#include <stdio.h>

int main() {

int n, sum = 0, i = 1;

printf("Enter a positive integer 'n': ");

scanf("%d", &n);

do {

sum += i;

i++;

} while (i <= n);

printf("Sum of natural numbers up to %d is %d\n", n, sum);

return 0;

}

Program 5 : Write a C program to implement a simple calculator that performs addition, subtraction, multiplication, or division based on user input. Use a switch statement to handle different operations.

Key Answer 5 (13 marks):

#include <stdio.h>

int main() {

char operator;

double num1, num2;

printf("Enter an operator (+, -, \*, /): ");

scanf(" %c", &operator);

printf("Enter two numbers: ");

scanf("%lf %lf", &num1, &num2);

switch (operator) {

case '+':

printf("%.2lf + %.2lf = %.2lf\n", num1, num2, num1 + num2);

break;

case '-':

printf("%.2lf - %.2lf = %.2lf\n", num1, num2, num1 - num2);

break;

case '\*':

printf("%.2lf \* %.2lf = %.2lf\n", num1, num2, num1 \* num2);

break;

case '/':

if (num2 != 0) {

printf("%.2lf / %.2lf = %.2lf\n", num1, num2, num1 / num2);

} else {

printf("Error: Division by zero!\n");

}

break;

default:

printf("Invalid operator!\n");

}

return 0;

}