**Decision Control Structures**

1. **If Statement**  
   The if statement is used to execute a block of code if a specified condition is true.  
   **Syntax**:

c

if (condition) {

// Code to execute if condition is true

}

**Example**:

int a = 5;

if (a > 0) {

printf("a is positive");

}

**Explanation**: If a > 0, it prints "a is positive."

1. **If-else Statement**  
   The if-else statement adds an alternative block of code if the condition is false.  
   **Syntax**:

if (condition) {

// Code if condition is true

} else {

// Code if condition is false

}

**Example**:

int a = -5;

if (a > 0) {

printf("a is positive");

} else {

printf("a is negative");

}

**Explanation**: If a is not greater than 0, it prints "a is negative."

1. **Nested If**  
   The nested if is used when one if statement is inside another if statement.  
   **Syntax**:

if (condition1) {

if (condition2) {

// Code if both conditions are true

}

}

**Example**:

int a = 5, b = 3;

if (a > 0) {

if (b > 0) {

printf("Both are positive");

}

}

1. **If-else Ladder**  
   This is a chain of if-else statements used when multiple conditions are checked in sequence.  
   **Syntax**:

if (condition1) {

// Code for condition1

} else if (condition2) {

// Code for condition2

} else {

// Code if no condition is true

}

**Example**:

int marks = 85;

if (marks >= 90) {

printf("Grade A");

} else if (marks >= 75) {

printf("Grade B");

} else {

printf("Grade C");

}

1. **Switch Case Statement**  
   The switch statement allows choosing between multiple cases based on the value of a variable.  
   **Syntax**:

switch (variable) {

case value1:

// Code for value1

break;

case value2:

// Code for value2

break;

default:

// Code if none of the values match

}

**Example**:

int day = 2;

switch (day) {

case 1:

printf("Monday");

break;

case 2:

printf("Tuesday");

break;

default:

printf("Other day");

}

**Explanation:** The switch case for 2 is executed, so "Tuesday" is printed.

**Iterative Statements**

1. **For Loop**  
   The for loop is used when the number of iterations is known in advance.  
   **Syntax**:

for (initialization; condition; increment) {

// Code to be repeated

}

**Example**:

for (int i = 0; i < 5; i++) {

printf("%d ", i);

}

1. **While Loop**  
   The while loop is used when the number of iterations is not known in advance. The loop runs as long as the condition is true.  
   **Syntax**:

while (condition) {

// Code to be repeated

}

**Example**:

int i = 0;

while (i < 5) {

printf("%d ", i);

i++;

}

1. **Do-while Loop**  
   The do-while loop ensures that the code runs at least once before checking the condition.  
   **Syntax**:

do {

// Code to be repeated

} while (condition);

**Example**:

int i = 0;

do {

printf("%d ", i);

i++;

} while (i < 5);